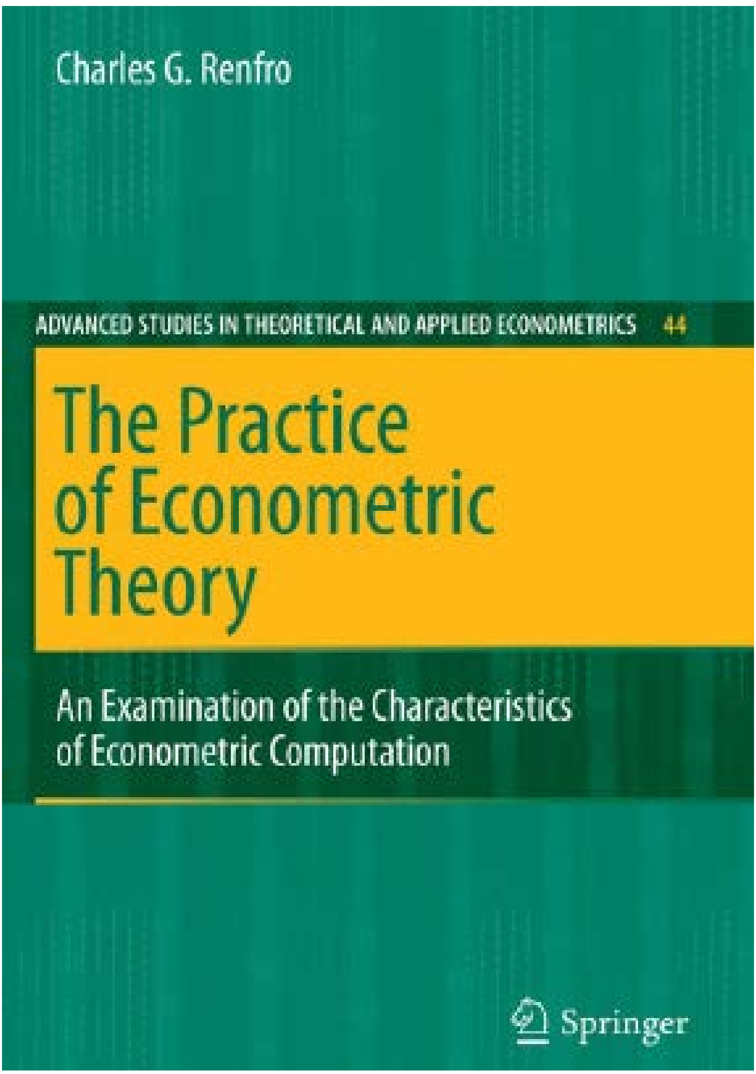
Charles G. Renfro

The Practice of Econometric Theory

An Examination of the Characteristics of Econometric Computation



Springer 2009

Preface

Econometric theory, as presented in textbooks and the econometric literature generally, is a somewhat disparate collection of findings, rather than the well integrated organized whole it might seem at first sight. Its essential nature is to be a set of demonstrated results that increase over time, each logically based upon a specific set of axioms or assumptions, but rather than becoming collectively complete, these inevitably remain a fragmentary body of knowledge. The practice of econometric theory consists of selecting from and applying this literature, as well as simultaneously evaluating it, so as to test its applicability and range, and support its further advance.

Today this practice is closely associated with the creation, development, and use of computer software and "econometric software" is the operational expression for this theory. Originally, the development of this software focused on the implementation of a progressively enlarging set of estimators, but now its best expression involves the attempt to provide not only the means to estimate parameters in different ways but also to test each of the underlying assumptions in the most meaningful way. The argument that might be made to buttress these assertions begins from the observation that the range of estimators that have been discovered by econometricians is reasonably extensive and that some of these are particular to a specific context. However, the most generally applied estimator is Ordinary Least Squares and, except in those situations where there is a priori knowledge of its unsuitability, its role is to be the starting point. To the extent that, either in practice or in principle, OLS plays this role, the consequence is then to give a particular importance to the set of supplementary evaluative tests that are applied in conjunction with it.

Following from these considerations, this monograph presents, classifies, and documents the particular diagnostic tests associated with Ordinary Least Squares regression that are provided by the existing econometric software packages, with the goal of supplying reliable and useful information to three categories of people: econometric software developers, econometric theorists, and more generally those economists who use this software. Towards this end, it attempts to both discover and evaluate the present state of the art. The research behind it has been undertaken in the form of an interactive survey, conducted not as an external examination but instead with the active assistance and collaboration of the econometricians who have created, designed and developed these packages. Furthermore, this investigation has been embarked upon with certain specific fact-finding intentions. One of these is to provide a generally useful set of benchmark values of these diagnostic statistics. Another is to determine how the individual tests have been implemented, assessing both their degree of commonality and, wherever there are differences, why these have occurred. However, this study should also be viewed in a broader context: it is one of an ongoing series of contributions collaboratively produced by a number of people since about 1995, who together attempt to consider and assess the various computational aspects of the modern applied economics research environment. The common perspective is that the existing software packages collectively define the operational state of the art of econometrics, and at least certain aspects of the general applied economic research environment, and that therefore it is vitally important for both econometricians and applied economists to understand the specific characteristics of the research facilities available to and actually used by economists.

The statement of intent just provided clearly serves as an abstract. However, it is possible that the scope of the investigation may not yet be wholly self-evident. On the face of it, the idea of identifying, classifying, and presenting a particular set of diagnostic tests seems simple enough, but any evaluation of the state of the art must also compare what is with what could be, inevitably spawning a number of questions, among them being: how and why were the particular tests implemented by developers originally chosen? More specifically, what constitutes the most appropriate set of diagnostic tests, both individually and collectively? Is this the set actually offered by the existing packages, perhaps as a result of some type of invisible hand selection process? Can a uniquely appropriate set actually be determined? In addition, certain more general questions may also ultimately need to be addressed, among them: why limit the present study to Ordinarily Least Squares?

What is an econometric software package anyway? The definition of econometric software is actually a nice question and to answer it raises more questions, the first of which can be expressed as one of nature versus nurture; that is, should this answer be approached by first addressing the nature of econometrics – or is its content simply the evolutionary result of its nurture? The content of econometrics is of course a question that has been addressed repeatedly since the 1930s, with as yet no definitive conclusion reached. However, assuming for argument's sake that its characteristics are capable of being pinned down one might ask, can the nature of econometric software actually be determined on this basis? Or to put this last question in slightly different words, is it reasonable to expect that "econometrics" is so well defined in its subject matter that a single econometric software package could in principle serve as an operative expression of it? Might it instead be more appropriate to ask, to what degree are the existing packages an operative reflection of particular topic areas of the published econometrics literature? Given this more restricted interpretation, is it then possible that the match between the literature and the software offerings is less than exact, not only in subject matter but also in terms of the specific characteristics of the algorithmic implementation?

These are all questions that finally need to be considered, but at the outset rather than attempt to answer them directly, particularly in the order asked, it might be better to approach the matter in a somewhat crabwise fashion, beginning by asking first, what is software? Fortunately, the word "software," generally referring to the set of instructions that cause an electronic computer to operate in a particular way, is nowadays a concept that is familiar to anyone who has used such a device, particularly during the past approximately thirty years, since the introduction of the microcomputer. "Hardware," in contrast, is of course the physical substance of a computer, without the electricity. These machines are now ubiquitous and their use ordinarily requires conscious interaction: from time to time, a user must "boot" and, occasionally, "restart" the computer in a rather direct, hands-on manner, as well as both "install" and "execute" software packages. Once the operating system and so-called "applications" packages are in execution, they cause the machine to perform its useful work. The idea of "software" therefore requires little explanation as a general concept, but there is nonetheless an evident degree of ambiguity inherent in the compound phrase "econometric software." Specifically, the definition of this software might be considered from any of several perspectives: as software created *for* economists and econometricians or *by* them, or that takes as its subject matter the set of econometric techniques, or, possibly, simply as being that software that economists or econometricians happen to choose to use professionally.

However, most fundamentally the critical issue is how this software shapes the economist or econometrician's interaction with his or her research materials and the specific way in which theories and hypotheses are then confronted by empirical evidence. Notice that the just mentioned alternative perspectives are each potentially important to a particular evolutionary consideration of the computational methods employed by economists. For example, "happening to choose," as a possible historical explanation, would appear to imply at least a degree of exogenous influence by some other discipline on the creation of this software, since the software's prior existence is thereby indicated. Furthermore, if this ex post choice is *commonly* made, it also implies at least a touch of disciplinary instability; that is, if economists persistently choose to use "other" software, rather than what has been developed endogenously, the extreme implication ultimately might be either the ongoing or a one-time reorientation of economics, if not econometrics, because of the software that happens to be used. After all, the tools employed can affect the specific work done and hence the results obtained. Therefore, in the end, the evolutionary outcome could finally be research that is done in conformity with and is shaped by some "other" tradition, then affecting for better or worse the progress of econometrics.

In contrast, the several questions raised earlier about the inherent nature of econometric software obviously take as given a presumed global stability, implicitly interpreting such software as being created in conformity with the existing econometrics literature, with the techniques offered originally established by that literature and then subsequently affecting its ongoing development. However, when such questions are considered in this endogenous context, casting the definition in terms of either *who* creates the software or *for whom,* such considerations in each case still potentially raise questions concerning the likely evolutionary path of economics and econometrics. *Who* creates the software, for instance, can result in unexpected effects if the creators bring to the task an imperfect understanding of the current practice of econometrics. *For whom*, in contrast, suggests the existence of a strongly held, or at least essentially self-conscious, concept of the particular practices of economists and econometricians. It is possible of course to view at least certain of these various potentialities as tenuous, but in the general scheme of things it is important to give each of them some consideration, for none can be simply rejected out of hand as being preposterous.

The fundamental question that is addressed by this study is the present state of the art and an important consideration is how this came about. Rejecting both chance and instability as likely possibilities, the selection of software for the present study was made on the basis of the stated intentions of the developers of the programs chosen, which initially required some investigation to identify the population of candidate packages. It was first necessary to examine the relevant economics and statistical literature, including the various lists of software that have been advertised or posted on pertinent Internet websites during the past five to ten years as being "of interest" to economists, particularly sites that aim to identify "resources for economists," such as www.rfe.org. Once a unified population list had been thus created, the developers or vendors of these packages, as relevant, were contacted and each simply asked if the software they offered was *intended* to be "econometric software." This "sampling" method is obviously somewhat informal and of course left open the possibility that a given supplier might make an unsustainable claim, or that an appropriate package might remain undiscovered, but actually each of these potential problems were easy to minimize. To minimize wrongful inclusion, each selected developer needed to provide information and participate actively, to continue as a member of the set. To minimize wrongful exclusion, the list of selected packages has been advertised widely among economists since early 2003. Initially, this list was circulated to discover possible additional candidate software packages. Later, in 2004, a compendium of these packages, complete with developer-provided descriptions, was published as both a chapter of a book and a paper in a special issue of the *Journal of Economic and Social Measurement*. Simultaneously, a fairly detailed study was made of the various design characteristics of the included packages and the findings also published in the book and special issue. This process has provided both publicity and a clear statement of the properties of the included packages, as well as contact information for the future. Notwithstanding any classification problems that might be associated with this approach, its evident virtue is that it permits inferences to be drawn about the characteristics of the included packages from their designers' and developers' stated intentions. Furthermore, it is reasonable to expect, over time, that packages that are developer-identified as econometric are more likely to characterize and reflect econometric practice than those intentionally developed for some other purpose.

An associated consideration was the choice of data to be used to make comparisons among the different software packages. The principal data set chosen was selected well in advance. It consists of a previously published set of observations that have the virtue of being both widely available and long-known to econometricians, thus placing little burden on each software developer to acquire and use. After all, it was not obvious in advance exactly what might be discovered, whether for instance the conformity between packages would be sufficiently great as to make the comparisons to all intents and purposes a non-event. In practice, as a general choice, this data set proved to be quite satisfactory as a means of demonstrating a variety of interesting results, results that *intentionally* do not depend upon any type of stress testing or the particular (read unusual) characteristics of the data employed. It has long been known, from numerical accuracy tests developed by Wilkinson, among others, designed specifically to test for specific types of computational errors and problems, that it is often possible – with some ease – to expose particular shortcomings. In other contexts, such testing might be desirable, even as a supplement to the present study. But, in this study, the first question to be considered was: what are the differences, if any, under normal, even benign conditions? It would always be possible, later, to stress test, but there is actually much to be learned in the absence of this type of, potentially hostile, accuracy testing. Recall from the discussion above that the results reported in this volume are the result of active (and cordial) cooperation among econometric software developers, so that it was critical, from the beginning, to proceed in a way that would provide a beneficial joint learning experience. Not only was the purpose of this study to determine what values and test statistics each econometric software package might produce, but why.

As the Chinese say, each journey begins with a single step; this study constitutes that first step. However, as the comparisons began to be made, it became clear that in certain instances the original data set chosen could not produce fully informative results. Therefore, it proved useful to employ an alternative, easy to obtain, set of observations on US GDP, which specifically have been used to illustrate aspects of the computation of specific unit root test statistics. But in all cases, with the sole exception of certain historical displays replicated to illustrate former ways of presenting information, the principal numeric results shown in this volume are generated using one or the other of these data sets and the observations themselves are included in its appendix, as well as still being available from the original published sources in hard copy, if not machine-readable form.

Furthermore, to insure the possibility of replication, each of these values has been independently computed by a minimum of two econometric software packages and usually by more than two. In particular, in the case of the diagnostic tests performed in common by a number of packages, most if not all of these have been tested for agreement. Whenever differences have been discovered, a concerted effort has been made to determine why. In certain cases this evaluation has led to recoding on the part of individual developers, although the details of such changes have not been examined here: the purpose of the present study is not to track and report on the historical changes in the individual packages, but rather simply to display reproducible results produced by the existing set of econometric software packages. An objective of this study is to provide benchmarks. In the future, anyone who wishes to examine the numerical characteristics of a given package will be able to assess them in the light of these results.

A central evaluative finding of this study is that in many cases the numbers that have been produced by the surveyed packages differ in various important ways. However, it must also be added quickly that there were actually only a few differences discovered in those instances in which developers *intentionally implemented the same formulae*. Differences found almost always occurred because of differences in the ways in which individual developers independently implemented the tests, sometimes reflecting the particular identifying names used for test statistics but in other cases the implementation of variations on known formulae. Furthermore, as is discussed later, program developers in some instances have intentionally implemented a particular form of a diagnostic test, knowing in advance that they were producing a variant. Sometimes these computational differences reflect the developer's firmly held belief that this variant has better diagnostic properties. In other cases, the differences occurred inadvertently. Ideally, a benefit of studies such as the present one is that inadvertent differences will be minimized in the future.

In addition to numeric differences, when comparing one package pair-wise against another, it was also found that packages commonly differ in their range of offered diagnostic tests. There are a number of test statistics that are, or optionally can be, displayed by all or nearly all packages, but there are also a number of others provided only by one or two packages. To a degree, the observed differences characterize the recent evolution of econometric software, for during the past ten to fifteen years there has been an increasing tendency for econometric software packages to display more and more test statistics, reflecting among other things the peer review process of economics and econometrics journals, but also the more forensic methodology of modern econometrics. However, as just indicated, the particular choice of the diagnostic statistics generated and displayed has often been made by econometric software designers both competitively and independently, with the result that there are now noticeable differences between packages — at least when they are evaluated, snapshot fashion, as of a particular date. These differences may not persist over time, once publicized, but they are documented here by a set of tables that identify the test statistics offered by each of the individual packages. In contrast, the numeric results displayed are more restricted: as mentioned earlier, only values *independently* reported by two or more packages are displayed. The italics reflect that individual packages are not always developmentally independent of each other.

This display choice is of course open to criticism. The range of statistics reported by the packages collectively, as well as the fact that a particular statistic is generated uniquely by a given package, are each findings that are of course quite relevant to the present study in its attempt to discover and evaluate the state of the art. Therefore, this information is provided. However, whenever a statistic is computed uniquely or agreement between two or more independent packages could not be confirmed, the consequence was to preclude its use as a benchmark value.

Evidently, this study is restricted in scope. Ideally, it might be desirable to describe each of the existing econometric software packages individually, explaining both their present characteristics and the reason for those characteristics. There is a story to be told in each case, but it is a story that can only be told by the individual developers. Similarly, as an ideal, it might be desirable to establish a full set of benchmark values for each of the statistics reported by each of the packages. However, this goal also needs to be pursued separately later. The particular aims, form, organization, and even the conclusions of this study reflect the newness of this type of investigation. It appears to be the first of its type, so that the best that can be expected is for it to provide a limited result. On the positive side, this limited result can act as an arbitraging ploy. One of the possible consequent effects may be a greater commonality of pertinent results in the future, for inevitably any study such as this one affects the design of software packages. It makes software developers and users each more aware of the facilities provided by the software packages included in the study. It exposes the ways in which a given package differs from other packages, which in itself is a spur to developers. The developers of individual packages inevitably view certain other packages as prime competitors, so that in this case especially, the information provided by a study such as this one is, at the minimum, market informing.

More generally, the intention of this study is not to judge any particular package relative to another, or to expose the computational defects of individual packages. No attempt has been made to identify a particular set of numeric results with any given package, except in the case of certain pertinent examples, which are provided simply to illustrate the differences that can occur between packages. This study does not evaluate individual packages, but instead the econometric software package as a classification. To contrast and compare individual econometric software packages more specifically might provide useful information for users of those packages, but this course of action would distract attention from the essential findings of both commonalities and reported differences. In the final analysis, the essential purpose of this study is to play the role of a mutual learning experience for all the participating developers. Ideally, one of the effects will be to establish a collective understanding of the present state of the art.

Of course, the present investigation is also limited by its focus on a single estimation technique. The choice to consider only diagnostic tests that are directly associated with the Ordinary Least Squares parameter estimation technique can be seen as motivated by several considerations. The first is that it can be argued that one of the principal distinguishing characteristics of any econometric software package is its inclusion of this technique among the offerings, notwithstanding that it may be the inclusion of other techniques that distinguishes such packages from statistical and other types. It is the most basic of all the econometric parameter estimation techniques, as well as the one most commonly used. However, in this choice too, practicality has been allowed to dictate. Historically, there have been surprisingly few attempts to consider either the design of econometric software packages or the way in which econometric techniques have been implemented in these packages, including their numerical accuracy and the other computational characteristics of the results produced, so that it has seemed most sensible to start with what is ostensibly the base case and to work outwards from there.

Even as a first examination, rather than a complete examination, the present study could not have been produced without the active assistance, advice and support of econometric software developers individually and collectively. I am grateful to my fellow econometricians who are software developers, as well as to others who have provided information, comments and suggestions. I appreciate in particular the assistance and advice of Jerry Adams, Irma Adelman, Micah Altman, Richard Anderson, Terry Barker, David Belsley, Herman Bierens, Jon Breslaw, Guiseppe Bruno, Ian Cahill, Paul Calvert, Allin Cottrell, Clint Cummins, Kenneth Berk, James Davidson, Tom Doan, Jurgen Doornik, Mark Eisner, Ray Fair, Arthur Goldberger, Richard Goldstein, Clive Granger, William Greene, Bronwyn Hall, Robert Hall, Stephen Hall, Tim Harrison, David Hendry, Peter Hollinger, Charles C. Holt, Lawrence Klein, Robert Lacey, Cynthia Latta, Edward Leamer, James LeSage, David Lilien, James MacKinnon, Keith May, Michael McCracken, Bruce McCullough, Marc Nerlove, Ray O'Brien, William Peterson, Peter Phillips, Richard Pierse, Robert Pindyck, Brian Poi, Ross Preston, Duo Qin, David Reilly, Colin Rose, Ronald Schoenberg, Lucy Slater, Houston Stokes, Daniel Suits, Robert Summers, William Teeters, Gareth Thomas, Kenneth White, Mike Wickens, Vince Wiggins, Mark Wiley, Clifford Wymer, Achim Zeileis, and Arnold Zellner for their patient willingness to assist me. I am also indebted to Arthur Goldberger, Bronwyn Hall, David Hendry, and Peter Hollinger for kindly making available to me both early manuals and particular examples of regression displays that have been used in the preparation of this monograph. I express my especial debt to Houston Stokes for his constant willingness to provide me repeatedly with code, documentation, and analytical assistance during the process of determining the characteristics of the various computations made. I am wholly responsible for all errors of fact and omission, and for all opinions stated that are not identified as quotations.

Newagen, Maine *Charles G. Renfro*
12 January 2007

Contents

Introduction 1

1 Econometric Computation 19
The Foundations of Computational Econometrics 28
Towards the Evaluation of Econometric Software 37
The Discovery of Econometric Software 38
The Computer as a Computational Medium 44
The Computer as an Imperfect Computational Device 47
Errors in Data as a Computational Circumstance 51

2 Econometric Software: Characteristics, Users, and Developers 55
Developmental Characteristics of Econometric Software 57
The Early History of Econometric Computation 58
The Takeoff Period of Econometric Software Development 60
The Adoption of the Microcomputer 66
The Characteristics of Econometric Software 68
Aspects of the Evolution of Software Features 72
The Development of the Human Interface 75
Directives Versus Constructive Commands 80
Developers, Users, and Use 86
Use and Users 87
Econometric Software Developers 94

3 Econometric Diagnostic Tests 97
Diagnostic Tests: A Division of Responsibilities 105
Realism Versus Instrumentalism 111
Specific Survey Characteristics 117

4 The Basic Statistics 123
The Historical Display of OLS Regression Results 126
Survey Results: The Core Statistics 135
The Core Statistics: The First Group 138
Known Special Cases 140

Variant Special Cases ... 141
Disturbance Properties Tests ... 145
Provisions for User-Constructed Tests ... 147

5 The Failure of Assumptions ... 151
Heteroscedasticity ... 154
Disturbance Properties ... 160
Specification Tests: Functional Form, Nonlinearity, and Simultaneity ... 165
Structural Stability Tests ... 166
Omitted Variables, Linear Restrictions and Related Tests ... 173

6 Cointegration and Alternative Specifications ... 177
Unit Root Tests ... 180
Evaluating Cointegration ... 207
Encompassing and Non-Nested Specifications Tests ... 209
Chapter Appendix ... 212
Grunfeld Data ... 212
Greene Data ... 221

7 Several Historical Considerations ... 231
Changes in Regression Displays 1969–2007 ... 233
MODLER Regression Output 1969–1970 ... 235
TSP Regression Output 1978–2007 ... 241
PcGive Regression Output 1985–2007 ... 247
Data Management Issues ... 252

8 The Implications of the Findings ... 265

Appendix A Version Information for the Surveyed Econometric Software Packages ... 271

Appendix B Grunfeld's Investment Theory: Time Series Data on General Electric and Westinghouse (Theil, 1971, Table 7.1, p. 296) ... 273

Appendix C U.S. Quarterly Macroeconomic Data Originally Used by Cecchetti & Rich (Greene, 2008; Cecchetti & Rich, 2001) ... 275

References ... 283

Index ... 305

Introduction

This study examines multiple aspects of the way in which the development of computer software, specifically econometric software, has affected and, in the future, might affect the practice of econometrics. Its publication is preceded by over 50 years of software development and use, during which time economists have paid little attention to the possibility that the particular computational tools they employ might have any discernable impact on their research, and which therefore might imply the lack of perceived need for such a study. Except for the speed of calculation effect of the electronic computer, which has long been recognized, software as an affective econometric tool is a novel idea. Indeed, historically, econometricians have usually interpreted the "tools of econometrics" to be its conceptual methods, often considered somewhat abstractly. For example, in 1966, under this rubric and when considering the econometric testing of an hypothetical statement about the empirical world, Jacob Marschak (1966), following Harold Hotelling, distinguished between economic theory-based maintained, or "prior" propositions, as assumptions or "specifications," in contrast to those properties to be immediately tested against observation. He optimistically characterized these specifications as possibly derived from prior observation, perhaps as a result of sequential testing, although he spoke of models and "structures" in a fashion that to modern ears might seem somewhat anachronistic. The idea of testing being a truth discovery process is implicit in his argument, perhaps stemming from a common acceptance then of at least quasi-axiomatic foundations for economic theory. Yet he also recognized, in a way that is still up-to-date (p. ix), the difficulty the economist has in assigning "future validity to the patterns of the past. For policy change may consist in changing the very mechanism by which the environment influences economic variables," requiring that the economist "must therefore peek in the interior of the notorious "black box" that operated in the past and describe policy changes as specific changes of that interior." This recognition of a policy inspired requirement to represent economic phenomena in a manner so as to permit the economist to make out-of-sample predictions using structural knowledge is historically significant, for the distinction that Marschak made between the "black box" reduced form and its corresponding, possibly changeable structural representation involves of course both the identification problem and the well-known Cowles Commission methodology (Marschak, 1953). Directly and indirectly, a consequence of this distinction was

C.G. Renfro, *The Practice of Econometric Theory*, Advanced Studies in Theoretical and Applied Econometrics 44, DOI 10.1007/978-3-540-75571-5,

the adoption of a conceptual approach, beginning in the 1940s (Haavelmo, 1944; Hood & Koopmans, 1953; Koopmans, 1950; Morgan, 1990; Qin, 1993), that then gave rise to a substantial econometrics literature.

However, before considering further this literature and subsequent developments, it is important to take note of the fact that the 1940s, when the original constructs and conceptual framework of this approach were conceived, was also a time during which only the first baby steps were being taken towards the creation of modern computational facilities. In addition, in retrospect these years mark also an early and formative stage in the development of the capability both to measure a broad range of economic concepts and to make these measurements widely available in a timely manner. In the 1930s or before, any economist who wished to conduct applied research was often first required, as an individual, to collect or at least gather and compile the set of observations used, if not always transform them for use, a situation that persisted even into the 1950s, or arguably even much later (Barger & Klein, 1954; Klein, 1950; Klein & Goldberger, 1955; Tinbergen, 1939). The modern, somewhat centralized, organized provision of macroeconomic statistics by governmental agencies, international organizations, and other data sources was only just beginning in the 1940s (Carson, 1975; Foss, 1983; Kendrick, 1995; Kenessey, 1994; Stone, 1997). The organized collection and widespread availability of microeconomic data for applied economic research is a more recent development, particularly in the form of both cross-section and panel data sets. Of course, some enterprising economists still conduct their research in soup to nuts fashion, but this is now the exception, rather than the rule.

This 1940s coincidence of circumstance is of greater present day import than it might at first seem. Both the computational advances of the past approximately 60 years and the improvements in data quality and availability might be viewed initially as wholly beneficial effects of specialization, providentially aided by technological progress. The productivity implications of the electronic computer are obvious, as are also those of the economist's present capability to obtain, almost as a gift, substantial quantities of economic data in machine-readable form. As a consequence, economists have been able to conduct applied research on a scale and at a level of detail that would have astounded Marshall, Mill, and Wicksell, or even Fisher, Keynes, and Schumpeter. However, when the good fairy gives, the bad fairy often manages to take something back.

Not invariably, but often over these years, the process of making measurements and organizing them for use, including providing the accounting framework, where appropriate, has become ever more the sanctioned province of the economic statistician, in effect thereby freeing the applied economist from the necessity. National income statistics, index numbers, and other such economic measurements of course predate their formal econometric use, but given the accounting framework supplied by Keynes' *General Theory* and *How to Pay for the War* (Hicks, 1990; Keynes, 1940; Kurabashi, 1994) and increased government support for the collection and dissemination of aggregate economic statistics in the post World War II period, there has been progressively a more pronounced tendency for the economic analyst to "outsource" to the economic statistician this measurement function. Likewise, cross-section and panel microeconomic data sets are also often collected separately and then made available to analysts. Although there is still a tendency in some countries and contexts to view both primary data and economic statistics proprietarily, more often governmental statistical organizations and others now readily provide these data (not always free of charge), sometimes in the form of public use samples and other times as aggregates or in another guise, to the point, most recently, that they are frequently supplied in electronic form via the Internet. For a balanced view, it is important to remember that to achieve this type of ready-access to data took many years. Even into the 1990s, although the economist no longer necessarily needed to make original measurements and take responsibility for the initial organization of economic statistics, data acquisition could be troublesome, sometimes involving the need for the end user to keypunch from "hardcopy" (Renfro, 1997, 1980). Certain significant transmission barriers still remain (Harrison & Renfro, 2004), as will be considered in Chap. 7. And of course, as mentioned, data collection or compilation has to a degree continued to be an integral part of applied research, but the predominant tendency is for the economic analyst to obtain data providentially.

Computational developments have followed a generally similar course. In the earlier years – between 1951 and in some cases possibly even as late as the 1980s – to apply the full range of econometric methods, or in the earliest years even a subset of these, required computer programming skills, or a willingness to use older, more manual methods (Desai, 2007). With the necessity then for the entire modern computational infrastructure to be developed, including not only the creation of particular econometric software packages but also computer programming languages, subroutine libraries, and more powerful operating systems, the capacity of the typical applied economist to employ even a few of the econometric methods described in the burgeoning literature was initially quite limited, a circumstance that persisted for years. To bring about the change that has occurred, the software developer might be viewed as having played a parallel role to the economic statistician. The ultimate outcome would appear to be to provide on the desktop what, with qualifications, might be seen as the capacity for any interested economist to employ almost any (standard) econometric method using any of a number of widely available software packages. However, to the degree that this result has been achieved, it has occurred only during the past ten years, providing another point of similarity.

A pertinent aspect of these historical developments is that they also involve significant externalities. Considered generally, one of the effects of "outsourcing" the provision of much of the data used for research has been some loss of control by the economist over the way in which economic measurements are made, as well as a loss of information about their particular characteristics, to the point that it is not always evident that all economists understand that theoretical concepts can sometimes differ significantly from the ostensibly equivalent measured concept. As a rule, the economic statistician neither ordinarily operates under the control of the analyst nor necessarily considers economic research requirements when establishing collection and data construction methodologies, as Lawrence Klein for example recognized in 1950 (Klein, 1950, p. 123ff). An additional consequent circumstance is that the analyst necessarily depends for information about this measurement process upon whatever descriptions the originating data source provides at whatever degree of generality or detail. Of course, the argument can be made that this outsourcing has occurred "within the family," inasmuch as the economic analyst and the economic statistician may share training and other background attributes – and may even jointly evaluate existing and prospective measurement methodologies. But it is true nevertheless that this functional separation potentially creates divergent incentives.

On the other hand, it is also possible to ask (although possibly difficult to answer) just how much freedom economic statisticians have had to select between alternative methodologies and how this freedom has lead them to make choices significantly different from those the research economist might have made instead? The economic statistician performs under various constraints. In many important cases, the primary data that are used to construct final economic measurements are originally collected in order to fulfill an administrative function (Alexander & Jabine, 1980; Cartwright, 1983; Cartwright & Aarmknecht, 1980; David & Robbin, 1981; Kleiner, 1980). Often political or administrative considerations may otherwise dictate the characteristics of the final data sets produced (Heckman, 2000; Keller & Wansbeek, 1983; Manser, 1992; McGuckin & Nguyen, 1990; McKelvey & de Leeuw, 1982; Popkin, 1993; Triplett, 1991, 1993). The environmental circumstances under which the data used for economic research are originally obtained and processed may therefore be more determining than exactly who performs these tasks. These are each important circumstances that can have particular effects, but what is fundamental is that applied research findings can be affected both by the measurement characteristics of the data used and by what the analyst does not know concerning them.

Of course, there are also additional, potentially related issues that might be examined, such as whether the statistical properties of such data make them suitable or not for modeling and testing using econometric methods (Spanos, 1995). However, whenever econometric theorists address this question, they ordinarily consider it in the context of a much more restricted measurement circumstance, usually involving the presupposition that, in the first place, the observations accurately represent the economic concepts to which they ostensibly correspond. From this perspective, judging by the literature, the relevant question for the theorist to consider is then commonly seen to be the econometric methodology to employ, once having obtained data that *as measurements* have properties that are in some sense contextually optimal, given the one or more concepts measured. Notice that the division of labor between the econometric theorist and the applied economist that thereby occurs is for the theorist simply to become a supplier of tools – tools in the sense of econometric techniques and methods – and for the applied economist to be left with the responsibility to recognize which tool should be applied in which context. The operational problem this labor division poses is that whatever the theorist's ultimate findings they may have at best limited applicability empirically, for in any likely real world circumstance the most appropriate "tool" for the applied economist to use may not yet have been created. As a practical matter, what is desirable (indeed necessary) is for the theorist to have considered carefully measurements that, as observations, have empirically realized properties. At the end of the day, the economic measurements available to the applied economist are what they are and must be dealt with on their own terms, rather than as measurements the econometrician might wish for.

On the computational side, among the questions that need to be considered is the degree to which the econometric software developer and the user of that software have divergent interests and incentives? One might also ask more generally, exactly how do the characteristics of particular software affect applied research? These are each important questions, for in the case of the first, the user of the software, the economic analyst, is clearly just as dependent on the good offices of the software developer as upon those of the economic statistician, both to make accurate and appropriate calculations and for necessary information. The software developer, in contrast, may not feel the need, nor necessarily be able, to provide software that is at once uniformly accurate, theoretically well founded, and capable of permitting the analyst to perform his or her research in the most efficacious manner. It is by no means self-evident that the econometric techniques so far proposed by theoreticians are each capable of being implemented accurately or, if so implemented, will provide empirically a theory-justifiable result. It is in addition difficult for the developer to know how to design software that is both easy-to-use and leads the economist to employ it in an effective manner. There is also the direct question of incentives, pecuniary and non-pecuniary; as to the latter, the econometric theorist occupies an honored place in the academic pantheon, but the econometric software developer, as such, does not, notwithstanding that it can be argued that both bear a similar level of responsibility.

McCullough & Vinod (1999, p. 633) have recently considered aspects of the marketplace for econometric software and have argued that

> Apart from cost considerations, economists generally choose their software by its user-friendliness or for specialized features. They rarely worry whether the answer provided by the software is correct (i.e., whether the software is reliable). The economist, whose degree is not in computer science, can hardly be faulted for this: is it not the job of the software developer to ensure reliability?

There is an evident element of rhetoric in these words, however, taking the McCullough-Vinod assertions and question at face value, what is the mechanism whereby the econometric software developer is encouraged to supply what the research economist needs, including reliable software, but may neither demand (assuming it to be already supplied) nor be willing to pay for? Does easy-to-use, as perceived by the typical economist, unambiguously equate to able-to-use well?

Until recently, topics like these have not been openly considered in the economic and econometric journal and broader literature, notwithstanding the frequent mention of the computer itself, beginning in about 1960 (Adelman & Adelman, 1959; Adelman, 2007; Klein, 1960). The idea that the use of this device might involve any particular interpretative or evaluative difficulties, either potentially or actually, only began to be addressed to any significant degree in the later 1990s (McCullough, 1997; Renfro, 1997). Before that, although not entirely unnoticed (Zellner & Thornber, 1966), there was very little, and then only sporadic, discussion of these aspects of the research use of the computer, even if, to a degree, they have been considered in the (arguably closely associated) statistics literature (Berk, 1987; Cohen, 1983; Francis, 1981; Wetherill, Curram, Burman, Duncombe, Hailstone, & Kempson, 1985). In addition, as will be demonstrated later, the economics and econometrics literature also exhibits a rather serene sense of computational timelessness beginning in the 1960s. Year to year, to the degree that computer capabilities and use have been described at all in this literature, their characteristics have been portrayed as being much the same, notwithstanding the actual, rather fundamental technological change during the past almost 50 years. For instance, a reader, however careful, will be hard put to identify, on the basis of this reading alone, what the precise successive effects of the mainframe, minicomputer, and personal computer, or of the Internet, have been, in their impact on research methodology, practices, or results. Of course, not every incremental change that occurred during the progress of this technological caravan had any necessary significance; for example, it might have been hard for anyone at the time to perceive the impact of many of the individual hardware changes.

However, overall there was a sea change, the like of which might never again occur, particularly in any future equal-size time period. Evidence of the magnitude of the change only very dimly appears – or, rather almost wholly does not – in the majority of the articles written by economists and econometricians. It is the lack of information that is especially notable. Among other things, only occasionally is it possible to determine which software package was used in a particular research project, or anything of its characteristics, or sometimes even *if* one was used – although that one must have been used may be evident from the nature of the computations performed. Not to describe the software employed is still normal for economists, but, as will be discussed in Chap. 2, this convention strongly contrasts with the research practices in other disciplines, for example, those in the biological and medical sciences. More generally, this silence may be symptomatically more indicative of the state of the art of applied economic research than much of what has been reported.

As a consequence of these circumstances, the inferences about applied research practices able to be drawn reliably by a reader of the economics and econometrics literature are now limited. Those *likely* to be drawn will often rest upon a number of questionable presumptions. Considering first the beliefs of the typical analyst, he or she often appears to think, judging in part from the lack of comment to the contrary, that the data have a particular, well-understood meaning as empirical measurements. There is also seeming faith that the software calculations performed are appropriate (McCullough, 1997; McCullough & Vinod, 1999). Yet there are also occasional indications of doubt. Economists who have revealingly considered aspects of the relationship between economic concepts and economic observations include Blinder and Deaton (1985), Eisner (1989), Griliches (1985, 1986), Leontief (1971), Morgenstern (1960), Slesnick (1998), and Wilcox (1998), although, of course, not all problems addressed are necessarily related to the use of the computer (Renfro, 1980, 2006). Furthermore, historical examples of research that explicitly consider the measurement characteristics of the data used certainly exist, such as Friedman's work on the consumption function. Of course, separate studies also continue to be made the explicit purpose of which is to examine measurement topics. But recently, with the development of the Internet, and as its role in worldwide data dissemination has progressively enlarged, increasingly data sets appear to have come to be accepted more and more on trust by analysts, rather than after painstaking inspection and evaluation.

This tendency to accept on trust may to a degree be a consequence of the modern greater separation between the process of making measurements and the research use of the resulting data, for relatively few economists now either need or wish to perform the roles of both data producer and user during their careers, either simultaneously or serially. As indicated earlier, it appears to be progressively less true that each individual analyst, or a closely supervised research assistant, will necessarily have compiled the data sets used or even understand from past experience the nuances of their measurement qualities. A lack of interest, familiarity, and possibly also respect for data collection and production activities may be fostered not only by this self-reinforcing separation, but also by the relatively few incentives that exist for economic research workers to take the trouble to ferret out the necessary information to understand in detail the particular measurement characteristics of the data they use, as will be considered later.

Similarly, there is a reinforced separation between software development and the analyst's use, reflecting both the skills required for software development and the current availability of already developed packages. Because the typical analyst often – perhaps usually or even almost always – employs software created by others, so long as the chosen package appears to work, there are obvious "don't rock the boat" incentives simply to presume that the operations said to be performed are those actually performed, especially to the degree it seems difficult or even impossible for that person to self-validate the software used – or to discover in the mainstream literature its reliability in each context. Even when curious, the analyst may often find it politic not to ask any questions, for it is not always clear exactly who to ask or which specific evaluative questions to ask. And, whenever there is no requirement to identify to others the particulars of the program used, there is yet another reason not to probe too deeply. See no evil, hear no evil; don't ask, don't tell. Today, for the analyst the most rational self-preserving action is to use a program that appears to be widely employed, which might also be construed to imply that someone else *must* have previously assessed its reliability.

In the absence of contradictory information, this situation obviously also imposes on the reader of published research reports almost the necessity to presume – often, perhaps almost always (Anderson, Greene, McCullough, & Vinod, 2007; McCullough & Vinod, 1999) – that the calculations performed are accurately described by the author and that any descriptions of the data used can be given full faith and credit, involving not only the absence of such things as transcription errors and "computer bugs," but also the expectation that the software used actually performed the described operations. Sports events commonly are viewed with the aid of instant recall, but in contrast published applied economic research normally offers little either to enable the subsequent replication of results or to assure the reader in some other confirmatory way that the "facts" presented are the facts (Anderson, 2006; Anderson et al. 2007; Dewald, Thursby, & Anderson, 1986; McCullough, McGeary, & Harrison, 2006; McCullough & Vinod, 2003). The lure of the surrogate production function for the father has seemingly been replaced by the charm of "stylized facts" for the children and grandchildren. Notice that the issue is not simply assurance of the particular calculations performed, but also whether the data observations employed can, if necessary, be determined after the fact. Notably, it is still the exception, rather than the rule, that journals, for instance, require authors to provide the reader, directly or indirectly, with sufficient information to permit later replication of the reported results (Anderson, 2006; Anderson et al., 2007). However, these statements are not made in order to suggest that the software available to economists is unreliable nor should they necessarily be read as commentary about actual research practices. They are simply statements about the current faith-based research environment.

Historically, in the various well-known assessments of the methodology of econometric practice, much has been made of the process whereby reported econometric results have been obtained, perhaps via the process of executing numerous regressions and then making arbitrary choices (Leamer, 1978, 1983). However, what is addressed here is a different issue. It much more fundamentally involves the consideration of the individual calculations made each step of the way, as well as the particular data observations employed, rather than the manner in which an economist might interpret and selectively present in a publication the results earlier displayed on a succession of computer screens or paper printouts. It is important to discriminate between effects that are due to sloppy or imperfect research practices, which are not the primary subject of this monograph, and those that are intrinsic to the modern use of computer software per se, which are. However, it is nonetheless important to take account of the ways that such research practices can amplify these intrinsic effects.

As indicated earlier and will be discussed in more detail later, the electronic computer and formal econometrics are, for the most part, contemporaneous. The computational situation that originally arose from the essentially coincident early development of formal econometrics and that of the electronic computer, as this played out during the 1940s, 1950s and 1960s, has since been accentuated by the parallel development, during the past 40 years, of a time series oriented, often economically atheoretic, more statistically cast econometric literature, in some cases originating in other disciplines, that to a large degree has now been absorbed into the econometric canon (Box & Jenkins, 1984; Greene, 2003; Hendry, 1995; Sage & Melsa, 1971; Spanos, 1986). Why these circumstances are particularly important is because of the extent to which econometric estimation and testing methodology has consequently increased in scope and variety. One of the results is that the textbooks, handbooks, and other summary presentations have necessarily become increasingly more selective, reflecting the difficulty of providing, even in volumes that total 800 to a 1,000 pages, more than a partial view of econometric ideas and theory. In addition, whereas it is conventional to view the emergence of at least certain of the competing econometric "schools" during the 1970s and 1980s to be the consequence of the carefully evaluated unsatisfactory forecasting performance of certain macroeconometric models as an inherent and necessary flaw, it is equally possible to interpret this criticism as a red herring and instead to see the overall development of econometrics in the years since as increasingly focused on and sponsored by what econometricians "can do well rather than the issues that are of importance in economics" (Wickens, 1997, p. 524). Symptomatic of this focus may be the sheer number of theoretical econometric results occurring as techniques seeking an application, rather than developed in response to evident need.

The argument can be made that, rather than the current "schools" resembling warring groups struggling over a coterminous common ground, what has instead happened is that each new band has staked out an essentially separate preserve with certain overlaps. The divisions between schools isolate those that are economically theoretic from the atheoretic, and those that stress economic theory from those that more enthusiastically embrace pure econometric theory and techniques, in a manner that defies a neat cross-tabular summary (Gilbert & Qin, 2006; Spanos, 2006). A recent taxonomy by Kevin Hoover that emphasizes macroeconometric applications identifies five "main" econometric methodologies, but also various minorities (Hoover, 2006, p. 73ff). An additional distinguishable separation is between those who use time series data, often macroeconometric, and cross-section or panel data, most often microeconometric. In some cases, it is also possible to isolate user characteristics, and thus, for instance, to speak of financial econometrics (Fogler & Ganapathy, 1982; Gourieroux & Jasiak, 2001) or cliometrics (Costa, Demeulemeester, & Diebolt, 2007; Demeulemeester, & Diebolt, 2007), as specially focused applications areas that, in time, may each involve the further development of relevant econometric techniques. Econometrics can be presented, on the one hand, as a collection of estimation and testing principles and techniques, and by emphasis glorify the building of econometric methodology tools. Alternatively, it is possible to present it as a collection of discovery and testing methodologies and to emphasize both the application of its methodological tools and its continuing special relationship to economics as a discipline.

Considered simply as an expanding body of techniques, econometrics can of course be lauded for its cornucopian diversity. However, in its relationship to its origins, a revealing contrast exists at the present time between econometrics and experimental economics. Prominent practitioners of the latter have noticeably and proudly recently praised its advance as implying an increasingly important partnership between experiment and economics (Holt, 2003; Plott, 1991). Econometricians are instead more likely to begin evaluations of their chosen (sub)discipline by extolling the development of "a powerful array of statistical tools for modeling all types of data" (Spanos, 2006, p. 5), only to confess subsequently that "what is conspicuously missing from current econometric modeling are genuinely reliable methods and procedures that enable one to discriminate between the numerous models and theories that could fit the same data equally well or better" (p. 7). It is certainly true that estimation methods have been developed that can be applied to time series, cross-section and panel data sets, to both linear and nonlinear specifications. In concert, numerous (mis)specification and other tests have been proposed in a variety of journals. Set against this profusion of techniques are judgments, like that by McCloskey, that "no proposition about economic behavior has yet been overturned by econometrics" (McCloskey, 1985, p. 182), which has since been quoted by others (Keuzenkamp & Magnus, 1995, p. 5). Yet it is also significant that, in practice, actual computational experience has (still) been comparatively restricted among economists, a circumstance that as indicated earlier is masked by the long-standing tendency of econometricians to speak and write as if the existing computational facilities mirror the theoretical literature, when in fact, there are significant differences and disparities. If the modeling and testing being done is still being performed by a small group of computer savvy econometricians, rather than economists generally, it might be appropriate to regard much of the continuing hue and cry about associated research practices as no more than a tempest in a teapot. As well as the number of techniques potentially available for use and the question of which of them should be used under what circumstances, the degree and amount of that use is critically important to the question whether, for all its pretensions, econometrics now has any true relevance for economists generally?

As always, it is possible to raise yet one more red flag. But the most pertinent follow-on question is, in the present context how much does the general situation matter? Given the limited scope of the present survey-based investigation, it might appear most reasonable to put aside all these broader, potentially discordant issues and to proceed simply, emphasizing the survey findings and taking the position that, within the bounds of this study, the computational circumstances are likely to be well understood by econometricians, if not by every economist. The argument might be made that Ordinary Least Squares is an estimation methodology that is familiar to all. At some point as students, most econometricians, and many economists, might be presumed to have made the relevant calculations themselves. The fact that computational experience at the extremes could be limited is only to be expected. Solace can be taken in the thought that it is not always necessary to examine the general case first in order to consider a special case. To the degree that the issues associated with the difficult cases are not familiar, they might be considered separately and later. Thus it would appear most reasonable to begin by presenting the survey findings in a very straightforward way.

However, the very newness of this type of survey investigation of the specific characteristics of a type of software argues for a more general and measured approach. It cannot be assumed that the history of either the development of the computer or of the way in which economists have used it is necessarily familiar to everyone as a self-conscious recollection. Furthermore, although it might be supposed that the computer, as a device, is suited to the computational needs of economists, this is actually a supposition that needs examination. To consider this matter carefully is of course in keeping with a long-standing investigatory tradition: for example, economists do not automatically accept a priori that factor prices will in all circumstances necessarily be equilibrated by trade. So what is the justification to accept on faith computability as an inevitable property? One way to begin to approach this topic is to start with a history of the use of the computer by economists, describing it in White Rabbit fashion, as recently characterized by Marc Nerlove (2004), courtesy of Lewis Carroll:

> The White Rabbit put on his spectacles. "Where shall I begin, please your Majesty?" he asked. "Begin at the beginning," the King said gravely, "and go on till you come to the end; then stop."

However, such a foot-forward approach automatically assumes the relevance of the history to the audience, when in fact it may not yet be absolutely clear either that there is a history – in the sense of the past as a necessary prologue to the present – or that there is anything particular now about the way in which the modern economist interacts with the electronic computer.

The approach that has been adopted in the successive chapters of this volume is to begin, at Chap. 1, with a description of the characteristics of economists' present use of the computer, but in a way that considers briefly several aspects of the general history of computing. Computation, in the original sense of making numerical calculations, of course started in the more distant past. It developed over the centuries in an increasingly sophisticated way so as to permit substantial engineering feats and provide support for such major events as the industrial revolution, in the process also enabling the development of accounting and banking, among other foundation activities. As indicated earlier, one interpretation of the effect of the electronic computer is that calculation has simply become more rapid since this device was introduced to its first users in the very early 1950s – including economists, as it happens. The speed of the computer evidently permits a sequence of possibly complex calculations to be performed in seconds or minutes, if not always nanoseconds, that prior to its introduction might take even multiple life times to complete. This time compression, so to speak, obviously implies an advance in human analytical capabilities. But notice also that this change is not simply a matter of speed. It is instead a combination of speed with the ability, because of that speed, to organize a sequence of logical operations, among them arithmetic calculations, so as to achieve, in at least some cases, a result quite different from that possible for a person, or even a number of people, to achieve with pencil and paper in the time allotted to human life. Time compression permits the computer, as in the case of chess playing machines, to achieve by accelerated brute force what the human mind must either achieve by intuition or not at all. As a consequence, this time compression is in effect transcendental, whether or not it ever provides a foundation for the development of artificial intelligence.

This organization of a sequence of logical operations is today usually referred to as programming and this activity has certain characteristics that need to be considered at the outset. In particular, it is useful to consider both the circumstances under which this process is conceptually straightforward, and those when it is not – and to consider also why in the latter case such aspects as design and precise implementation matter. In their historical treatment of econometrics as a subject, economists and econometricians have traditionally placed the stress upon mathematical representations, including statistical results and formulations, rather than upon the numerical analytical characteristics of the corresponding calculations. The conventional assumption has almost always been that there is essentially no difference between the statement of an econometric result as a mathematical representation and in the context of a computer-resident algorithm. In contrast, the argument that will be made in Chap. 1 is that there *is* a difference and that one way to view this difference is in terms of the degree of ability to replicate results from one algorithmic implementation to the next. An obvious question is, what are the necessary and sufficient conditions to achieve this property of replicability? Considered from a slightly more philosophical point of view, replicability at will implies the ability under the right circumstances to confirm that a proposition is empirically true or, more strictly speaking, to confirm its lack of falsification. A single reported result involves simply one degree of freedom, from which valid inferences cannot be drawn.

However, there is also the question that this use of the charged terms "necessary" and "sufficient" raises, namely the degree to which mathematical and statistical representations are inherently "computable." Anyone with the least familiarity with calculus cannot help but be aware that, except in the simplest cases, the calculations associated with the computation of the point value of a derivative or the area under an integral will ordinarily involve a degree of approximation. Alternatively, Taylor's expansion is a familiar example of a mathematical representation that explicitly incorporates an approximation residual. But, in addition, there is also a broader sense in which the step from the familiar mathematical and statistical econometric representations to the computer world of algorithmic implementations can be considered – and this has to do with the degree to which it is possible to view econometric theory as having an operational significance. To put this idea in its starkest terms, to what degree can it be supposed that this theory actually has "real world" relevance? There is no question in any econometrician's mind of the circumstances under which linear Ordinary Least Squares has the property of being Best Linear Unbiased, but the associated existence proof of this property casts no light upon whether there is necessarily a real world applied economic context in which this or any other set of optimal or near optimal properties can actually be achieved.

These are several of the themes and topics that are considered in Chap. 1, not withstanding that within the scope of this chapter they cannot be exhaustively considered. This monograph is fundamentally concerned with the state of the art of econometric software as an expression of econometric theory, so that the purpose of the discussion in that chapter is not to do more than to address a series of issues that may ultimately need to be considered in greater detail in order to help to place econometrics on a secure logical foundation. Achim Zeileis (2006, p. 2988) has recently distinguished between "computational econometrics" and "econometric computation," arguing that the first of these mainly concerns mathematical or statistical methods "that require substantial computation" whereas the second involves the algorithmic translation of "econometric ideas into software." The first of these initially might appear to be more naturally associated with the existential question of computability, yet there is also a sense in which they both are. As an operational issue, econometric methods that involve substantial computation require the use of the computer, so that in both cases the question whether the property of computability can be achieved is quite relevant. However, econometric methods necessarily have an additional operational quality, which is defined by the existence (or not) of one or more states of the world in which they can be expected to be applicably valid.

Chap. 1 ends with a consideration of the presumptions that stand behind and frame the investigative survey that yields the empirical results reported in this monograph. The characteristic assumption that both econometricians and economists have almost always made about the development of econometric software is that this software represents a distillation of econometric theory. Actually, this assumption is false, for over the years not only has econometric software become the test bed of the operational relevance of received economic theory, it has in addition increasingly defined the operational limits of that theory. As is demonstrated by the findings reported here, as a group and in practice the existing econometric software packages increasingly have begun to establish the developmental program for econometric theory. This operational result has occurred as a consequence of the particular estimation methods and statistical tests that have been selected by econometric software developers, that have thereby been made available generally to economists and econometricians, not to mention the particular way these have been implemented. However, to assert such a role for this software of course presumes the existence of such a thing as "econometric software." One of the reasons to consider such matters in Chap. 1 is the finding that econometric software packages do not generate identical results, even when a given data set is used, a consequence not only of the way in which elementary calculations are made, but also the more general choice of what calculations have been performed. However, it is also relevant that the explanation for the variety of results is *not* that certain packages necessarily generate erroneous results. The explanation is more subtle, in a way that is difficult to summarize in an introduction, so for the moment, this mystery will be left unrevealed.

Once having considered in Chap. 1 certain defining aspects of econometric software, Chap. 2 provides an historical description of its development and salient properties, but also the characteristics of its use, users, and developers. The use, users, and developers have together been formative to the properties of this software, as might be expected. Some of these properties are a consequence of the particular way that computers and the computational environment have developed during the past more than 60 years. Some are a consequence of the historical pattern of the development of economic and econometric knowledge, and the way in which that knowledge has been transmitted among economists and to students, particularly by textbooks in the earlier years. Econometrics is a comparatively new sub-discipline with still-developing mores and practices, originally communicated inter-generationally by textbooks and personal contact but today increasingly by the software used. Particular properties of this software are the result of the various incentives that have conditioned its development, as well as the combined academic and commercial environment for and in which it has been created and developed.

A specific aspect of this software is the set of diagnostic tests it makes available, some of the circumstances of the development of which are considered separately in Chap. 3. This treatment in isolation reflects, in part, the specific circumstances that have lead to the software implementation of these tests, as well as the timing of that implementation. Many of the misspecification and other tests now implemented were originally conceived as many as 30–40 years ago, a few date from the 1940s and 1950s and one or two perhaps earlier. Most have been implemented widely only during the past 10–15 years. These circumstances are a consequence both of the particular historical development of econometric methodologies and of the development of the personal computer and even the Internet since the 1980s. The econometric methodology debates that punctuated the later 1970s and the 1980s created the current range of methodologies and schools of thought, but the general awareness of these schools permeated the body economic rather slowly, especially in terms of econometric practice. Not until the 1990s did the textbooks noticeably begin to consider explicitly the various tenets of these schools of thought and even then, it was mainly the publication of supplementary methodology texts that began to create the general awareness that exists today. More or less simultaneously, the advent of the personal computer and the modern distributive environment changed the context in which software is created, developed, and made available worldwide. Obviously the spread of information about this software has additionally played a part.

As just indicated, to make sense of the current offerings requires some historical perspective, so that it is perhaps not surprising that only in Chap. 4 are the survey results first presented. The statistical tests considered in that chapter are to a degree historically determined, for the majority of those displayed make up the set of "core" statistics that could be found in econometric software packages even as early as the 1970s and, in a few cases, before that. The tables presented starting in Chap. 4 are intended to reveal the statistics and tests that are offered by each of the existing packages, as well as to provide some information about their particular characteristics. The "core" statistics are general to almost all packages, but nevertheless involve differences in implementation and thus perhaps interpretation.

These statistics arguably can be interpreted to be closely associated with the characterization of the specification error. Qin and Gilbert (2001), in their consideration of "the error term", describe the various ways this unobserved specification component has been viewed by econometricians in the context of the history of time series econometrics, beginning in the 1930s, which range from an errors-in-data interpretation to a residual representation of shocks and innovations. They point out, among other things, the interpretative linkages between such opposing methods as the VAR approach and the LSE method. However, most econometric software developers seem to have followed generalized tradition in their presentation of the "core" statistics, notwithstanding that few have ever even briefly commented on the selection choices they have made. Therefore, it seems most appropriate, at the present time, to consider these statistics as simply customary, as being those that developers perceive that econometric software users expect.

The survey results that are presented in Chap. 5 are presented under the title "The Failure of Assumptions." This title is intended to refer to the standard Gauss-Markov assumptions. The statistics and tests considered are those most closely associated with these assumptions. As is pointed out early in the chapter's introduction, one way to view these assumptions collectively is to recognize that they state the characteristics of the parameter estimates that are computationally imposed by the OLS estimator on the estimates themselves, although not always: for instance, the sample residuals necessarily do not have, even ideally, the covariance properties of the disturbances (Stokes, 2004a; Theil, 1971). The tests considered in this chapter probe such pathologies as hetroscedasticity, serial correlation and other disturbance properties, improperly omitted variables, and parameter non-constancy, but also misspecification tests related to functional form, nonlinearity, and simultaneity. Fundamentally, the survey results are presented from a fact-finding perspective, rather than normatively.

The question of how to present these tests, as well as the range of tests presented, was to a degree answered by the original decision to survey the existing packages and to consider their development historically. A normative consideration of misspecification tests would need to take into account the entire universe of such tests, and perhaps also such issues as the aims and appropriateness of each. Usage is predicated on the availability of the tests, but of course availability does not determine how the typical analyst applies them. As Keuzencamp & Magnus (1995, p. 6) have recently commented, usage apparently verges on being "abundant" but the motivating purpose can be obscure:

> Why test? Sometimes one wonders about the abundance of tests reported in empirical papers, as the purpose of many of these tests is not always communicated to the reader. Occasionally, the number of test statistics reported in a paper exceeds the number of observations used in calculating them! In many cases, the implications of a positive or negative result are not made clear. If a null hypothesis that apes behave perfectly rationally is rejected at the 5% significance level, do we care? And should we be interested in the normality of the residuals, or would it be more useful to put the tests aside and read Darwin's *Origin of Species* instead? But perhaps it is inherent to our occupation as econometricians that we stick to providing statistical inferences.

Possibly one of the follow on consequences of the present survey will be to cause both econometric software developers and econometricians generally to consider the normative issues that may be exposed by the test selection that developers have made. This choice enables their use by the typical applied economist, but the Keuzencamp-Magnus comments seem to imply that this use is not very well thought out. An obvious question is how to encourage both thought and appropriate research behavior?

Incidentally, the restriction of the survey to those tests associated with Ordinary Least Squares reflects two considerations in particular. The first is simply that Ordinary Least Squares is the most obvious place to start. However, as suggested in the Preface, it is also true that this survey has constituted a beneficial learning process. In particular, it has revealed to econometric software developers what they each have done, compared to others. Because of the nature of the survey, the survey itself has in addition apparently caused the test coverage to expand. Some time for further digestion may be necessary before it will be appropriate to consider the tests made available in association with other estimators. If a further rationale is needed for the limited choice of Ordinary Least Squares, it is simply that – in the absence of a priori knowledge that would suggest a different specification and a different estimator – applied economists today still begin an investigation with OLS. This is the point at which misspecification testing begins. After that, it is not yet clear what the typical applied economist does next.

Of course, it is also possible to call into question the usefulness of this course of action. Some econometricians would advocate quite a different approach. The trouble is that econometricians have seldom considered, or perhaps even known previously, the range of facilities that either has been available or how the individual offerings have been used. The motivation for the survey was to discover some of the facts and then to present them, also illustrating by this presentation some aspects of the developmental history of the supplemental statistical tests provided by the existing packages. This thread of history, as much as anything else, explains why Chap. 6 separately presents information about cointegration tests, as well as those that involve the consideration of alternative specifications, including encompassing and non-nested specifications. From the results presented in that chapter, it quickly becomes apparent – if not obvious before – that cointegration tests have become popular with both econometric software developers and users of the software. The findings also make it clear that the explicit testing of rival specifications nevertheless represents an underdeveloped area. In the case of those packages that promote or otherwise support the so-called General-to-Specific methodology, such as PcGive, the stress has been upon a strategy that incorporates, so far as possible, rival specifications, rather than to frame tests that discriminate between distinctly different specifications. Other approaches are not always quite so self-informing in purpose, but, in any case, all tests of alternative specifications essentially rely upon a degree of overlap between them.

Chapter 7 represents a change of pace. This chapter presents a series of historical displays, from several econometric software packages, that are collectively intended to show how the "regression display" has evolved over the years. There are two aspects of the presentation that deserve particular attention. The first is the content of the displays. Regression displays in the early days exhibited less information than they do today. The second is the evolving design of these displays. It is probably fair to say that few, if any, econometric software developers have taken the time and trouble to consider deeply the ergonomics of information display when creating or updating their packages. Easy to interpret, easy to read displays are possibly not evocative of the spirit of applied economic inquiry, and perhaps as a consequence, other than as a matter of salesmanship, it is not common to find econometric software packages being promoted as easy-to-use or easy-to-interpret and understand. Of course, no developer describes his or her package as hard to use and difficult to interpret, but it is generally the econometric features where the marketing and sales emphasis lies. However, it is also true that what is displayed, and how it is displayed does affect the interpretation of the results generated. Therefore, it is worth considering how these displays have changed over the years, especially as historical examples of the output generated are still available. This availability reflects that, even in the case of the earliest of these, the developers are still active and have preserved at least a few examples of their work over the years. The interest in these displays may increase over time, to the point that they could ultimately be seen to have a particular deeper significance, but even today there is something to be learned from them as a group. It is in this spirit that Chap. 7 is provided.

Finally, Chap. 8 is the concluding chapter and in the nature of a conclusion, it acts as an epilogue to this monograph. There is in addition a general appendix that provides the data set used to generate the numerical results that are displayed throughout this monograph. Actually, these data are now also available on several websites, sometimes in a form that permits the observations to be imported easily into a number of packages. As a consequence, it may not always be necessary for anyone interested to keypunch the observations provided in this appendix, or from another hardcopy source. Still, their provision in this appendix puts them near at hand and available, should they be needed.

Chapter 1
Econometric Computation

"Arithmetic" is the elementary branch of mathematics that involves making specific calculations using operators and rules governed by a relatively simple set of algebraic principles. Until approximately the middle of the twentieth century, a common synonym for "calculation" was "computation," each of these terms then usually being understood to refer to the process of making arithmetic calculations. Furthermore, until about that time, the word "computer" was the designation for a person who professionally made these. As David Grier has pointed out (Grier, 1996, p. 53) "human computation reached its zenith in the late 1930s and early 1940s and was considered a substantial field. It had completed large, successful projects, such as the Work Projects Administration mathematical tables project and had demonstrated the effectiveness of organized computation. It had a journal, *Mathematical Tables and Other Aids to Computation*, and prominent leaders..." including well-known statisticians and mathematicians. During the seventy or so years since, a revolution has occurred, both terminologically and computationally.

The word "computer" is now of course generally understood to refer to a non-human electronic device that among its possible applications can perform calculations, but can also be used for large scale organized information storage and, most recently, for wide area communications. The variety of individual examples of these devices is considerable. They can range in weight from a ton or more to significantly less than a pound. They can vary in volume from room size to palm size, or even smaller. An equally notable aspect of today's machine is the technical change it embodies – so much so that, in at least certain of their capabilities, the most recent personal, even palm-sized devices dominate many, perhaps most room-sized computers used as recently as 15–20 years ago. In addition, this change has continued to accelerate. Arguably, when compared to the comparatively recent machines of the early 1990s, the most modern computers exhibit greater differences in capabilities, characteristics, and potential than do the 1990s machines compared to those of the early 1960s. Added to this, the computer has progressively become an increasingly ubiquitous object, immediately familiar and recognizable, whether in the form of Personal Digital Assistants (PDA), such as the Palm or Pocket PC; mobile phones, particularly "smart phones;" or desktop and notebook computers. Last but not least, "intelligent" computer chips are now to be found in a diverse range of everyday products, among them automobiles, refrigerators, and music players.

C.G. Renfro, *The Practice of Econometric Theory*, Advanced Studies in Theoretical and Applied Econometrics 44, DOI 10.1007/978-3-540-75571-5_1,

However, proximity and widespread use have not wholly domesticated this machine. Its capabilities and essential nature remain somewhat mysterious to many people, possibly reflecting its continuous change, as well as its modern diversity, but perhaps also the degree to which it failed to be fully understood from the start. In the early 1950s, the media often portrayed it as an incredibly powerful "electronic brain." As if to confirm this characteristic, one of the computer's first public acts was to forecast correctly the outcome of the 1952 US presidential election. What more might it do? Shortly thereafter, from time to time, it began to be touted as having the near term potential to acquire "artificial intelligence," raising in the minds of some the specter of the machine as a conscious controlling intellect. This threat was most famously evoked in 1968 by HAL, a soft-voiced, apparently sentient presence, cast in a central and ultimately menacing role in the then futuristic film *2001*. HAL was described as having "become operational" at Urbana, Illinois on 12 January 1997, its name an acronym for Heuristically programmed ALgorithmic computer. Yet when the computer is identified abstractly, simply by the name "computer" – rather than by example as a member of an everyday device classification – it is even now most commonly perceived to be a numerical calculation device, notwithstanding its widespread use for word processing and a variety of other more general tasks, on the desktop, laptop, or else behind the scenes as a server or in some other collective role. Of course, such evocative comparisons as that almost any modern computer can add numbers at well over a million times human speed, whereas the modern passenger jet flies (only) 150 times faster than a person walks (Knuth, 1996, p. 35), cannot help but reinforce the perception that the computer's ability to make rapid numerical calculations is its most significant characteristic.

This magnitude of calculation speed is impressive, and no less so now that it can be achieved on both desktop and laptop, but to assess this machine properly, whether in general or as used by economists, it is instructive to begin by recognizing that its capabilities fundamentally originate in its essential nature as a logical device and, following that, by next considering exactly what this might mean. The most significant characteristic of the computer is its ability to store and process not only the data used in calculations or other operations, but also the logical instructions that govern the work performed. These instructions, although individually elementary and in each case selected from a restricted, machine-dependent set, nonetheless give it the capacity to operate effectively. A logically ordered collection of these instructions, formed into a particular recipe or, more strictly speaking, an "algorithm," can be used to perform complex tasks, such as inverting a matrix. Algorithms can in turn be sequenced and, in addition, then related to each other logically. Since 1946, the process of selecting from among a fixed set of instructions in order to create a logically organized sequence of coded statements, first as algorithmic tasks and then as a collection of these to form a "program," has been commonly referred to as "programming," this term being of course the gerundive form of the modern verb "to program" (Grier, 1996).

In the early days, as a pedagogic aid, electronic computers were sometimes described as "automatic digital computers" (Wilkes, 1956), essentially in order to distinguish them, on the one hand, from human computers as "digital," rather than flesh and blood, and, on the other, from the then common electromechanical desk calculator, as operating "automatically," without step by step intervention. As a calculating device, the electromechanical desk calculator itself had only relatively recently been introduced as an improvement on pencil and paper human computation, its introduction having occurred sometime around 1930. This calculator was capable of operating on entire numbers, which removed the need for the user to perform each of the subordinate elementary arithmetic calculations step by step (Desai, 2007; Goldberger, 2004). However, each of these "gross" calculations only took place at the moment it was individually invoked by the human operator. In contrast, the "automatic" computer, by removing the need for intermediate human intervention during a sequence of calculations, provided a transformative innovation, for its automatic operation allowed numbers to be fed in as inputs and then operated upon, perhaps repeatedly and variously, in much more complex ways than possible, or even conceivable, before. Furthermore, the logic embodied in the programs stored in its memory permitted not only tasks, but also even composites of these to be performed "automatically." Of course, the "electronic" computer, operating electronically rather than electromechanically, could also perform each operation faster, but this is a qualitative issue, or at least was in the beginning, notwithstanding the important qualification that its electronic nature arguably also permitted much more efficient access to "memory." The initial effect was as just described, for subroutines and even re-usable subroutine libraries were among the first practical innovations (Wilkes, Wheeler, & Gill, 1951). Other important advances came later, effectively as enhancements. For instance, until 1957, programs needed to be created using machine or assembly language, at which point "high level" procedural languages were introduced, originally in the form of Fortran (FORmula TRANslation) (Backus et al., 1957), followed in relatively short order by other programming languages, some of them functionally comparable (Nerlove, 2004).

Machine and Assembly language are mutually distinguished by the characteristic that the first is entirely numeric whereas the second incorporates mnemonics, making it more intuitive to write, but both require programming to be done using the elementary instructions particular to a given machine. In contrast, the creation of higher level programming languages not only permitted, to varying degrees, the transfer of finished programs from one brand of machine to another, usually requiring some modification in the process, but also the ability to instruct the machine in a "language" closer to, albeit still quite different from, the natural human language of the programmer. Fortran in particular replaced the need to specify mathematical operations arithmetically by the ability to use familiar algebraic forms and conventions, to at least some degree. It also provided a notation quite intuitive to engineers and other people with similar mathematical training. One of the constituent innovations was the use of a "compiler." The particular reference to "translation" in the name Fortran refers to the use of an automated translator program, a "compiler," which converts the human written Fortran "source code" statements into machine language, and which is then followed by the use of a "linker" to resolve external references and otherwise bind, or "link," together the procedures and other parts of the program for execution. Although logically more complex, involving machine interpretation, translation, and linking of the compiled source code written by people, the consequence was a substantial simplification of the human programming process, if also a reduction in the individual programmer's ability to control exactly how the machine executes the programmed tasks.

The primary intention of this particular description is to convey in what ways programming is in principle both conceptually straightforward and from a certain perspective rather basic, viewed as a matter of organizing tasks by selecting from a fixed and restricted set of instructions. With the advent of higher-level languages, programming obviously also became less tedious, faster and much easier to do. However, when this activity is explained in the way it has just been – with the focus upon the historical transition from human to machine computation, so as to emphasize the parallels that exist between the formulaic execution of a series of instructions by an electronic machine and the same calculations when manually performed by a person – the effect is to abstract from the deeper design issues that are associated with the choice of which tasks to perform, as well as the particular benefits of performing them in a specific way. Such abstraction can also have the effect of making the modern software development process seem much less vital and creative than it is, as can also any simple extrapolation of the past environment and practices into the future.

The past has provided a formative learning experience in several important respects, in terms of econometric practices and theory, as well as econometric computation. During the period from the late 1950s through the early 1970s, econometrics emerged in a specific way, its self-discovery shaped and channeled by the particular historical circumstances of that time and the antecedent causes of these, as well as the unreflective youth of this sub discipline then. In those days, econometricians focused a substantial amount of their attention upon the types and properties of parameter estimators. Likewise, the emphasis in the published descriptions of econometric software naturally tended to be upon these estimators (Belsley, 1974; Eisner & Pindyck, 1973; Hendry & Srba, 1980; Pesaran & Slater, 1980). Furthermore, the stress was almost always placed upon their mathematical representations, rather than upon their numerical analytic characteristics and the evaluative aspects of their use. Perhaps as a consequence of this history – which has been for the most part unexamined in the literature, and occurred rather unselfconsciously at the time – many economists today appear to regard econometric software development as consisting of no more than the act of stringing together a sequence of source code statements in order to copy and make operative algebraic formulae easily extracted from the published econometrics literature. This perception seems to have given rise to a perceptible tendency for many econometric software users to regard its creation to be simply an act of translation, or perhaps the process of fitting together according to blueprint a prefabricated set of building blocks.

There have also been certain recognized disincentives for econometricians to focus attention on software development any more than minimally necessary, or to examine carefully the interrelationships between the logical aspects of the software creation process and the development of econometrics. David Hendry, for instance, has suggested that "empirical research and programming both require disproportionately large time inputs relative to their professional evaluation, have very long gestation lags,...and [the findings] are difficult to publish. Thus [these pursuits] cannot be recommended as ways to establish or develop a career... Developing new technology (estimators, tests, distributions, etc.), however, is relatively straightforward, fairly rapid for the most part, and easy to publish when the result is 'new' and 'correct"' (Hendry, 1993, p. 115). Incentives and disincentives are the very stuff of economics. To take an interest in programming and econometric software development would seem therefore to be the graveyard of any academic economist's professional ambitions, justifiable only as a minimally diverting hobby, spoken of only to trusted colleagues.

These ideas have here been broached quite tentatively. The several circumstances just recounted, and inferences that initially have been drawn from them, will need to be evaluated more carefully before any final conclusions can reached, but taking them for the moment at face value, it would appear to be nothing short of rash, and certainly likely to be considered misguided, to suggest any sort of a central disciplinary role for econometric software development in the future. However, such a role may not be unthinkable. It can be argued that the prevailing impressions of this activity constitute a serious misunderstanding of the essential nature of econometric computation, both actually and potentially, in its relation to both econometric theory and practice. The interpretation of software creation as being necessarily parasitically derivative ignores, for instance, not only the possible complexities associated with turning a mathematical-statistical problem into an operative algebraic-arithmetic solution, but also the numerous large and small design choices that are involved in this process and that, in the end, may have an important effect on the degree to which econometrics develops successfully in the future.

For the theorist, a particular econometric problem can appear to have only certain, often seemingly axiomatically determined mathematical and statistical aspects, but such a perspective necessarily abstracts from the larger question of the real world context that defines the actual problem and its nature. Whatever apparently definitive theoretical result is obtained can be only an incomplete and partial solution operationally, pertinent and valid only under very particular circumstances that may or may not ever be empirically realized. As a software developer, the econometrician who incorporates new theoretical econometric results may therefore be faced with the often-difficult task of not only evaluating the relevance, hence the operational validity of the theoretical solution, but also implementing these results in a way that is contextually meaningful. This operationally focused econometrician consequently not only needs to understand the theoretical advances that are made but also to exercise independent judgment in the numerical implementation of new techniques, for in fact neither are blueprints provided nor are the building blocks prefabricated.

Nor can econometric software development, properly pursued, be characterized as passively receptive, evaluative only ex post. Instead, it should be regarded as being an inherently creative activity, potentially co-equal and mutually complementary in content and purpose with theoretical econometrics, in a way that helps to establish the areas in which theoretical advances are required. Of course, a self-serving seemingly fulsome declaration of this type is always easy to make; what needs to be addressed immediately is why it might be true. One of the reasons to take it seriously is the changing nature of econometric computation and its consequent growing importance to the ongoing development of economics and econometrics, especially since the early 1980s. It has not always been true that econometric software development has necessarily been creative in the above sense. The first programs, created by economists in the 1950s and 1960s during the early days of computing, often – but by no means always – both were limited in function and implemented calculations that intentionally mimicked formulae found in econometric textbooks. Even today programs can be written in the same way. However, increasingly as the years have gone by, it has become much more common for econometric software packages to be developed as progressively more capable tools intended to permit a broadly defined variety of sophisticated operations to be performed. Although in the beginning, as will be described, the focus was often simply upon learning to use the computer, both as a alternative to and an extension of the desktop calculator and other manual methods previously employed, as the years have passed the implications of this computer use have become better understood. As a consequence, programs have increasingly exhibited more and more of a conscious design element both inside and out. In modern times, one aspect of the econometric software development process is that it fundamentally involves both the designer's choice of which econometric facilities to include or leave out and, properly done, also the need to consider carefully the numerical analytic characteristics of the specific algorithms implemented. Furthermore, not only have econometricians become increasingly aware of the numerical subtleties that may be involved, starting in about 1967 (Longley, 1967), but in addition the types of calculations performed have become more numerically analytically challenging (McCullough & Renfro, 1998, 2000; Stokes, 2005) and may in the future become even more so.

An additional, very significant aspect of the modern computer experience is that specialization among economists has increasingly occurred: there is now a noticeably sharp separation between the development and the use of econometric software packages. In principle, any economist might aspire to program a computer and then to self-create a software package but, in practice, programs today are developed by the few for the many. One of the consequences of this specialization has been to introduce an element of user dependence, now grown to a sufficiently great degree that for many economists whichever set of econometric operations can be performed by their choice of program or programs has for them in effect become the universal set. It has become more common for economists to specialize their analyses to a particular program, rather than to choose among programs at the beginning of each project based upon knowledge of the particular specialty characteristics of each.

When stated baldly in this fashion, such assertions smack of exaggeration. However, it is not necessary to debate whether or not the typical economist today has already become computationally dependent upon the programs and facilities that are presently familiar. What is significant is that it is possible to imagine a time and circumstance that the computer on a person's desk or in the palm of his or her hand will at that time define the effective choice set, at least for particular tasks, shaping and controlling the way they are performed. Even short of this event, it is also possible to imagine both a time and circumstance that the programs and facilities made immediately available to the person will significantly influence and limit the choices made. Anyone who takes the time and trouble to evaluate how people employ any of the analytical programs available to them is likely to agree that a degree of dependence inevitably occurs. For example, fully 20 years ago Kenneth Berk (Berk, 1987) suggested that although statistical "software should [have] the features to do readily what needs to be done," that in fact there is a "tendency for the user to do what is readily available in the software," and consequently that "...packages have enormous influence over... analysis, especially over [that of] the less sophisticated users." More recently, Stephen Hall (Hall, 2003; Renfro, 2004b, c, p. 71) expressed the view that, as a determinant of econometric practice, "software plays a much greater role than either we realize or than it really should... The basic point is that econometric practice is defined by what most econometricians [are easily able to] do." A complementary consideration is that espoused by David Hendry (Hendry, 1993, p. 314), when he asserted that "no matter how powerful and general econometric theory may become, it is only of operational value after it has been implemented in computer software."

Another important consideration is the particular way in which software, once created, tends to evolve in its life cycle. It is a commonplace that widely adopted programs soon begin to exhibit even a glut of features, possibly reflecting both the desire of existing users to capitalize on their knowledge investment and the incentives for program developers and vendors not only to attempt to retain these users but also to profit by selling upgrades and enhancements. However, what is presently most significant about these tendencies is not the particular whys and wherefores, but rather that one of the consequences has been for programs to evolve into progressively more comprehensively applicable tools. To some degree this evolution has taken the form of the development of interrelated program suites, such as Microsoft Office. But irrespective of the particular manifestation, this comprehensiveness is also the effect of technological developments such as the Internet, as well as the metamorphosis of computers from large stationary objects into small, easily transported, and widely used devices. It takes little imagination to visualize a future scenario in which the various aspects of the combined process of both acquiring and analyzing economic, financial, and societal data will increasingly become, from the perspective of the user, completely internal to the computer, albeit via a broadband, possibly wireless Internet connection.

This progressively greater comprehensiveness of features, to include broadband wide area data capture, might well ameliorate some of the effects of the dependence mentioned earlier. However, it is important to recognize that a characteristic of a computer program is that the results it displays almost never permit its users to infer from external observation the precise characteristics of the internal operations. Notwithstanding the limited set of machine instructions available, their permutations are sufficiently great, particularly as the absolute number of instructions used increases, that it is generally impossible, when confronted by a program that performs complex operations, to make reliable inferences about the characteristics of the program's operation simply from the output observed. It is of course always possible that programming errors could have been made, but although an important consideration, this is not the most critical issue. Even in the absence of these, there are very few, if any, instances that the set of algorithms employed are unique, jointly or severally. The old adage that there are nine ways to skin a cat generally applies also to almost any computer program, especially one that is comprehensive in terms of the operations it performs. Except in trivial cases, it is a complete misunderstanding of the nature of a modern econometric software package to think that for each operation there is always a unique, and necessarily best, algorithm – interpreting this term here not in the narrow sense of code to perform a specific well-defined task, but rather in the broader sense that includes also the choice of which to perform. Furthermore, as will be discussed later, it also may not be possible to extrapolate from this qualitative evaluation, if successful, to the way in which the program will behave in other contexts, even when that evaluation is made using independent benchmarks.

The implications of the several circumstances and tendencies that have just been considered have not so far attracted sufficient attention, especially as regards their possible impact on econometrics, both theory and practice. For instance, it is still customary for both econometric textbooks and articles in econometric journals to be written as if the software facilities available to the reader exactly represent the material discussed or, in the absence of this, that he or she is almost certain to make any necessary calculations and choices independently. In fact, this monograph stands in testimony to the degree to which such representation does *not* occur, particularly in the case of the existing widely used econometric software packages. Furthermore, given the existence of these packages, self creation of computational facilities by the typical economist is no more likely to happen than a modern traveler bound for San Francisco who finds himself in New York is likely to choose to walk rather than to fly, or at least take the train or drive. As a pedagogic exercise, the modern student might occasionally perform certain simple calculations by hand or possibly learn to program to some degree, but he or she cannot be expected, as a rule, to create a new econometric package to replace or improve upon the existing ones.

Nor will any but a few trained econometricians attempt this task, for in a number of respects it is burdened with difficulty. Among the constraints, only a small proportion of the source code that comprises an econometric software package pertains directly to econometric calculations. Much of the code instead relates to the program's human interface or to the general management of numerical data. Simply to understand the range of operations and ancillary calculations that need to be performed can be daunting. The process of self-creating a new software package therefore involves a significant expenditure of time and effort, not to mention the need to develop particular skills, investments that are likely to be seen as disproportionate to any benefit the typical econometrician is likely to receive. As an indication of what might be involved, it is revealing to ponder David Hendry's comments on his experience not of creating a program from scratch, but simply converting an existing one, previously developed principally by him, from one context to another. He writes (Hendry, 1993, p. 314) that "had [he] correctly appraised the effort that would be involved, [he] would not have started," numbering among the problems the difficulties posed for the "part-time programmer" to meet the interface and documentation standards set by mass market packages. Just the amount of documentation required can be significant, whether or not professional standards are enforced. Even to use and maintain a self-created program effectively over time can require a substantial amount of documentation, which is therefore part of the "cost" of program development and maintenance.

Of course, for the econometrician, several high level econometric programming environments exist (Renfro, 2004b, c), including both Gauss and Ox, the existence of which to a degree obviates the need for the serious student or the professional econometrician to start from scratch. Each of these, as well as other readily available high level statistical-mathematical programming environments such as Mathematica, Matlab, R, and S-plus (Amman & Kendrick, 1999; Belsley, 1999; Herbert, 2004; Zivot & Wang, 2002), offer the potential to self-program specific econometric calculations in a way that minimizes the need to cope with the human interface, the computer's operating system, and other econometrically extraneous tasks. It is nevertheless true that these facilities are unlikely to be of immediate interest to the typical student or even the typical applied economist, in a world in which a number of other, more pre-formed econometric software packages already exist. Furthermore, there is a fundamental difference between a single econometrician using the computer for his or her own work and the creation of software intended to be used by economists generally. The report by an individual economist of a given result does not usually provide economists everywhere the capability to evaluate closely or replicate that result, especially given the minimal amount of detail that is customarily provided in the economics literature, even when the results are reported in a manner that potentially could allow examination of the way in which they have been generated. The idiosyncratic way in which the details are usually promulgated itself creates a barrier against their careful consideration by others, especially given the environment in which applied economic research is conducted.

Of course, there may also be a more fundamental need for economists to adopt a more scientific approach to both the presentation and evaluation of research findings, involving a profound change in the conventions of modern economic and econometric research. The pertinent issues are certainly not limited to the rather narrow question whether or not any given economist or econometrician has the time, interest, skills, and experience to create new software, either for local or more widespread use. To read what has just been said with only this limited interpretation is to misunderstand the essential nature of the argument made. Equally, the inference to be drawn is not that econometric textbooks should no longer attempt to instruct their readers about the calculations involved. Neither should it be that the modern applied economist, or even the well-trained econometrician, standing alone, is confronted with a difficult computational problem, although this may be true. Instead, as will be discussed, the appropriate inference to draw much more broadly is that economists and econometricians each need to recognize, and in their work explicitly take account of, the particular characteristics, aspects, and impact of the computational developments of the past 60 years. Furthermore, it will be argued, as suggested earlier, that the time has come to treat these computational realities as integral to the ongoing development of econometrics. It is possible to argue this position as a particular necessity that follows from certain negative findings concerning the research practices of economists (Anderson, Greene, McCullough, & Vinod, 2007; Dewald, Thursby, & Anderson, 1986; McCullough, Mc Geary, & Harrison, 2006; McCullough & Vinod, 1999, 2003). Alternatively, the argument can be based upon the role that the development of econometric software now plays and will play in the future under the best circumstances, for unless one wishes to argue that the purpose of econometric theory is merely to amuse and divert the theoretical econometrician, it is difficult to contradict the Hendry dictum quoted earlier about the circumstances under which that theory becomes operable in the modern world.

So far the stress has been placed on the division that has occurred between those econometricians who have, for one reason or another, chosen to develop econometric software packages and the dominating majority of economists, who either use these packages or possibly abstain from the use of the computer. This division is demonstrable, but it is not yet necessary to consider further exactly why it might persistently exist. Nor is it important per se that the work of a minority might be somewhat generally under appreciated, short of the possibility of market failure. However, what is immediately of concern is the potential for a significant effect on economics and econometrics of future econometric software development: for example, the consequence either might be that the practice of econometrics could be measurably affected or else that the existing or future software packages might provide results that differ in some significant way from those implied by the theoretical literature. Either or both these outcomes could be realized independently. To begin to examine the specific questions that they each pose, it is relevant to start by considering the circumstances of current and future econometric practice, both theoretical and applied.

The Foundations of Computational Econometrics

As indicated, until very recently, there have been few attempts by economists to consider the joint implications of the ongoing development of econometric theory, on the one hand, and econometric software, on the other. However, at least some of the underlying computability issues have previously been examined, and in a pertinent context, for the nature of machine computation itself has been addressed as a metamathematical question. Furthermore, certain fundamental results pertaining to this form of computation can be seen to have an historical connection to the attempt by Hilbert and others to demonstrate the completeness and consistency of mathematical reasoning. Although the degree to which all this is immediately econometrically pertinent may initially seem questionable, there are aspects that it is important to consider and that ultimately will be seen to be quite pertinent to the future development of econometric theory and practice.

In the early twentieth century, David Hilbert championed the power of a carefully developed axiomatic approach as the starting point of the development of any scientific discipline. In principle, theory would provide an intuition free basis to examine the connections between the axioms and the logical rules. A potential problem, however, is that axioms, even if they accord with possible states of the world – observed facts – might be contradicted by a hitherto unobserved fact. Hence, not only is there an obvious requirement that these axioms be both independent and mutually consistent, as well as according with the observed facts, but it is also necessary, for completeness, to rule out the possible existence of any, as yet, unobserved facts. As a mathematician, Hilbert was of course familiar with Euclidian geometry and the degree to which its axioms correspond to the almost self-evident characteristics of physical space. Furthermore, he had already shown that, by restating geometric axioms algebraicly, it is possible to demonstrate their consistency on the grounds that if algebra is consistent, then so are the Euclidian axioms. However, for a variety of reasons (Nagel, Newman, & Hofstadter, 2001, Chaps. 2–3), in order to deal with mathematical reasoning as a whole, Hilbert concluded that it was ultimately necessary to construct and rely upon absolute proofs, beginning from a complete formalization of a deductive system, with the aim of revealing the pure logical structure while dispensing with potentially misleading aspects. The plan, or "program" as it was called, explicitly required all mathematical theories to be based upon a finite and complete set of consistent axioms. The envisioned strategy was to permit the proof of the consistency of more complicated mathematical systems using progressively simpler systems, finally reducing the consistency of all mathematics to arithmetic, and by so doing provide mathematics with a secure, unassailable foundation.

As is done in the current *Wikipedia* descriptive entry, Hilbert's program can be stated in the form of five tenets:

- Completeness: all true mathematical statements must be formally provable.
- The Principle of Conservation: that, in a Platonic sense, proofs concerning real objects, even if originally demonstrated in terms of ideal objects, should be capable of proof employing only real objects.
- Consistency: the formalism of mathematics should admit no contradictions.
- Decidability: the truth or falsity of any mathematical statement should be algorithmically decided.
- Formality: mathematical statements should be expressed formally, with the operators defined by precise rules.

It should be noticed that the use here of the adverbial form of the word "algorithm" constitutes modern usage. The idea of calculating the values of an ideal mathematical expression, such as an integral or differential equation, employing an organized sequence of steps is not new, even to the twentieth century. However, the formal study of the characteristics of this process appears to date no earlier than 1936 and, with the advent of the electronic computer, the question whether an algorithm can be established for *any* mathematical statement immediately became quite crucial.

At first sight, Hilbert's program per se cannot be said to have an immediately obvious bearing on the development of econometric theory or practice. However, there are close connections and these are possibly more direct than might be thought. During the past seventy or so years, explicit mathematical reasoning has of course become more and more adopted by economists, and although economists may not always be conscious of what are, for them, pre-adoption developments in mathematical thought, this is not to say that these have had no effect and do not still cast a shadow. A noticeable characteristic of the theoretical econometrics literature is that it abounds with ideal results. For instance, the effects of errors in data are today most often ignored in this literature, except when this phenomenon is explicitly considered in isolation as a particular thing in itself, notwithstanding that the existence of such errors is in practice endemic. Of course, it is not that any theorist is unaware of this circumstance; the worst that can be said is that he or she is a victim of method: the particular justification for proceeding in this manner is that, as a method, it permits precise and unambiguous results to be obtained. This method is of course also employed pedagogically: when teaching econometrics, it is a time-honored practice to begin with a particular set of assumptions, in the role of axioms, using these, together with logical rules, to deduce a particular result. Subsequently, the assumptions can then be progressively relaxed, usually one by one, in order to expose the specific effect of each of them. However, when considering the transition from econometric theory to practice, it is pertinent at some stage to ask whether, in practice, it is operationally sufficient to understand these specific effects *in isolation*? At some stage it is also pertinent to ask which, if any, of the sets of assumptions specified by econometric theorists represent axiomatically a possible state of the world? That is, when considering the junction between econometric theory and practice as a subject to be investigated, an often-ignored question needs to be raised, namely whether historically econometricians have adopted too narrow a methodological focus? Moreover, if the answer is "yes," it can then be asked whether, as a consequence, the theoretical results obtained should be interpreted to rest upon a foundation that is possibly neither logically consistent nor complete?

From the operationalist perspective, which is to say the perspective of the econometrician who designs and develops econometric software, questions such as these represent neither heretical thoughts nor a wayward desire to punch holes into the theoretical edifice, but instead must occur at some stage as an occupational necessity – at least if, and to the degree, that this person sets out with the explicit intention to provide a research tool that can be objectively evaluated as inferentially valid, which will be assumed. Given this assumption, this person's appropriate role can be interpreted to encompass necessarily the study of any and all aspects of econometric practice without restriction, including in particular all related aspects of econometric theory. It is arguably incumbent upon him or her to begin by asking questions about the degree to which econometric theory is both complete and consistent, in the metamathematical sense; that is, pertaining to the analysis of formal mathematical systems and the mathematical reasoning process. However, the term that will be used here is *metaeconometrics*.

Metaeconometrics can be considered to be the study of econometrics as a research process, and there are potentially a number of distinct aspects to this study. For example, questions could be raised about such things as the conventional interpretation of what is meant by $T -> \infty$ in the context of the consideration of estimator properties. In particular, in a given context, if a specific, theoretically conceived economic specification is considered, which is defined (arbitrarily) in terms of some observational frequency, such as annual, it is conventional to interpret the concept of time going to infinity as an aspect of the extension of chronological time, in fact as what amounts to be an infinite lengthening of the sample period. However, among its characteristics, this convention abstracts from and raises certain questions about the potential for structural change as the sample period is thereby conceptually extended. Notice that for this imagined event to have meaning effectively requires the presumption that all the circumstances of the observational process are time invariant for all time.

As an alternative, it is possible to consider instead the progressively finer subdivision of the original sample period, with perhaps an incidental extension of this sample period in the process (for reasons that will be explained), and to recognize that as a unit of time a "year" exists not only as a calendar year or other such standardized classification of a length of time, but can be considered to begin at any arbitrary point along the time continuum. Consequently, *within* the time spanned by the original, presumably multiyear, sample period there are of course a potential infinity of segments, each a year long in length, for which in principle observations could be conceived to exist, especially if what is considered to be measured is either stocks or (average) flow rates at each of the infinity of "year" time intervals (in fact, in truth, points) within the original, at least approximately *fixed* sample period. So long as the lags, if there be any, are still considered to be of a year's duration or multiples thereof – which is why some incidental extension of the fixed sample period might still need to be conceptually permitted – the specification itself can be conceived to retain its properties as $T -> \infty$ in a manner that implies a progressively greater density of observation points within the original sample period. Hence, rather than to consider the implications of a longer observation of a process that can potentially change over time, an alternative is to consider the more intensive observation of that process during a more or less fixed period of time, at least as a mental construct.

Of course, the essential conceptual reason to imagine $T -> \infty$ is to be able to consider the effect as the number of observations on the given economic process increases without limit. But is it actually any less fantastic to imagine the result to depend upon situational characteristics at the end of countable time, at the instant of maximum physical entropy, than it is to contemplate counterfactually the more frequent observation of variables within an essentially given fixed sample period of historically elapsed time? That is, a time segment during which it is arguably plausible to presume that the historically observable economic process has definite characteristics? Of course, it is true that these alternatives might be expected to have different implications for the properties of each estimator considered: for instance, at least in a time series context, they could in fact imply something different about the possibility of estimator consistency, all other things equal. Notice, in particular, when considering the specific aspects of each these $T \rightarrow \infty$ cases, that to evaluate the *precise* implications as a manifestation of $T \rightarrow \infty$ requires a meaningful concept of what is actually being measured in each case. Notwithstanding years of econometric theorizing, it is in reality not good enough – in the sense of being truly meaningful – simply to consider abstractly a sequence of observations on a set of variables, for in the abstract anything can be assumed. It is instead logically necessary to be able to examine meaningfully exactly what is being measured and the precise properties of those measurements.

Actually, this example is primarily intended to illustrate that "sample period," "observation interval," "observational rate," "sequence of observations on a given set of variables," and other such elements of econometric assumptions play an analogous functional role in econometric theory as do the geometrician's "points" and "lines," or "between" and "lies on," and so on, as the elements of the axioms in geometric and other mathematical proofs. This correspondence is hardly surprising, given the adoption by econometricians and statisticians of mathematical reasoning as their logical method. However, there may in fact be an important and relevant distinction between mathematical and econometric reasoning, inasmuch as mathematicians such as Hilbert (Ewald, 1996) have held that not only are such terms primitive in the sense of being formally undefined, but in addition that, properly, primitives should take their meaning from the axioms into which they enter.

Not all mathematicians have since agreed. Frank Ramsey, for one, who is well-known to economists because of his seminal *Economic Journal* paper investigating optimal savings rates (Ramsey, 1928), as well as other contributions on subjective probability and utility and optimal taxation (Samuelson, 1970), famously took exception (Ramsey, 1925) to this Hilbertian formalist position, which effectively (and intentionally) leads to mathematical propositions that have been stripped of meaning in order to focus attention purely upon the logical proof generating process. Hilbert essentially argued that the benefit of rigorous abstraction is that it focuses attention on an examination of the logical relationships between the axioms, and prevents the intuition from investing the primitives with meaning that can have the effect of obscuring these logical relationships.

Hilbert's program was a response to the mathematical environment of the late nineteenth and early twentieth century and can be seen as an effect of the formulation of antinomy paradoxes by, in particular, Cantor and Russell – the Russell paradox of course arising from a consideration of the Set and Class of all things not members of themselves. The antinomies each manifest themselves as an unsound argument with a false conclusion. Furthermore, this conclusion is commonly an evident logical impossibility. Antinomies characteristically are solved by resolving either an axiomatic weakness or an inferential fault, however as Penelope Maddy has pointed out, among its effects, the "axiomatization of set theory has led to the consideration of axiom candidates that no one finds obvious, not even their staunchest supporters." She goes on to suggest (Maddy, 1988, p. 481) that, "in such cases. . .the methodology has more in common with the natural scientist's hypothesis formation and testing than the caricature of the mathematician writing down a few obvious truths and proceeding to draw logical consequences" and, furthermore, that "...the central problem in the philosophy of natural science is when and why the sorts of facts scientists cite as evidence really are evidence."

One of the reasons to examine aspects of the Hilbert program here is in order to bring out certain aspects of econometric reasoning. It is of course possible to consider econometric theory in the context of mathematical reasoning without an explicit reference to probability and the statistical calculus, in large part because of the measure theory formalization of probability by Kolmogorov and Cox, among others, a formalization that itself involves the representation of probability concepts in such a way that they both can be considered apart from their meaning and can be manipulated according to the rules of logic and mathematics. Mathematical logic and reasoning can therefore be seen to absorb abstract statistical representations, but does it then directly follow that these representations, mathematical or statistical, can be applied immediately to econometric computations and actual economic measurements?

Certain of the historical aspects of the development of the Hilbert program have been mentioned in passing. A sufficient reason to have mentioned this program at all is to avoid an otherwise elaborate first principles discussion, but it is actually no more necessary to consider this program in its historical context than it is to consider Keynesian economics only in the setting of the decade in which the *General Theory* was published. However, another particular reason to explore here certain aspects of early twentieth century mathematical thinking is that this time represents a transitional period. Before then, even pure mathematics was often conceived as involving an argument based upon "true," perhaps even self-evident axioms, that led, through the proper use of logical rules, to a "true" conclusion, whether or not this conclusion was a priori obvious (Maddy, 1988). Since then, it has become more natural to consider arguments in which neither the axioms nor the conclusions are necessarily based upon real world evidence. However, whenever this is done, there is always some danger that the axiom primitives will become so nominalist in character, so divorced from any actual reference to a corresponding real world entity or practice, that the only meaning they have is derived from the axioms in which they are found. Commonly in this context, their empirical characteristics are otherwise entirely undefined, so much so that it becomes impossible to match them exactly with observable entities. Furthermore, it is often easy in such circumstances to become insensitive to the fact that this transition has been made, unless careful attention continues to be paid to the precise characteristics of these primitives in the statement of the axioms.

It follows from such considerations that it is important to understand not only that there are methodological connections between the development of econometric theory, in terms of its logical reasoning and developments in certain other disciplines, but also that methodologies need to be fully contextually meaningful. It is furthermore important to recognize that, although it may be justifiable to develop a theory in an ideal context, any proper attempt to apply its findings empirically, necessarily requires a consideration of the observability characteristics of its axiomatic foundations. Therefore, in an operational context, econometricians cannot avoid asking such metaeconometric questions as: Does the historical inattention of theoretical econometricians to the observational meaning associated with primitives in econometric proofs imply the inability as yet to formalize econometric theory? If so, the result is the ambiguity that although the truth of its proofs are generally held to be independent of the truth or falsehood of the axioms, in fact they actually may depend crucially upon a particular conception of the economic future and the circumstances of the observational process. Indeed, they might depend upon a particular conception that is quite abstract and Platonically ideal in nature, rather than being in any way grounded in the real world of actual economic measurements. It might then be asked as a consequence, what permits econometric theory to be considered to be applicable in practice?

As a possibly pertinent example, consider the metaeconometric implications, when making a unit root test, of the non-rejection of the hypothesis in particular. This outcome can be, and often is, construed to represent an inferential finding concerning the economic process being modeled, specifically that shocks to the system are permanent in their effects. Such a test, originating in econometric theory as a (mis-)specification test, becomes in its practical application also a test of economic circumstance – assuming of course its proper formulation and possibly a number of other such caveats. However, the absolutely fundamental operational questions that then arise include: are the axioms and postulates of econometric theory sufficiently well founded that inferences of this type can be justified? Are the axioms themselves sufficiently well established *in this context*, both as real world relevant statements and in terms of the real world relevance of their primitives? Are the logical rules that are used to develop the postulates, *given these axioms*, appropriate, not simply in terms of the mathematical logic they incorporate (which might be presumed), but also in terms of their selection? Or is the econometric theorist, consciously or not, simply manipulating mathematical symbols in what is actually an entirely abstract, nominalist way, yet choosing to pretend a real world meaning? Consequently, are those operationally focused econometricians who, in turn, innocently if unthinkingly provide the applied economist with economic research tools and facilities in danger of being no more than hapless accomplices? The theoretical econometrician possibly may be able to avoid the need to answer, or even address such questions, by adopting a similar position to that of the pure mathematician, at least to the degree that he or she is willing to disavow anything other than purely coincidental correspondence between econometric theory and the real world (although it is not clear that this is actually sufficient salvation). However, the operationalist, the applied econometrician, has no such choice. He or she cannot avoid, at some stage, addressing the real world meaning of econometrics.

Clearly, there are two metaeconometric questions in particular for which answers would appear to be needed. First, there is the question whether econometric theory can be said to apply to the real, as opposed to the ideal world, in the Platonic sense. Second, there is the question of the possible logical contradictions, if any, contained in this body of knowledge, as well as its degree of completeness and reliability as a logical system. It is possible to consider these questions in a deeply philosophical sense, which is generally not to the taste of economists. Alternatively, it is possible to consider these questions simply on the level of the creation of an econometric software package that is said to embody econometric theory. Assuming that it is possible to develop the algorithms to realize the formulae that inhabit this theory in a numerically accurate way – an assumption that itself still needs to be considered carefully and will be later – there remains the issue whether this theoretical embodiment is likely to be sufficiently complete and self-consistent that the user of the package would in principle be able to make valid inferences?

In order to establish the roles of the possible participants in any investigation of such questions, it might be argued, as was done earlier, that the theoretical econometrician stands in relation to the operationalist as does the pure mathematician to the user of mathematics who employs it to investigate or make statements about the real world. The position of the theoretical econometrician might as a consequence possibly be characterized as one of fundamental unconcern whether the axioms and postulates are more than abstractly meaningful – although arguably the theorist must still address whether these conclusions logically follow from the assumptions made, as well as the logical consistency and completeness of econometric theory. The operationalist, in contrast, *must* be concerned, if not with the question whether the axioms assumed are actually true in each case, then at least with whether there exists a realizable state of the world in which they could be true. However, there are responsibility limits: the operationalist, insofar as he or she is an econometric software developer, has limited responsibility for how the software is applied; it is the economist who uses an econometric software package who in the end must consider the question of the truth of a particular set of assumptions, assuming of course that he or she can depend upon the mutual reliability of both econometric theory and the software package that is used.

At this point, it might be useful to stop briefly, and attempt to state clearly – and in a fairly straightforward and essentially summarizing manner – the particular way in which the foregoing should be interpreted. It is easiest to do this with the aid of a familiar textbook example, the Gauss-Markov demonstration of the properties of the Ordinary Least Squares estimator. As a theorem, this demonstration represents an abstraction. For instance, it says nothing about the characteristics of economic phenomena, their measurability, nor the particular way in which they might be successfully modeled. However, historically, the economist's inferences about economic phenomena have stimulated and shaped the development of econometric theory: given particular inferences that invest the individual axioms with an economic relevance (such as why particular variables could be jointly dependent), econometrics can be seen to have evolved in the hands of econometricians via a process of examining each of the axiomatic assumptions in turn. Successive variations on these assumptions have in time led to the development of the complete set of estimators that have been derived and studied by econometricians. However, it has been characteristic of the method of econometrics that, despite notable attempts to define classes of estimators and to consider an interesting, rather extensive variety of circumstances, no completely general theory statement has emerged that subsumes all, or even a substantial portion of econometric theory collectively. There is as yet no general theory of econometrics. It may even be true that this body of results is more properly to be conceived as necessarily made up of separable logical parts. Nothing said here should be interpreted as being critical of either this historical investigation or the results obtained, considered abstractly. It is only upon an attempt to apply these findings to real world phenomena that it becomes fundamentally necessary to begin to consider the completeness and consistency of this body of knowledge. Furthermore, it is only upon this attempt that it becomes fundamentally necessary to examine the circumstances under which this body of knowledge can be applied to measured real world phenomena, even if it can also be argued that it may be difficult to imagine a defensible justification for econometric theory in the absence of the ability to apply it.

It has of course been taken as given here that the modern tool used to apply the theoretically developed methods of econometrics is econometric software. The historical and present development of this tool raises a number of econometrically related, yet to a degree formally separate issues that have often simply been taken for granted. One of these is its reliability. The intrinsic reliability of any such tool can be seen to depend upon the fulfillment of at least two circumstances in particular. First, given possible states of the world, there obviously must be corresponding possible outcomes. These would appear to be econometric theory dependent; that is, there must be a correspondence between the axiomatic foundations of econometric theory, including primitive terms embedded in those axioms, and these possible states of the world. Second, there is clearly also a computability requirement: the concept, stated in terms of abstract econometric theory, must actually be computable, involving its successful operational translation into one or more corresponding computationally feasible, sufficiently numerically accurate algorithmic expressions, which are embodied in the software that is developed. Of course, a calculation may, in principle, be computable, without necessarily implying that in a given instance it will always be computed correctly or in a numerically accurate way – or in a way that will necessarily achieve the correct result; for instance, consider the difference between a local and a global maximum. Nonlinear problems in particular can pose significant computational problems, and there is also the issue of knowing after the fact that the actual solution has been achieved (Brooks, Burke, & Persand, 2001; McCullough & Renfro, 1998, 2000). Even ostensibly linear results are in fact, more often than sometimes thought, potentially computationally difficult (Stokes, 2004a, 2005).

In order to explore each of these topics rigorously requires more careful development and elaboration than this discussion has undertaken, but for the moment enough has been said by way of introduction to establish some sense of the relevant issues. Notice specifically that once one begins to think along these lines, it becomes almost immediately evident that the more explicit development of operational econometrics, as an approach, poses a challenge to econometric theory, which arises from the circumstance that questions both can and do occur in practice that have not previous been examined in the theoretical literature. Furthermore, from such deliberations it also becomes obvious that econometric practice is not only the test bed, so to speak, of the relevance of received econometric theory at each point in time, but also that operational econometrics must inevitably, over time, define the developmental program for econometric theory. Historically, there has been at least a tendency for econometric theory to develop without regard to the questions that, from an operational perspective, especially need to be answered. Recall from earlier Hendry's assertion that "Developing new technology (estimators, tests, distributions, etc.)... is relatively straightforward, fairly rapid for the most part, and easy to publish when the result is 'new' and 'correct.'"

Towards the Evaluation of Econometric Software

The discussion so far has simultaneously entertained two potentially conflicting ideas: on the one hand, that econometricians have almost always acted as if both econometric theory and software are applicable as a matter of course and, on the other, that it is not yet certain that either of these ideas are necessarily justified. To this end, the preceding chapter section has addressed certain of the issues that must stand behind any attempt to evaluate the existing econometric software as an operational representation of econometric theory. Its purpose was to begin to establish a general frame of reference in order to permit the productive consideration of econometric computation, as well as to determine the most relevant evaluative issues. Fundamentally, it has been argued that it is not justifiable simply to assume the applicability of econometric theory. There is a need to examine both the implications of the theory and the specific characteristics of the environment in which that theory is applied. This environmental context obviously includes the computational tools that might be used, in the form of the software available, in the past, now, and in the future. A contextual aspect that also needs to be recognized is that the adoption of the electronic computer as a replacement for earlier manual methods of calculation did not occur on a one for one basis. Because of its relative speed and the capability it offers to automate the computational process, the electronic computer has increasingly been called upon to perform tasks that would not have been attempted before its existence. Therefore, is there a point at which too much might be asked of this machine? Do the developing econometric technologies (estimators, tests, distributions, etc.) already threaten to exceed the ability to implement them computationally? Would it be immediately apparent if this were true?

Such questions cannot be ignored. But is it yet immediately obvious exactly what ideally defines "econometric software," enough to distinguish it from any other type and also to permit some type of qualitative assessment to be made of each potential candidate example? In earlier years – certainly in the 1960s and even into the 1970s and perhaps later – when economists referred to the software used for parameter estimation it was conventional to speak simply of a "regression" program, possibly implying the inference that the program's perceived dominant characteristic at that stage was its use of (statistical) regression to estimate parameter values. Notice that a characteristic of this inference might be that, then, any arbitrarily chosen "regression program" was viewed as offering the econometrician or applied economist all the necessary capabilities to practice econometrics. In effect, the technique and the method of performing it was seen to be one and the same. Does this conflation also reflect an implied lack of clear definition of the term "econometrics," at least as viewed from a modern perspective, and how long did this ambiguity last? In particular, when did the econometric tools become "econometric", rather than "statistical" or something else?

Unquestionably, at some point, the separateness of "econometrics" became generally recognized. It might be reasonable to choose 1950 as a defining date, although this recognition did not then instantly take the form of an illuminating flash for every economist or statistician. So too, with the passage of time, the idea that a distinct set of associated computational properties exists has gradually emerged, to the degree that the term "econometric software" is now easily found in the modern literature, but even today the specific distinguishing features may not be universally agreed. Therefore, a subject that needs to be addressed is the process of discovery of exactly what constitutes this category, in the form of an appropriate tool that might be specifically designed for the use of an economist. It is an obvious truism that the ideal tools of applied research do not simply appear like wildflowers, sown by a benevolent deity. Furthermore, whenever any tools appear, from whatever source, they do not always present themselves fully formed and instantly capable of being used effectively. For best results, active cultivation is necessary, as is also – ideally – constant attention and evaluation. There are at least two important issues: First, what is needed? Second, how closely does what exists correspond to this necessity?

The Discovery of Econometric Software

An exploratory software capabilities survey is an obvious first step in this process of discovery. Surveys of this general type first began to be made in the 1970s, ostensibly to determine the properties of either statistical software generally or, in a few cases, econometric software specifically (Francis, 1973, 1981; Francis, Heiberger, & Velleman, 1975; Muller & Wilkinson, 1976; Rycroft, 1989, 1993, 1999). Unfortunately, these attempts have historically raised as many questions as they have answered, especially in the case of the broader category, statistical software. One of the inherent difficulties has been both the variety of programs given this label and the diversity of their actual and potential functionality. Each of these circumstances poses problems for an orderly, perceptive assessment. Furthermore, whether as a direct consequence or not, the presentation of the survey results has itself often been problematic: the categorical functional classifications of program offerings have at times been too vaguely specified; for example, sometimes identifying simply whether or not each program performs some type of "multivariate analysis" (de la Vina, 1985) or other too broadly defined operation. At other times it has not been obvious why certain operations have been stressed and others ignored.

In general, the range of features and properties considered have commonly tended to be somewhat arbitrary and, in particular, only rarely have they included those, such as numerical accuracy, that would permit the qualitative evaluation of each program's operating characteristics (Brillet, 1989; Lilien, 2000; McCullough, 1998, 1999, 2004; McCullough & Vinod, 1999). An obvious aim of these surveys has been to define a group of programs that, on the basis of certain properties, fall into the category or possibly a sub-category of econometric or statistical software and then to consider the characteristics of each of the members of that group. However, the benefit of making such surveys has not always been clear, neither to software developers, who supply the information about their packages, nor to readers of the published results. Even if a particular classification of programs has been identified, the amount of associated disinformation can be substantial. Often, the consequence has been to emphasize novelty, or perhaps only the developer-declared range of each program's coverage (seldom independently audited), rather than the degree to which the software packages available at that time have jointly and severally provided the most appropriate facilities to support meaningful research.

In contrast, in order to begin to discover and evaluate the existing econometric software as a collective research facility, the approach adopted here has been to impose on the surveyed population a mildly restricted definition, the reason being to focus attention upon specific, possibly defining program features. The extent of this particular population was determined from a recently-taken census of econometric software (Renfro, 2004b, c). Essentially, the programs included in this census were those declared by their developers to be econometric in orientation and, in addition, to be capable of performing at least the basic misspecification tests and other operations usually associated with linear-in-parameters Ordinary Least Squares regression in an applied economic research setting. The focus of the present study is consequently narrow, with the consideration of additional techniques and capabilities left to the side, at least for the moment. However, during the initial candidate discovery stage these restrictions were only lightly imposed: for instance, in order to be selected for consideration, a particular program certainly did not need to perform this type of regression exclusively. Nor did the program need to offer it as a principal program feature. Inasmuch as each included program's relevant properties are revealed individually in a series of tables, the criterion for inclusion was not whether a given program did or did not exhibit any specific feature, beyond offering Ordinary Least Squares. Furthermore, as indicated earlier, even this criterion was established primarily because of the fundamental nature of this estimation technique, rather than as an exclusionary requirement. As might have been expected, there was no instance of a candidate program that performed any pertinent type of parameter estimation technique that did not perform Ordinary Least Squares.

From the start, the fundamental goal has been to discover what econometric software developers have *intentionally* done, both in response to a perceived existing demand and as conscious innovations. As indicated in the Preface and Introduction to this volume, this focus on Ordinary Least Squares represents only the initial stage of a larger inquiry. Within the declared category of econometric software packages, the goal is to examine the characteristics of the way in which such software has been developed over time as an aspect of the present characteristics. To that end, this survey's general purpose is to identify the degree to which each of the facilities these packages offer, individually and collectively, can be said to support adequately the research needs of applied economists. This is an aim that makes it prudent to start slowly and proceed deliberately, for this judgment evidently involves simultaneously the need to consider and evaluate the state of the art of econometrics. This broader evaluation is important, for as discussed earlier, software developers ply their trade within the evolving context of present-day disciplinary conventions and, at each point in time, a particular body of knowledge. What is of interest is both the ideas that have been embodied in the software and those that have not.

It is also relevant that the packages surveyed have in common the property of being made unrestrictedly publicly available (although not free of charge in most cases) and therefore, by implication, intended for wide general use, even if the expectation of each program's developer might be that each user will have at least some econometric training. One aspect of this situation is that a desirable characteristic of the design and development of each program is what might be called *operational completeness*, a phrase intended to express that, ideally, each possible circumstance of its operation should be taken into account in advance by its developer. In the case of a program created by one person (or a development team) for the use of another, that user is inevitably placed in a dependent position – dependent on the software (and the hardware) to perform each operation. Moreover, as will be discussed in the next chapter, this dependency exists irrespective of the degree to which that user is able to control the selection of the specific techniques employed. Dependency would also occur even if the software were to be supplied always on an open source basis, for only some economists have the knowledge and skill required to become intimately familiar with every program detail, not to mention both the interest and time that would be required.

Furthermore, short of intimate familiarity, it is not always apparent which of the various seemingly possible operations are computationally sensible, according to the rules of the program's logic, and which are not, even to an experienced user of this type of software. Nor is this situation specific to the electronic computer: for example, in the same sense, the driver of an automobile can suddenly find that its speed is too great for present road conditions, possibly because of some just-encountered environmental change. At each point in time, in both contexts, actions beget reactions, leading to an outcome that could have serious implications. What is more, in its consequences and interpretation, this outcome may be either difficult to foresee or later comprehend, or both. Actually, in the case of software, it is not always obvious either exactly when or that a "problem" has occurred. Certain of these may manifest themselves as a program "crash," or other obvious malfunction. Others may yield serious errors that are not necessarily evident to the user. Consider in contrast the rather more generous tolerances involved when reading a book: misspellings, awkward sentence structure, and other such sins ordinarily have a negligible effect on the information transmission process or the usability of the book.

However, speaking of books, the present one has several characteristics that specifically need to be taken into account. Fundamentally, its particular choice of subject matter reflects that its primary purpose is to assist with the ongoing design and development of econometric software packages. Therefore, as mentioned in the

Preface, its aim is not so much to provide either the general reader or user of econometric software with information about the individual programs surveyed, but rather to place the stress upon the research facilities they offer collectively. It is principally addressed to theoretical econometricians and those who develop econometric software in order to identify for this audience collectively which of the relevant econometric techniques have to date been incorporated into the existing packages. One of its functions, perhaps, is therefore also to help to set the agenda for a later general discussion of other techniques that are candidates for inclusion in these packages, in addition to, or even in lieu of at least certain of these. Furthermore, although the purpose of the underlying survey is not per se to assess the numerical accuracy of the each of the surveyed packages individually, the numeric values of the statistics displayed throughout the volume have been computed as intentional benchmarks, in a way that will be described. Therefore, after the fact, each package can be evaluated separately against these benchmark values – although certain qualifications still apply.

Recall from the discussion in the Preface that, in order to maximize the usefulness of these numbers, the findings were made available to the developers of the surveyed packages before publication – in an iterative fashion as the book was being written – so as to permit a reasonably detailed collective evaluation of the degree to which these programs all produce the same or comparable results. Out of this interactive, feedback evaluation has come much of the information presented later about the specific differences between packages, for as mentioned earlier one of the discoveries made early in this process is that the numbers displayed can differ between packages, even when the input data are the same and the calculations made are equally accurate. Usually, in each such instance, these differences reflect differences in developers' interpretations of the appropriate formulae to use.

Sometimes the differences are nominal, but, in other cases, substantive. Almost always, they reflect that individual developers have made deliberate choices, illustrating also an aspect of the way in which the development of econometric software affects econometric practice. Consider, for instance, Fig. 1.1, which provides an arresting example of differences between packages. Not all the surveyed packages are represented in this particular display. Its purpose is simply to show *nominally* identical values, for a given input data set, from each of a selected group of packages. What is shown in this figure row by row are the values of the Akaike Information and Schwarz criteria that each of the selected programs respectively generates, *given exactly the same observational inputs*.

The particular reasons for these results, and their characteristics, are discussed in Chap. 4, but it is obvious that they are not what might be expected a priori. The pertinent question is, do such differences matter? The numbers are certainly different, but in this instance should they be regarded as being substantively so, enough to affect the inferences that might be drawn? Is the possible effect on these inferences likely to be different for different users, depending upon the knowledge and experience of each individual? An interesting aspect of this case is that none of the programs represented here is guilty of any numerical inaccuracy, at least as

	Akaike	Schwarz
B34S	194.6275	198.6104
EViews	9.6314	9.7807
MODLER	6.7935	6.9429
Stata	192.6275	195.6147
TSM4	-96.3137	-97.8073
TSP	96.3137	97.8073

Fig. 1.1 Akaike and Schwarz test values

regards the numbers displayed in this figure. As a reflection of each developer's own intentions, each number shown is arguably correct and logically defensible.

Of course, econometric conventions and knowledge evolve. As a result, there has been a noticeable change in the content of econometric software packages during the past 30 years or so, towards providing not only a range of parameter estimation techniques, but also a much broader selection of evaluative statistics, reflecting the modern day emphasis on specification search. It is characteristic of this joint process of search and parameter estimation that it involves the generation of both parameter estimates and associated statistics. However, the aspect that is more emphasized in this volume is the particular misspecification test facilities provided, rather than the numeric parameter estimates. This emphasis is hardly surprising inasmuch as, in the case of each test statistic, the parameter estimation technique has been controlled for in the design of the underlying survey. Nevertheless there are certain evident broader implications of the results presented: in particular, although the volume's treatment stops short of attempting to consider as specific topics either the methodology of the use of software or the effect of the particular way in which the considered tests can be or are used by analysts (Hoover & Perez, 1999; Lovell, 1983), its coverage still makes it possible to consider not only *what* but also *why*. Both these aspects are important inasmuch as the characteristics of econometric software packages are not preordained, even if, to a degree, they are a direct consequence of the particular historical development of econometrics as a sub-discipline of economics.

Recall as well the earlier assertion – supported by the results presented in this study – that these programs and their evolving characteristics have increasingly shaped both that development and the general progress of economics as a discipline. Broadly considered, this symbiosis is not surprising: economics from its beginning has sought to explain the nature and causes of observed economic activity. Together the electronic computer and software now provide the means to organize and evaluate the data that economic activities generate. From this evaluation comes in turn the ongoing development of the discipline, which then affects the further development of its research tools, including econometric software. However, notice that there is actually a more precise dynamic at work than just this. As touched upon earlier, software development originally occurred self-creatively, to suit the personal needs, practices, and beliefs of the individual investigator. In those days a motivating aspect was commonly a previous absence of suitable software, making self-creation often a necessary first step (Renfro, 2004b, c). In contrast, today, a number of econometric software packages exist. These programs might or might not be exactly what is ideally needed, in each individual case, but jointly and severally they have been created in order to suit what their developers perceive to be the needs of economists generally, as well as their own.

Significantly, on the demand side, the existing packages now appear to be sufficiently acceptable to economists that collectively they are widely used. In addition, they seem to have coalesced into a relatively fixed and stable population over time. This change from self-fabrication to the habitual use of what is off-the-shelf can be ascribed to the wider availability of the computer, which has fostered specialization and the creation of a market. However, as discussed earlier, rather fundamentally this change also reflects the skills, interest, and time required to design and develop additional or replacement software. Arguably, the necessary skills and development time have both increased with the passing years, in part because of progressively greater user expectations. Apparently there is also a high degree of habit persistence on the part of users, with users commonly unwilling to convert from one program to another, once a particular program has been adopted, which has resulted in a perceptible inertial effect.

A danger inherent in these circumstances and frictions may be more of a tendency for the econometric theorist to become isolated from the economic practitioner. Obviously, to the degree that practitioners accept the software that is available, rather than writing it themselves, having carefully studied the theory, the effect is to impose an intermediary barrier between these two groups. Such intermediation can perversely affect the interchange of information about econometric techniques. Already, both textbooks and software manuals are perceptibly becoming focused on the description of the techniques actually available in the existing econometric software packages – rather than always on the issues of greatest present interest to theoreticians. Of course, the econometricians who develop the software and those who write the textbooks still read the theoretical literature, and some contribute to it, which can be seen to explain the characteristics of the current offerings, but it is interesting that, when asked, textbook writers such as William Greene, for example, will indicate that, when considering the contents of each new textbook edition, they examine the journals, but interact with applied economists directly and through conferences "to see what people are doing" (Greene, 2003).

As significant, from the perspective of the typical applied economist, it also seems to be true that the econometrics journals and handbook literature has become progressively more forbiddingly specialized, as well as more voluminously daunting. A measure of this effect can be seen in the fact that, as of January 2006, a single journal, the *Journal of Econometrics*, had published a total of 130 volumes since 1973, compared to the *American Economic Review's* 95 volumes in the years since 1911 or the *Economic Journal's* 115 since 1891. Because of differences in page count, this comparison is, to a degree, misleading, but it is nonetheless suggestive. What may be equally revealing is the ease with which it is possible to find instances in the theoretical literature that an author will describe as "popular" a particular technique or specific test that is not found in any of the existing econometric software packages, which suggests that its popularity exists, if at all, only in theory, or perhaps just among the relatively small group of computer savvy econometric theorists.

Of course, anyone approaches survey results with certain expectations and preconceptions. Stepping back from the survey, and then considering from this longer perspective the results presented in this volume, there are two possible polar inferences that could be drawn. The first is that the techniques that are currently included in the existing econometric software packages are exactly those that should be, and that these are comprehensively applicable, sufficiently easy to use, and constitute best practice. Alternatively, it is possible that these techniques do not, or in some other way fall short. Of course, a more nuanced conclusion might instead be reached by and by. However, the present survey is intended by design to be neutral in these respects. Its purpose is not to draw or force any particular conclusions, but rather simply to provide a statement of currently available offerings, which will then permit later evaluation by anyone who chooses to consider the evidence presented.

In addition, as indicated earlier, this volume also provides benchmark results. These are included for two distinct reasons. The first is that they provide both current and future developers of econometric software packages with a ready means of testing their own results, inasmuch as those shown are in all cases based upon published data that are readily available in both printed and machine-readable form. The second is that these results also permit developers, users, and evaluators of econometric software to determine if, in each case, the technique implemented by a specific package is standard or else particular to it. However, it is very important to appreciate that if a value, independently generated later using the same data set, is found not to match one or more of those shown here, the implication is not necessarily that an error has been made by that program's developer; the observed difference may indicate only the unremarked use of a variant technique. Certain examples of this common phenomenon are examined in the following chapters and, of course, one example has just been displayed.

The Computer as a Computational Medium

It is obvious that the electronic computer has become indispensable as a computational medium. However, it is still important to assess both its characteristics and adequacy in this role. The Hilbertian "program" to establish secure foundations for mathematics was considered briefly earlier. As a statement of requirements for logical consistency and completeness, this initiative is interesting as a touchstone for a consideration of certain of the econometric issues that have been discussed, even if it might be difficult to establish a sufficiently close correspondence to make it worth attempting to pursue this idea more formally. However, in any case the Hilbert program has a limited applicability, for in 1931 Kurt Gödel demonstrated that it is impossible to achieve, at least when stated in its full generality (Gödel, 1992). Gödel directly addressed the issue of decidability, or using Hilbert's terminology the *Entscheidungsproblem*, the "decision problem." In particular, Gödel's second incompleteness theorem states, in effect, that any realistic mathematical theory is unable to prove its own consistency (Giaquinto, 2004, pp. 165–230); or more particularly that "there are an endless number of true arithmetic statements which cannot be formally deduced from any given set of axioms by a closed set of rules of inference" (Nagel et al., 2001, p. 109). An implication is that arithmetic does not provide the means to prove its own consistency, and therefore cannot be used in the context of or as a foundation for a more general proof. Gödel's result of course does not affect the need to consider theoretic completeness nor axiomatic consistency; it simply establishes the necessity for a degree of uncertainty to be associated with this investigation.

On a more positive note, in 1936, as a variation on Gödel's proof, Alan Turing (Turing, 1936, 1937) demonstrated that, in principle, a machine can compute anything that can be calculated using "computable" numbers and, at the same time, that this machine cannot determine whether the potentially solvable problems can be solved, thus also showing that the Entscheidungsproblem cannot be solved, even by what Turing called a "universal computing machine"; that is, using an idealized general purpose computer. As a matter of definition, "computable" numbers are defined by Turing (Turing, 1936, p. 230) to be "the real numbers whose expressions as a decimal are calculable by finite means" using arithmetic operations. Turing characterized the process of computation as a sequence of these operations, in fact as capable of being represented by the movement of a tape, subdivided into "squares" one each to a number, that could move forwards and backwards or, in his telling, left and right and in the process be "scanned" by the machine. In the modern context, Turing's "tape" can be interpreted as representing computer memory, so that his paper establishes abstractly the mathematical foundation of a computational algorithm as a sequence of elementary operations. However, equally important as a practical matter, the finite representation of real numbers in electronic computers has the inevitable effect of restricting their numerical precision, the implication being that computer-based algorithms need to be considered as being distinct from idealized mathematical representations. What in addition distinguishes any actual computer from a universal computing machine is its finite total memory capacity.

Nevertheless, Turing's results can be viewed as providing a perspective against which to view the later development of the general purpose electronic computer, the first example of which was the ENIAC (Electronic Numerical Integrator and Computer), which became operational at the Moore School of the University of Pennsylvania in February 1946. It was created by a team led by Presper Eckert and John Mauchly. The US Army funded the construction of this machine, its particular original attraction for the Army in the early 1940s being its prospective ability to automate the tabulation of artillery firing tables; actually one of its earliest uses was in connection with the design of the hydrogen bomb. Work on the ENIAC began in 1943. However, before it became operative in 1946, a different computer was conceived by the same design team, namely the EDVAC (Electronic Discrete Variable Automatic Computer). The EDVAC is historically significant as the first stored program computer. It is therefore the direct ancestor of the modern computer, as well as being generally compatible, within real world restrictions, with Turing's idealized machine. At the time of the EDVAC's design, the team included John von Neumann as a consultant, who in 1945 produced the paper (Neumann, 1993/1945) that first set out the architectural principles for the EDVAC and essentially those of the modern stored program computer, an event that has since occasioned considerable debate concerning the origins of each of the ideas expressed. However, without question, von Neumann is the first person to create a computer program (Knuth, 1970).

This brief account risks leaving the impression that the computer suddenly emerged fully formed during a given 10 year period in one particular context. In order to provide a more rounded perspective, it is pertinent to consider also several other, generally prior, circumstances. Arguably, economics is a naturally quantitative discipline. The need to process considerable quantities of economic, financial and societal data was certainly originally felt by economists and others much earlier than the 1940s, as is suggested by the extent to which human computation became an organized, professional activity during the early twentieth century, if not before. Out of this need came the creation of what later became computer peripheral devices, among other effects. In particular, the practical problems associated with counting the robustly growing population of the United States, especially during the second half of the nineteenth century, provided the stimulus to data processing innovation on the part of Herman Hollerith.

The taking of a decennial census is mandated by Article 1 of the United States Constitution, in order to determine both the proportional representation in the US House of Representatives of each of the states and their proportionate shares for certain direct taxes that may be levied by the federal government. The first census took place in 1790, which was purely a population count, but by 1810 economic data began to be collected in the form of a few questions on manufacturing activities. Coverage of mining and a few commercial activities began with the 1840 census. In 1905 the first separate Census of Manufactures was taken and since that time a relatively comprehensive set of separate censuses have been instituted (albeit with a noticeable lag) that have since helped to provide the foundation of the US economic accounts, among other benefits. The amount of effort required in the earlier years simply to tabulate the results, essentially explains this pattern of the slow extension of the decennial census, although of course a certain degree of political infighting was also involved.

The 1890 census was the first time mechanical tabulating machines were employed, which made it possible to compile that census in 2.5 years, compared to 7 years for the previous one, notwithstanding further population growth. These machines, designed by Hollerith and based upon punched cards, featured the first automatic card feed mechanism and included the first keypunch machine, operated using a keyboard. Subsequently, Hollerith's Tabulating Machine Company merged with two others to become in 1924 what is now known as the International Business Machines Corporation. Some years later, in 1940, IBM underwrote the creation at Harvard University of the so called "Mark I" Automatic Sequence Controlled Calculator, not an electronic computer, but nevertheless a large scale digital calculating machine (Bloch, 1948). Among its characteristics, this electromechanical machine incorporated card readers for input and, for output, both a card punch and two electric typewriters. The later, "Mark II" version, also incorporated for input a paper tape reader. These input devices were then perceived to provide in addition a way to extend the machine's memory (Bloch, 1948).

Much hangs on the single word "electronic." The Mark I was not an electronic computer. Significantly, when tested against the Mark I, the ENIAC was found to be able to perform individual calculations as much as 900 times faster (Campbell-Kelly, & Williams, 1985; Wilkes, 1956). Because of this speed differential – which obviously would be even more pronounced today, were it to be tested against a modern computer – the Automatic Sequence Controlled Calculator, being in important respects a mechanical device, represents a dead end. It was an interesting application of early to mid-twentieth century electromechanical technology. However, it *was* an automatic computer, in the sense defined earlier. Moreover, as Maurice Wilkes has pointed out, it was the "first automatic machine ever to be completed" (Wilkes, 1956, p. 20). It also happens to be the first automatic computing machine to be used by an economist in his research.

The Computer as an Imperfect Computational Device

Turing's universal computing machine has the characteristic of offering infinite memory. In contrast, actual computers have finite memory, which inevitably implies a degree of representational imprecision. Only certain real numbers can be represented exactly but, in addition, there are limitations on the ability to create algorithms that are themselves precise representations of mathematical expressions. The inherent sources of computational error are usually identified as:

- approximation error
- roundoff error

Generally speaking, approximation error is due to inexact representations. In turn, roundoff error can be ascribed to calculations made using numbers the representations of which are limited in storage length to a finite number of digits. The software developer's challenge is to limit if not ameliorate the impact of these errors, taking into consideration that, in some cases, the effects will be amplified – under certain circumstances, even catastrophically so.

For various programming purposes, the memory of a computer can be logically subdivided into groupings of binary digits, commonly characterized as "nibbles," "bytes," and "words," the latter of which is machine dependent but will ordinarily be some integer multiple of each of the other two. A byte consists of 8 binary digits and a nibble 4. However, economic observations, as well as the intermediate and final results of econometric calculations, will ordinarily be stored as floating point numbers, taking the form of a fractional part plus an exponent. The standard default, which can be changed, is to represent such numbers using 32 binary digits as so-called single precision numbers. As an example, consider the number 3,345,647, the fractional part of which can be given as:

.334565

If it is represented to six places. Notice that here, as usual, the fractional part is expressed in normal form, so that only significant digits are represented, inasmuch as leading zeros would simply inhabit space to no purpose. In fact, the decimal point is itself also redundant in this context since position alone conveys the necessary information. Observe in addition that the rounding up convention has seemingly been used, but this is not the only possible representational choice. The exponent is obviously 10^7.

Of course, it is possible in principle to choose an exponent that, in effect, will map any real number into the set between -1 and $+1$. However, because of finite computer memory, the number of digits of precision available to represent the fractional part will necessarily be limited. This statement is true whether one considers single precision numbers, double precision numbers, or for that matter numbers of any finite storage length. Specifically, if f represents the number of binary digits available for the fractional part of each number stored, and e those available for the exponent, then these two numbers, taken together with the basis of the number representation (generally base $=$ 2 or base $=$ 10), will determine the (finite) set of *machine numbers*. These are the set of real numbers able to be represented exactly within a given machine. In particular, notice that these are base specific.

The approximation problem mentioned earlier involves error that is due to the inherent inability in particular cases to achieve an exact solution, even in the absence of rounding error, such as occurs when integrals and derivatives are approximated using discrete expressions or a partial, finite sum is used to approximate an infinite series. These instances should not be interpreted to include those that involve approximation or other error that can be assigned to ignorance of the appropriate functional form or errors made in the specification of an objective function, but only those instances that there is a need to approximate a *known* mathematical expression. This type of approximation error is sometimes spoken of as *truncation error* or *discretization error*.

During calculations, a fundamental computational problem can be seen to be how to best approximate any real number that is not a machine number using one of the numbers that is. It is important to realize also that, for any given machine numbers x and y, the results of each of the elementary arithmetic operations $x \pm y$, $x\,y$, x/y need not be machine numbers, leading among other things to a breakdown in the associative and distributive laws of arithmetic (Knuth, 1998, pp. 229–230). However, the particular error implications of each of these operations are not immediately intuitive. It is conventional to consider the implications (for any number $u \neq 0$ and its represented value $\hat{u}$) in terms of the relative error associated with this value:

$$\varepsilon_u = (\hat{u} - u)/u$$

In the case of both multiplication and division (and even in the case of square roots) relative errors in the individual operands do not propagate strongly, although certain operations are restricted by the computer's standard treatment of numeric under and overflows as calculation "irregularities" (in particular, in the case of division, it is necessary that $y \neq 0$; in fact, y must be different from zero by at least a certain, machine dependent, small amount). In contrast, in the case of subtraction and addition (when the signs of the operands differ), the relative errors of at least one of the operands will be amplified, possibly drastically. In the extreme case that $x = y$, subtraction will lead to cancellation, one of the serious effects being the propagation of errors made in earlier calculations of x and y, before the subtraction (or, what is the same thing, the addition of numbers of opposite sign). Notice also that if, before the subtraction, x and y agree in one or more of their most significant digits, partial cancellation will occur, itself resulting in the amplification of propagated error.

It is obvious that the various approximation errors, including those that occur because of the approximation of real numbers using machine numbers, will propagate during computations. Furthermore, mathematically equivalent expressions, such as $(a + b) + c$ versus $a + (b + c)$, can lead to different results as a result of the need to use floating-point arithmetic. As a consequence, the fact that two different algorithms might appear to be mathematically equivalent is not necessarily a consolation. An important consideration is the degree to which one of these is numerically more *trustworthy*. However, as a matter of usage among numerical analysts, trustworthyness as a property simply represents a comparative judgment; one of two algorithms may be more trustworthy, but still not be numerically stable. Of course, in some cases, it may be possible to develop an algorithm, considered on its own, that is numerically stable, in the sense of involving roundoff error or propagation error that is limited within certain bounds. Such an algorithm is said to be *well behaved* or benign.

In addition, it is also necessary to recognize that certain circumstances exist in which a chosen algorithm will become significantly more error sensitive. In particular, consider as an example the two equation system (Macon, 1963, p. 65):

$$\begin{aligned} x_1 + 10x_2 &= 11 \\ 10x_1 + 101x_2 &= 111 \end{aligned}$$

versus the alternative:

$$\begin{aligned} x_1 + 10x_2 &= 11 \\ 10.1x_1 + 100x_2 &= 111 \end{aligned}$$

In the first case, the solution is $x_1 = 1$ and $x_2 = 1$ whereas, in the second, $x_1 = 10$ and $x_2 = 0.1$. Obviously, the effect of the small difference in the coefficient values in the second equation is to cause a substantial difference in the computed solution values. This example is sufficiently simple that the consequences are readily apparent. However, the potential problem of ill conditioned, near-singular matrices is of quite general concern precisely because of the implication that small relative input errors (or errors that occur in intermediate calculations) can give rise to large relative errors in the final computed results.

Stated in this way, the problem of ill conditioning is cast as one that lurks in the computational underbrush, ready to spring out at any moment. As will be considered further in the next chapter, the inference that might be drawn normatively is that numerically stable algorithms should always be used, just in case. In the particular context of OLS parameter estimation, bearing in mind that matrix inversion is rather fundamental to the process, it is hard to fault this recommendation. But, of course, there is also the implied corollary that in the absence of ill conditioning, result errors will not be disproportionate to input, approximation, or intermediate errors, so that the likelihood of ill conditioning can be a valid design consideration, particularly to the degree – as was true years ago – that a numerically stable algorithm could be significantly more computer resource intensive than another.

The problem of ill conditioning might be regarded as an aspect of the general question whether to view the various possible errors associated either with the input data or roundoff as *the* fundamental cause of computational problems? Alternatively, it might be possible to interpret ill conditioning to be an inherent property of the computational problem at hand, arising from the characteristics of the process represented, rather than as directly *due* to errors in the input data or subsequent roundoff or other error propagation problems. For instance, in the case of parameter estimation in the presence of multicollinarity, such a circumstance might be either an inherent characteristic of the economic process being modeled or else present only in the particular sample of data that is employed. If the latter, the essential problem might then be regarded as being simply the consequence of a "bad" data sample. If the former, errors in data or even roundoff error might be interpreted to play essentially a catalytic role: if there is ill conditioning, but if errors in data and other errors were not present (at least in part due to infinite memory capacity), then the fact that a particular matrix might be near singular would be simply a circumstance. Of course, in practice it is impossible (or at least foolhardy) to ignore that if there is the slightest amount of roundoff or other error an arbitrarily selected computational algorithm could then experience a catastrophic error – an event that may or may not be obvious to the analyst. For instance, considering the alternative simple systems above, which of the two dramatically different solutions is the "right" one?

The several numerical analytic issues just discussed are each considered in much greater detail and variety in any of a number of texts (Golub & Van Loan, 1996; Householder, 1974, 2006; Ralston & Rabinowitz, 2001; Wilkinson, 1963); for a more detailed discussion of floating point arithmetic, see, for example, Knuth (Knuth, 1998, p. 214ff) and references given there. It is nevertheless important to have reviewed at this stage certain of the potential problems in order to convey that, although the electronic computer is an imperfect computational device, if properly utilized it still permits calculations to be performed sufficiently accurately to serve the needs of economists and econometricians. It is possible to be frightened by the potential for catastrophic error and useful to stress the dangers. However, the consequent requirement to design and develop numerically stable algorithmic representations, even if these necessarily do not have the properties of ideal mathematical expressions, imposes a salutary realism.

Errors in Data as a Computational Circumstance

However, let us not slip away too quickly from a forthright consideration of the empirical measurements that are made, in the form of observations on economic phenomena. Measurement errors, as "input errors," were mentioned in the last chapter section, but little emphasis was placed upon their role as a source of computational error, except insofar as these represent a type of approximation error that is propagated in roundoff error. At the point that measurement errors too are explicitly blended into the mix, it immediately becomes evident that there are possibly much greater conceptual complexities to be wrestled with than might be inferred from the seemingly simple term "observations." Notwithstanding the title of Morgenstern's well-known book (Morgenstern, 1960) and the continuing customary usage of this term, many of the economic measurements that are symbolically represented in the literature cannot honestly be assumed to take the relatively benign form of simple observations with minimal white noise errors. Especially in the case of macroeconomic measurements, these are often constructions, fabricated in various ways on the basis of samples, or occasionally censuses. From time to time, they even take the form of an outright patische, stitched together, using benchmark extrapolations and more doubtful methodologies, just barely garnished with a few solid facts. Only sometimes are such observations primary measurements, observable in the strict sense of the word, or the aggregates of primary measurements, and even at best these are still potentially contaminated with all the inaccuracies and defects against which Morgenstern warned. Of course, Morgenstern was well aware of the constructive nature of some economic "observations," but he wrote during a relatively early time and perhaps as a consequence more emphatically stressed observational accuracy, rather than specific aspects of manufactured economic statistics in a way that might be done today based upon the experience of the past nearly 60 years.

It is not difficult to become increasingly pessimistic the longer one stays in Morgenstern's shadow, for the process of measurement has some claim to be regarded as the black hole of econometrics. On the one hand, it exerts a strong pull on the light of the econometrician's theoretical advances. On the other, it has no separate theoretical role to play and the properties of measurement errors are often difficult to empirically assess. Mogenstern's stated purpose in his consideration of measurement was to introduce a leavening degree of realism into what he considered too optimistic a view among the Cowles Commission economists concerning the application of the econometric methodologies they were in the process of developing. At that point in the evolution of econometric thought, fundamentally as a consequence of the argument made earlier by Haavelmo that focused on the disturbance as an errors-in-equations phenomenon, errors in data had been emphatically supplanted as a necessary rationale for the use of statistical methods.

Time has passed and, today, with not only the Haavelmo probability revolution complete, but also the absorption of the concepts of time series analysis, among other acquisitions, few, if any, econometricians now feel the theoretical requirement for observational measurement error to serve as a motivator for the disturbance term. Consequently, it has seemingly become progressively easier over the years to view errors in data as not only theoretically inessential, but also as serving solely to introduce complexity and inefficiency, if not bias and inconsistency, and therefore as being fundamentally obscurantist in effect.

It is nevertheless quite easy to incorporate errors in data into the general linear model:

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{u}$$

where:

- $\mathbf{y}$ – a vector of T observations on a variable Y
- $\mathbf{X}$ – a matrix of T observations on k regressor variables
- $\boldsymbol{\beta}$ – a vector of k unobserved constant parameters
- $\mathbf{u}$ – a vector of T unobserved disturbances

and the variables $\mathbf{y}$ and $\mathbf{X}$ are interpreted to be observed with error:

$$\mathbf{y} = \mathbf{y}^* + \mathbf{e}_y$$

$$\mathbf{x}_i = \mathbf{x}_i^* + \mathbf{e}_{xi} \; i = 2, 3, \ldots k$$

The result is a revised specification, stated in terms of the observed $\mathbf{y}$ and $\mathbf{X}$:

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{v}$$

where

$$\mathbf{v} = \mathbf{u} + \mathbf{e}_y - \mathbf{E}_x\boldsymbol{\beta}$$

and, if a constant term is included, the first column of $\mathbf{E}_x$ is a vector of zeros, in order to reflect that this constant term is not observed. The remaining columns of $\mathbf{E}_x$ of course consist of the unobserved measurement errors on the regressor variables.

It is not difficult to characterize ideal properties for the augmented disturbance term, v_t, or even less-than-ideal properties, possibly at the cost of additional representational complexity. However, more important presently, notice that this formulation results in the measurement error seemingly being absorbed into the econometric representation, rather than being located (as input errors) in the context of the computational error. But the implication is not that the computational process is therefore necessarily unaffected by the presence of measurement error. To the contrary, should $\mathbf{X}'\mathbf{X}$ be ill conditioned, for instance – but not $\mathbf{X}^{*\prime}\mathbf{X}^*$ – the properties of the computational situation will obviously have been worsened. Recall the discussion in the last section of this chapter essentially to the effect that it is the near singularity of the matrix actually used in the calculations that will lead potentially to catastrophic computational error. The matrix of observed values is here a surrogate for the unobserved matrix of "true" values $\mathbf{X}^{*\prime}\mathbf{X}^*$ and this surrogacy raises the question of the degree to which the properties of this latter matrix are preserved in the observed values matrix $\mathbf{X}'\mathbf{X}$. Quite clearly, if the measurement process results in measured values that do not have essentially the properties of the "true" values, the calculations might be presumed to result in parameter estimates and other calculated "econometric" values as having statistical properties that are flawed, all other things equal. Yet it is also conceivable that, *purely as calculations*, the calculations could have better computational error properties in the presence of measurement error than if the calculations were performed using the "true" values of **y** and **X**. It is easy to demonstrate that errors in data in an idealized computational context can be expected to bias parameter estimation (Theil, 1971, pp. 607–615), but notice that the point just made is *not* a statement about the properties of estimators in such a context.

In contrast to this consideration of the properties of the observed values as elements of the computational process, econometricians have instead historically focused on the statistical properties of the unobserved measurement error – except when this error has been summarily banished from thought. Commonly, when considered, the treatment has involved the assumption of both an additive error with white noise properties and an idealized computational context. However, especially in the case of macroeconomic measurements, which are usually provided by the originating data sources as numbers *ostensibly* containing 5–6 significant digits (in the case of recent current dollar US GDP) or more often fewer (in the case of, for instance, consumption expenditures on motor vehicles and parts), not only is there a hard-to-answer evaluative question about the accuracy of the measurements – to one digit, two digits, three digits of precision? – but there is also the consideration that the internal representation of these numbers in the computer using double or higher precision might be justifiable solely in terms of the effect on roundoff error propagation. Representing these numbers in the machine as if they were reported to 12–14 significant digits certainly does not otherwise improve the computational precision of parameter and other "econometric" values – not even in the complete absence of any measurement error. In keeping with Morgenstern's argument, the coarseness of these economic measurements – seen as a lack of measurement precision due to too few significant digits – might be as important a computational consideration, if not more so, than the inability to represent validly the measurement error as an idealized white noise process, as might be more commonly presumed.

Chapter 2
Econometric Software: Characteristics, Users, and Developers

As the title indicates, this chapter takes as its major themes three aspects of econometric software, namely its characteristics, its users, and its developers. What it is, as a manifestation of its characteristics, is obviously quite relevant to this study as a whole, but so also are those who use it, how it is used, and by whom it has been created and developed. The users and developers of any type of software are clearly each formative influences, as may also be the particular way it is used, since collectively these circumstances shape and possibly explain its characteristics. Of course, to a degree, these are separable topics, or at the very least they are topics that can be considered progressively, beginning with the general characteristics of this software.

Following from the discussion in Chap. 1, there is a useful distinction to be made initially between the act of creation of software that is motivated by the goal of performing a specific task and the production of a finished program that is intended to be used for a more generally defined purpose and possibly by others. For instance, a given source code routine, or more often a set of routines, might be written specifically in order to calculate and display a set of parameter estimates. Alternatively, the goal might be to create a program to be used to build a "model," or a class of models, or to perform some other composite, possibly quite complex extended task. Notice that what distinguishes these examples from each other is that the first defines what can be viewed as a *pure* computational problem, with success or failure to find the solution judged by the degree to which the specific calculations are accurately, efficiently, and even elegantly or perhaps quickly performed. What could also be included in this same qualitative assessment is whether, and to what degree, the results are informatively and even attractively displayed. In contrast, in the alternative, more general case, what is involved and needs to be evaluated is the potentially much more elaborate creation of the necessary computer code to perform an integrated series of tasks, not all of which are individually computational problems in the same pure sense. Furthermore, the performance of these tasks might also incorporate operations that only by an extreme stretch of imagination are likely to be classified as either economically or econometrically interesting – at least as the subject matter of these disciplines is usually defined. Nevertheless, the storage, retrieval and management of data, for example, as well as the development of the human interface of a program, are just as much a part of econometric software creation as is the programming of specific calculations. Moreover, whether all these aspects are

C.G. Renfro, *The Practice of Econometric Theory*, Advanced Studies in Theoretical and Applied Econometrics 44, DOI 10.1007/978-3-540-75571-5_2,

now individually considered to be econometrically relevant or not, the future development of economics and econometrics may nonetheless be affected by the degree to which each of these constituent software development problems is appropriately solved.

At first sight, the essential distinction between these two cases is a subtle one, at this stage even possibly obscure, making it necessary for clarity to recognize that the crux of the matter is the question of the extent to which the work performed should be seen by economists and econometricians to be within-discipline, not somebody else's concern. In the most general case, the econometric computational problem should be considered to comprise not only the process of performing specific calculations but in addition provision for the acquisition of an *appropriate* data set, the management and useful display of that data – possibly displaying both intermediate and final results – and maybe even the integration of a large number of computational tasks. In effect, in this general case, this computational problem can be defined as incorporating all the operational aspects of a given applied research project ranging from obtaining the data used from the original source or sources to the presentation of the published results, in whatever form these are presented – rather than just the implementation of specific formulae that might appear in an econometric text book or elsewhere in the literature.

However, although such an argument can be made, it can fall on deaf ears. Most often, economists have been focused in their research interests upon economic agents and their behavior, individually and collectively, and the econometrician upon such topics as the properties of particular parameter estimators, the circumstances of their use, and the evaluation of that use, frequently defined ideally. Consequently, econometric software development is an area of inquiry that is ordinarily interpreted to fall well outside the formal bounds of economic or econometric investigation, as a subject best left to the computer scientist or someone else, certainly as regards its detailed aspects. It is indicative that, with few exceptions, econometric textbooks and the more general econometrics literature ordinarily only refer to the existence of econometric software, without considering its specific characteristics. It is moreover telling that the various aspects of this software, even those most obviously "econometric" in nature, have seldom been considered and evaluated within this literature, except for relatively uninformative "software" reviews (McCullough & Vinod, 1999). Mainstream journals are not the usual locus of ongoing discussion of even the major developments most likely to affect applied economic research. Indeed, these journals normally reject independent submissions that focus too obviously on specific computational matters, even in those cases that these underlie and are critical to the evaluation of applied economics findings. The possible exception to this rule is numerical accuracy (McCullough & Vinod, 1999), but only recently and still in token form (Brooks, Burke, & Persand, 2001; Bruno & De Bonis, 2004; McCullough, 1997, 1999, 2004; McCullough & Renfro, 1998; McCullough, Renfro, & Stokes, 2006; Stokes, 2004b, c, 2005). Economists ordinarily deal with computational issues at a full arm's length and depend upon others, often statisticians and the statistics literature, to intercede (Altman, Gill, & McDonald, 2004; Cordeiro, 2007; Hotelling, 1943; Longley, 1967; Simon and

James, 1988; Velleman & Welsch, 1976), notwithstanding that some related conceptual issues have been considered (Belsley, 1986, 1991; Belsley, & Kuh, 1986; Belsley, Kuh, & Welsch, 1980). Reflecting this situation, most economists are quite aware of John von Neumann's role in the development of game theory, but comparatively few know that he also wrote the first computer program and participated in the design of the first stored program computer (Knuth, 1970; Neumann, 1993/1945). Of these accomplishments, which ultimately will be judged to have had the greatest impact on economic progress is an interesting question.

In order to appreciate better the relevant issues, it is pertinent to consider in contrast another technological development, namely the first production of a printed book, using movable type, which occurred in 1455 or thereabouts. The invention of printing obviously has had a fundamental impact on the entire world of knowledge. More specifically, the transmission of information about economic and econometric thought and research historically rests upon the previous development of this technology, as does also the general diffusion of economic knowledge. Therefore it is not difficult to make a connection, as an enabling circumstance, between the development of the printed book and the development of economics or econometrics. Yet it is also immediately apparent that book production by or for economists or econometricians – either as regards particular technical production aspects or in terms of the publication or distribution of these objects – is only disciplinarily pertinent insofar as the specific focus of a particular investigation becomes their pricing or marketing, or perhaps the industrial organization of the book trade. Even then, it might still be difficult to identify those properties of the book as an object that would merit its isolated consideration as a uniquely characterized economic good. To make a convincing argument that the investigation of the technical aspects of book production appropriately falls within the discipline of economics or econometrics would thus appear to be rather thankless, notwithstanding that most economists spend much of their working lives participating in activities intimately associated with either the production or use of books in either hardcopy or digital form. In short, the example of the book illustrates that simple proximity is not the issue, nor are per se any historical developmental dependencies. On what grounds does econometric software present a better claim for attention?

Developmental Characteristics of Econometric Software

The essential nature of econometric software can be considered prescriptively, following from a definition of econometrics and then proceeding to a determination of what characteristics this software does, should, or must have. An alternative approach, and the one that has been adopted here, is to define econometric software as consisting of that software intended to be used as econometric software; that is, software that is purposefully created by its designers and developers to be "econometric" in nature. This second definition of course accords with the description in this volume's preface and in Chap. 1. However, if too literally interpreted, proceeding in this way risks a high degree of definitional circularity.

A possible solution to this potential circularity problem is to begin by considering the history of the development of this software, placing it in the context of the historical development of the electronic computer, as was done in Chap. 1, which considered also at least certain aspects of the particular economic and data processing environment from which both the computer and economic computation developed. These histories are not as separate as might be imagined. For instance, as the example of von Neumann illustrates, certain of the people closely associated with the design of the early computers and computer peripherals, if not obviously economists, are at least recognizable for their important contributions to the discipline or for the part they played in creating the context from which econometric computing came. In addition, it so happens that a number of easily identified economists and econometricians were among the earliest to start using the computer as a disciplinary research tool, use that has continued from then to the present day. This story also has several other interesting and relevant aspects that lead naturally into a consideration of the modern characteristics of econometric software.

The Early History of Econometric Computation

The first use of computers by economists was a case of need meeting opportunity. It goes almost without saying that the 1930s and 1940s constitute a formative period for both economics and econometrics. During this period, in Britain, the United States, and elsewhere, the organized compilation of economic statistics by governments, trade organizations, and other such originating data sources began to be pursued in earnest, although at first much of this effort was actually expended by individuals. As described briefly in the Introduction, the time just after World War II more precisely marks the stage at which economic and social accounting first began to become fully institutionalized (Hicks, 1990; Kenessey, 1994; Kurabashi, 1994; Meade & Stone, 1941). These developments, although the culmination of an even longer process, were supported by a generally felt need to monitor and better understand economic phenomena, reflecting both the impact of world wars and the economic trials and tribulations of the 1920s and 1930s. In the 1930s, several statistically significant events of course occurred, among them the establishment of the Econometric Society and the publication of Keynes' *General Theory*. In addition, in 1936, not only was the *General Theory* published, but also Leontief's first work describing input-output relationships (Leontief, 1936). And, at about this time, Jan Tinbergen created the first macroeconometric models. Meanwhile, associated with the work of Ragnar Frisch, Trygve Haavelmo, Tjalling Koopmans, Richard Stone, and a number of others, there was also a quickening development of the methodology of econometrics and quantitative economics, which of course extended through the 1950s and beyond.

A revealing perspective on the circumstances of those times is provided by a comment incidentally made by Lawrence Klein in 1950 (Klein, 1950). Referring to the seminal Cowles Commission work and the methodological progress of the 1940s, he rather pessimistically observed (p. 12) that

> An annoying problem that arises with the new methods is the laboriousness and complexity of computation. Very economical techniques of dealing with multiple correlation problems have been perfected, but they can no longer be used except in special cases ... where the system is just identified. Unless we develop more economical computational methods or more efficient computing machines, the problems will remain beyond the reach of individual research workers.

This remark clearly demonstrates why economists were then ready to embrace the computer. Yet further evidence of this readiness is Leontief's purposeful participation in 1947 at one of the first gatherings of computer designers, the Symposium on Large-scale Digital Calculating Machinery, where he discussed certain aspects of his then-current work on interindustry relationships (Leontief, 1948). In his presentation, he described some of the challenging computational problems he saw that might be ameliorated by the use of such machines. Leontief was also during that time successively an active user of the IBM-supported electromechanical Mark I and II data processing and computational machines, in the process becoming the first economist to use a computer, in the particular sense of using an "automatic" device (Wilkes, 1956).

Other economists took similar advantage of the opportunities available to them. The first stored program computer to begin operation was the EDSAC (Electronic Delay Storage Automatic Calculator), independently built at the University of Cambridge by a team under the direction of Maurice Wilkes, yet in design a sibling of the EDVAC (Wilkes, 1956, 1985; Wilkes & Renwick, 1949; Wilkes, Wheeler, & Gill, 1951). The EDSAC was made available for academic research starting about 1951, the first electronic computer to be used in this way (Rosen, 1969; Wilkes, 1956; Wilkes & Renwick, 1949). Members of the recently formed Department of Applied Economics (DAE), several of whom had earlier been invited to Cambridge by its Director, Richard Stone, not only employed this machine (Aitchison & Brown, 1957; Farrell, 1957; Houthakker, 1951; Prais & Houthakker, 1955) but, in particular, Alan Brown, Hendrik Houthakker, and S. J. Prais (Brown, Houthakker, & Prais, 1953) appear be the first to describe in print the process of using such a device as a disciplinary research tool. Lucy Slater, working with Michael Farrell at the DAE, created the first econometric software, consisting of regression and matrix manipulation programs for the EDSAC (Barker, Dada, & Peterson, 2004; Slater, 2004, 1962).

Computationally inclined economists elsewhere still operated electromechanical desktop calculating machines, including Lawrence Klein, Arthur Goldberger and their colleagues at the University of Michigan (Goldberger, 2004). However, in 1954, these econometricians were able to use the semi-electronic IBM Card Programmed Calculator (CPC), in addition to the 602A Calculating Punch, an electromechanical plugboard and card punch device, in order to estimate moments in preparation for the estimation of model parameters for the Klein-Goldberger model (Klein & Goldberger, 1955; Sheldon & Tatum, 1973). A few years later, in 1958–1959, at what would become the Lawrence Livermore Laboratory, Frank and Irma Adelman were the first to solve an econometric model using a computer, an IBM 650 (Adelman, 2007; Adelman & Adelman, 1959; Knuth, 1986). Contemporaneously, possibly at the IBM Scientific Center in New York, Harry Eisenpress wrote the first program to perform Limited Information Maximum Likelihood estimation (Eisenpress, 1959), and a few years earlier, with Julius Shiskin in Washington, DC (Shiskin & Eisenpress, 1957), created the Census X-11 seasonal adjustment method using a UNIVAC (UNIVersal Automatic Computer), developmentally also a sibling to the EDVAC. This machine was built by another team under the direction of Eckert and Mauchly (Rosen, 1969). It was installed at the US Bureau of the Census in 1951 and was the first stored program computer to be sold commercially. Another UNIVAC was the machine used to predict the outcome of the presidential election in 1952.

The Takeoff Period of Econometric Software Development

This description of early computer use by economists possibly appears to be a series of exhibits, selectively chosen, but actually this work effectively constitutes the complete early record, the salient exceptions being additional work by others at the DAE (Barker et al., 2004; Begg & Henry, 1998; Cramer, 2006; Prais & Aitchison 1954) and the beginning of Robert Summers' Monte Carlo study of estimator properties using an IBM 650 at Columbia University (Summers, 1965). The simple fact of the matter is that, throughout the 1950s, computers were scarce, difficult to gain access to, and expensive to use; an hour of machine time could cost literally triple the monthly salary of the economist using it (Adelman, 2007). As Bernard Galler has pointed out, "before 1955, any university that wished to establish a computing activity either had to build its own computer or have a special relationship with a manufacturer" (Galler, 1986). Consequently, only 47 universities had installed them by 1957 (Keenan, 1963). In addition, as mentioned in Chap. 1, the necessary programming infrastructure took time to build: recall that it was only in 1957 that Fortran, the first high level programming language, was developed (Backus, Beeber, Best, Goldbereg, Haibt, & Herrick, 1957); prior to that users needed to program in machine or assembly language. Hardware reliability was also a factor: only in 1959 did transistors begin to replace vacuum tubes (Rosen, 1969) and only in the 1960s did computers based upon this more reliable technology become available in the full sense of the word.

As a consequence, it was during the next two decades, starting in the early 1960s, as computers began to proliferate and programming languages and facilities became generally available, that economists more widely became users (Bodkin, 1999; Desai, 2007). Beginning then, there were a number of econometric firsts, including the implementation of increasingly computationally complex econometric techniques, among them Two and Three Stage Least Squares, Seeming Unrelated Regression Equations, and Full Information Maximum Likelihood (Renfro, 2004a, b, c, d). In addition, starting about 1964, economists created some of the earliest large scale computer data bases (large scale for that day), also developing the software to manage these (McCracken, 1966, 1967a, b). However, even in the mid 1960s, the progress made was neither uniform nor universal: the Wharton model, a direct descendent of the Klein-Goldberger model (Bodkin, Klein, & Marwah, 1991; Intriligator, 1978), was then still being solved using an electromechanical desktop calculator (Renfro, 2004a, b, c, d; Schink, 2004). It was only in the second half of that decade that economists at the University of Pennsylvania first used the electronic computer to solve large-scale macroeconometric models as nonlinear simultaneous equation systems, rather than as "linearized" reduced systems (Desai, 2007; Evans & Klein, 1967, 1968; Preston, 2006; Schink, 2004).

During the 1960s, expanding bootstrap-like on what had been learned, and making use of the ongoing technological developments, the level of sophistication progressively increased. Much of this work represented the efforts of individual economists, although often in the context of formal research projects (Duesenberry, Fromm, Klein, & Kuh, 1965, 1969; Evans & Klein, 1967, 1968; McCarthy, 1992; Preston, 2006). This process began with the creation of single purpose software to estimate parameters, manage data sets, and later solve macroeconometric models (Renfro, 2004a, b, c, d), but at the end of this decade, with the advent of time-sharing computers, economists were among the first to create and implement network-resident interactive software systems (Renfro, 1970, 2004a, b, c, d), a significant initial step in the development of the modern econometric computational environment. Beginning in the early 1970s, they made several concerted attempts to give focus to this research. At the just founded MIT Center for Computational Research in Economics and Management Science, under the direction of Edwin Kuh, economists began to devote considerable effort to the study of relevant computer algorithms (Berndt, Hall, Hall, & Hausman, 1974; Dennis, Gay, & Welsch, 1981a, b; Dennis & Welsch, 1978; Holland & Welsch, 1977; Kuh, 1972, 1974; Kuh & Neese, 1982; Kuh & Welsch, 1980) and closely related regression diagnostics (Belsley, 1974; Belsley, & Kuh, 1986; Belsley et al., 1980). Following such advances, during the 1970s economists then proceeded to develop several wide-area online telecommunications-linked economic data base, analysis and econometric modeling systems that by the end of that decade became used worldwide (Adams, 1981, 1986; Drud, 1983; Renfro, 1997a, b, 1980a). In other places, such as at the London School of Economics, the development of software played an integral part in the improvement of the methodology of specification search (Pesaran & Pesaran, 1987, 1997; Pesaran & Slater, 1980), with the software specifically conceived to be used as a tool to foster the so-called General-to-Specific, or LSE, method (Hendry, 2003b; Hendry & Doornik, 1999a, b; Hendry & Srba, 1980; Krolzig & Hendry, 2001; Mizon, 1984). As a consequence of initiatives of this type, including numerous smaller scale, even individual efforts, economists were then ready to adopt the emerging microcomputer at the beginning of the 1980s, by the middle of that decade starting to use it widely as the primary locus for analytical processing, including even the solution of econometric models of 600 and more equations (Renfro, 1996).

However, to provide the proper perspective for a more detailed evaluation of the implications of these advances, it is first necessary to place them in their appropriate historical context. The mention of microcomputers here is potentially misleading, given their ubiquity today, for although it is true that the first microcomputers became available to "hobbyists" as early as 1975, as a general proposition the 1970s were still very much the age of the bigger machines, for economists as well as most other people. Only later in this decade did economists begin to write software for the microcomputer (Renfro, 2004a, b, c, d). Furthermore, many of the econometric software initiatives just mentioned took almost 10 years to come to term, so that considering 1970s computer use in context it is important to recognize that the predominant computational characteristic was not only its mainframe focus but, as a matter of use, also a continuing selectiveness. The 1960s were a time during which economists more generally became computer users, as suggested earlier, but this was a change relative to the use in the 1950s. In 1970, only a small number of economists, or people at large, had directly begun to use the computer. In the case of economists, many were then graduate students, and of these only a proportion customarily spent each night and weekend in the keypunch room. Today, in contrast, computer use of course begins at kindergarten, or even before, and extends to any age pervasively, but such behavior began no earlier than the late 1980s.

Today there is also stress upon user accessibility. The human interface is much more evolved, in a way that permits the user to operate programs at a much higher level of abstraction. In contrast, programs in the early 1970s often, and in the later 1970s sometimes, still needed to be operated by placing numbers in fixed fields of punched cards, a given integer number from 1 to k indicating which among k options the user wished to select – or by the use of 0 the user might indicate omission. Sometimes data transformations needed to be made explicitly, the user coding these in Fortran or other high level language. One of the most significant differences between then and now is that in those earlier times, especially in the years prior to 1975, it was almost always requisite for a user to begin by writing some code, particularly in those instances that an application involved new techniques. So much was there a common need to program, and comparatively so much less computer use than today, that practically any econometric software written prior to about 1975 involves some aspect that can be declared a "first," or at minimum viewed as involving some type of pioneering effort. There are certainly examples of off-the-shelf software used by (comparatively small groups of) economists during the period between 1960 and 1980 (Bracy et al., 1975; Brown, 1975; Goldberger & Hofer, 1962; Hendry & Srba, 1980; Kim, Chung, & Kim, 1979; McCracken, 1967a, b; Press, 1980; Slater, 1967, 1972); nevertheless, it was then usual that an intending hands-on user needed to learn to program, to at least some degree and often at a rather fundamental level. Until at least the middle 1970s, an applied economist would commonly either start with nothing more than a textbook, or, at best, be given a box of cards, or sometimes one or more paper tapes, onto which were punched the source code statements for a program – although by 1975, perhaps even earlier, it was not unusual for the "box of cards" to have been replaced by a machine-readable card image file.

However, in 1970, there were also certain incentives that led individual software developers to begin to focus on the human interface and abstract symbolic processing, as well as large scale data management, whereas others created particular algorithms or programs that since have been developed effectively as representative of what might almost be considered econometric "schools" of thought. Yet other economists and econometricians, as just indicated, created or modified particular programs for their personal use, certain of which ultimately became publicly available and widely used. Particular examples of programs of the first type, not all of which are still being developed, include DAMSEL, EPS, MODLER, TROLL, and XSIM, each of which were then associated with the creation, maintenance and use of often sizeable macroeconometric models (Renfro, 1980a). The use of such models characteristically imposes the need to create and maintain relatively large time series databases, involves the processing of symbolic data in the form of equations, and establishes a requirement for software that can be used by teams of people, hence the incentive to focus on the human interface. Examples of programs of the second type, individually interesting because of their internal algorithmic characteristics in the 1970s, include B34S, TSP, and Wysea, although TSP is also notable for the early development of its human interface. Programs of the third type, those that can be construed to be individually associated with a particular econometric "school," include AutoBox and AUTOREG (represented today by its direct descendant PcGive). Finally, programs created originally in the 1970s for local, even personal use, but that have since been developed for public use, include AREMOS, CEF, FP, LIMDEP, MicroFit, Modeleasy+, RATS, REG-X, SHAZAM, SORITEC, and WinSolve. Some of these might also be regarded as being assignable to a "school." These pigeonhole categories should all be regarded as being only tentative, but they nevertheless illustrate aspects of econometric software development before 1980.

During the 1970s, there were also larger forces propelling the development of econometric software. Overall, the economic spirit of that time was distinctly activist. At the end of the 1960s, in keeping with the particular Keynesian paradigm then prevailing, not only was there some feeling among economists that the economy was possibly precisely manageable (although possibly not to the degree this belief has been represented since), but – just as important – there was also a broader willingness on the part of government officials and corporate leaders, particularly in the United States, to believe in the capability of the economist to "fine tune" the economy. All this was to a degree the consequence of the fact that then the memory was still vivid of both the 1930s depression and the escape from it in the 1940s and 1950s. In 1945, it was popularly believed, an expectation shared by a number of economists, that an economic downturn was likely, that World War II possibly represented a temporary period of "full employment" that would give way to widespread unemployment when "the troops came home" (Klein, 1946; Orcutt, 1962; Woytinsky, 1947). However, 25 years later, at least in the case of the major industrialized countries, the recessions that had occurred had proved to be of short duration and represented minor fluctuations about a distinctly upward trend, particularly in comparison to before the war. One of the consequences, in the later

1960s, was substantial forthcoming corporate and government support for the creation of economic consulting firms such as Data Resources, Chase Econometric Forecasting Associates, and Wharton Econometric Forecasting Associates, each of which about 1970 began to create and support large scale, computer-resident economic data bases and econometric software systems (Renfro, 1980a, 2004a, b, c, d).

The development of econometric software during the period between 1960 and 1980 can therefore be seen as reflecting two distinct "motivating forces": those internal to econometrics and those external, often deriving from economics as the parent discipline. Individual econometricians characteristically created software dedicated to parameter estimation using a variety of estimators; the academic imperative driving this type of innovation was of course the usual desire to present the new and different. For the major economic consulting and forecasting firms, the imperative was to provide billable services to clients; that is, to applied economists in governments, corporations, and other organizations. Software developed for the use of these clients was usually motivated by the goal to provide time-sharing software services in combination with access to substantial time series economic data bases, in many cases via telecommunications links (Adams & Ross, 1983; Mendelssohn, 1980; Renfro, 1980a). An important aspect of this software development was its emphasis on the creation of semi-natural language, symbolic command interfaces of the type that can be found even today as macro languages, intended to be easy-to-learn and easy-to-use (Adams, 1981; Drud, 1983; Kendrick & Meeraus, 1983; Meeraus, 1983; Renfro, 2004a, b, c, d). Another important aspect of this work, reflecting the widening range of users – many of whom might be less likely to call themselves econometricians than simply economists or even planners, statisticians, or something else – was the creation of more broadly-based facilities to support onscreen tables, graphs, and even maps, although not yet at today's standards.

Both types of software development have proved to be important ultimately, each in its own way. Individual software development, for self-consumption, does not normally result in programs that can be used easily by other people, no matter how econometrically interesting the algorithms created. In contrast, programs developed intentionally for general use tend, by design, to offer not only "user friendly" interfaces, but also even extensive data display capabilities, which actually can be quite important in applied research, even if this type of software is open to criticism for "usually [lagging] some years behind the state-of-the-art technical econometric frontier" (Hendry, 1993, p. 314). The academic payoff, such as it is, tends to be much greater for software development that leads to the publication of information about new econometric technologies or embodies what are perceived to be theoretical advances, but this return represents a private, not necessarily a social benefit, beyond the creation of new knowledge that may or may not be of wide interest. In contrast, the greatest social benefit may derive from the implementation of new computational technologies that support the research of all economists and econometricians alike.

The particular historical details matter. The 1970s were years of substantial aspiration, yet also a time when the economist's reach might be seen to exceed his grasp. Among the reasons this circumstance needs to be recognized is that economists have been known to indulge in a form of poetic license. Specifically, in introductions to journal articles and other publications, or in the body of the piece, they have too often provided stylized, frequently not particularly well-founded statements of “fact,” intended as motivators for the argument presented. But even if not originally written to be definitive history, nor expected to be wholly believed, at least some of these statements have later tended to be cited repeatedly and in the end read as gospel, finally entering into the folk wisdom of the discipline. It is easy to demonstrate that the perspective provided by contemporaneous statements about technology, especially when read 10, 20, 30 or more years later, can be materially deceptive. For example, John Diebold, writing in 1962 (Dibold, 1962), presented an enthusiastic description of the capabilities of the language translation programs of that generation, asserting that that they could be used to “scan a printed page, translate its contents from one language to another, make an abstract of the translations and store both text and abstract in ‘memory’ until they are called for by an information-retrieval network” (pp. 40–41). To anyone historically knowledgeable, this description clearly involves more than just a touch of hyperbole, as it would even if applied to modern language translation programs 45 years later.

In addition, it can be difficult even for those who were there to recall accurately all the relevant characteristics of the past when there has been substantial technological change in the meantime. Modern readers of an historical account may be even more inclined to assign present day capabilities to the past, making it difficult for them to separate historical fact and fiction. For instance, also in 1962, Daniel Suits wrote about macroeconometric models (Suits, 1963), asserting that “in the days before large digital computers, individual relationships generally had to be kept simple, and the number of equations that could be used in a system was rather small. Today [that is, 1962], with the aid of high speed electronic computers, we can use models of indefinite size, limited only by the available data” (p. 7). For proper perspective, it needs to be appreciated that this assessment precedes by several years the first successful computer-based solution of any econometric model, the single special-case exception being the Adelman work mentioned earlier. Furthermore, from 1962 to 1965, *every working macroeconometric model* (of the few operating) was solved using electromechanical desktop calculating machines, not computers. Even including the Adelman simulations, before 1965 model solutions were invariably enabled only by linearization of the model and then by first solving out all but a few variables. As indicated earlier, modern methods of solution only began to be implemented in 1967 (Desai, 2007; Preston, 2006; Schink, 2004); even then, in certain cases, desktop calculating machines continued to be used. Suits’ assessment was written before *any* software systems had been developed that had the capability to manage *to any significant degree* computer-resident economic data bases or to estimate, particularly at the scale suggested, the parameters of individual relationships or to handle any of the other computer-related tasks necessary to support the use of even small models, much less those of “indefinite size.” If unrecognized, past technological mis-assessments can cause hard-won successes to be undervalued and the causes of past failures to be misjudged.

The Adoption of the Microcomputer

The computers ordinarily used by economists prior to the middle 1980s can be classified as mainframes or minicomputers, although by then supercomputers had also appeared, after being introduced in the 1960s, even if seldom used by economists. Mainframes were of course intended (and priced) for "enterprise" use, whereas minicomputers were meant to be used in departments and other, generally small, sub-classifications of organizations. Supercomputers, typically a category of fast, vector processor machines in the 1970s, had the interesting characteristic that they generally used mainframes as auxillary processors for the input of data and output of results. However, irrespective of such classification, the fundamental characteristic of econometric computing before the later 1980s was not only that the computers used by economists were ordinarily organizationally owned, rather than personally, but also that they were shared. Mainframe sharing at this time meant that, however fast the machine's operation when used by a single person, it could be quite slow for the average (multi)user. It might be slow because it operated in batch mode, with programs prioritized and queued as they were read in and, on output, the hardcopy results sorted and distributed by hand by its one or more operators. In many contexts, output in those days was essentially hardcopy, notwithstanding that it was possible to create a permanent or semi-permanent machine-readable record. The result could be turnaround times of an hour or more, or even 24 hours for "large" jobs, those requiring memory in excess of 512 KB or, sometimes, as little as 256 KB. But even when machines were used in "time-sharing" mode, the fact that individual users' "jobs" were still almost always prioritized and then queued, and might require a human operator to mount tapes and removable hard disk drives, could mean that several minutes or more might pass between the entry of a single command and the computer's response. The first personal computers were themselves slow, compared either to mainframes or minicomputers (or personal computers today), but they were single user and self-operated. These characteristics caused them to be time competitive with mainframes, and sometimes even supercomputers, as early as 1981 or 1982 (Fried, 1984, p. 197).

The adoption problem the microcomputer initially posed for the econometrican was the lack of software, which always occurs when the hardware characteristics are radically changed because of the introduction of an entirely new Central Processing Unit (CPU). In addition, the architecture of this new class of computer at first also represented a significant step back in capabilities: maximum memory size on the order of 64 KB, rising to 640 KB only more than a year later, and small, slow diskette drives for permanent storage rather than hard disks, with hard disks initially unavailable and then reaching the size of 20 MB, as a common characteristic, only in 1984 – not to mention CPU operating speeds of 6–8 MHz or less before 1986 (Byte, 1984, 1986). Furthermore, it took several years before microcomputer software provided the capabilities of mainframe software, reflecting that writing software takes time, but also that the language compilers and linkers available for the microcomputer were at first "bug" ridden, idiosyncratic, and originally designed for small, memory-undemanding programs. In at least one case, in 1982, it was necessary to "patch" an existing linker in order to make it possible to convert an econometric software package from the mainframe to the PC.

Nevertheless, by the autumn of 1983, the first econometric software package capable of estimating and solving (small) econometric models was available for the IBM Personal Computer and compatibles. In September 1984, a microcomputer based economic forecast service was introduced at the annual meeting of the National Association of Business Economists, combining a 250 + equation Wharton Quarterly Econometric Model of the United States with the MODLER software (Renfro, 1996). The solutions for a 12 quarter forecast horizon took less than 4 min. This software had the same capabilities as its mainframe version (Drud, 1983); in particular, it could be used to create, maintain and solve econometric models of as many as 1,000 equations. By the end of 1985, other packages available for the "PC" included AREMOS, AutoBox, Gauss, PcGive, RATS, SHAZAM, SORITEC and Stata, as well as limited versions of both SAS and SPSS. Even earlier, at the beginning of the 1980s, more limited packages had been implemented on both a Tandy machine and the Apple II (Renfro, 2004a, b, c, d), including a program called "Tiny TROLL," created by Mitch Kapor at MIT, parts of which were then incorporated into the VisiCalc spreadsheet package and subsequently also influenced aspects of the development of Lotus 1-2-3, and later other packages, such as Excel.

Many of these individual efforts continue to have a modern day relevance, but to explain the subsequent evolution of this software during the present microcomputer age, it is possible to trace the broad outlines of the computational developments of the past 20–30 years. The computational shift during the 1980s, from the creation of software and systems on large institutionally based machines to the use of the personal computer as the locus of such work, can be viewed as responsible for the range of econometric software that exists today. The personal computer, because of its affordability, ultimate wide distribution, and steadily increasing capabilities, not only provided an important context but also became the basis of a progressively more extensive market. The 1960s may have been the decade that economists first began to learn to use the computer, but it was the 1980s and subsequently that computer use became widespread in a pervasive sense. The comparative degree of market extensivity is even more apparent today, given the ubiquity of the notebook, or laptop, computer, and otherwise the sheer number of personal computers now commonly found in offices and homes, not to mention such things as the recently accelerating convergence of television and computer technologies. Of course, the development of the Internet as an effective successor to the more local, mainframe-based wide area networks of the 1970s has obviously had a significant impact, particularly since the middle 1990s, especially on the distribution of economic data and information.

Consequently, although it is possible to talk in terms of nearly 60 years of evolution, the impetus for the development of today's number and variety of econometric software packages is decidedly more recent. Their present characteristics are the direct result of a combination of relatively modern circumstances, among them being the introduction of the microcomputer in the 1970s and 1980s, the essentially simultaneous expansive development of econometric techniques since the 1960s

(Gilbert & Qin, 2006), and most recently the increasingly common adoption of a graphical interface, often while preserving macro language capabilities, in conjunction with the progressively more widespread use of the Internet since the 1990s. Among the effects, the broadening and deepening of econometrics – and, more generally, quantitative economics – especially during the past 30 years, has had a significant impact on the range of the present day properties of these programs, resulting in considerable diversity. For example, functionally classified, today they can be placed in categories that include basic regression, advanced estimation, and econometric modeling languages. Considered in terms of both functionality and interface, they can be classified as ranging from those defined by specific-selection, menu-oriented econometric features to algebraic quasi-natural language econometric modeling and programming languages that provide also the capability for an individual user to create new techniques (Renfro, 2004a, b, c, d, p. 59ff).

Substantive changes in hardware have also occurred during the past 20 years. As indicated, the desktop Personal Computer in 1987 operated at 6, 8, or 10 MHz; in contrast, many modern notebooks operate at or near 2 Ghz or better. The 1985 microcomputer ordinarily contained at most 640 KB of easily accessible memory; today's variety commonly contains as much as 1 gigabyte or more. Furthermore, the microcomputer has progressed to the point of incorporating (in a single chip package) even two to four processing units (with the prospect of eight or more in the foreseeable future), as well as having other characteristics that make it more and more difficult to conceptually distinguish between the capabilities and types of large and small machines in a meaningful way that does not involve mind-numbing detail. What is certainly true is that the microcomputer found either on the desktop or an airline tray table is now the locus of the vast proportion of all the empirical analysis that is done by economists. In almost every sense, the composite history of the electronic stored program computer is now present in the modern personal machine.

The Characteristics of Econometric Software

To this point the focus has been upon the developmental characteristics of econometric software during the past nearly 60 years. An inference that might be drawn is that, on the one hand, there are historical examples and, on the other, modern examples. It also might be thought that the historical examples are only of historical interest. To a degree, this characterization is reasonable: a number of econometric software programs were created, used for a period of time, often years, and then dispensed with. However, to a significant degree it is misleading. None of the programs created in the 1950s are still employed today, but certain of those from the 1960s continue to be, although in modern form. In particular, AutoBox, B34S, Microfit, MODLER, Mosaic, PcGive, TSP and Wysea all have their origins then and at least some still incorporate a certain amount of original source code. Furthermore, the majority of these programs continue to be maintained by their original principal developers. Others, including AREMOS, FP, LIMDEP, Modeleasy+, RATS, SHAZAM, and SORITEC began life on mainframes in the 1970s, as did also SAS, SPSS, and other well-known statistical software packages; in some cases, these too continue to be maintained by their original developers. All were converted to microcomputers, in most cases beginning at various times during the period 1980–85. In contrast, REG-X began to be developed on a Tandy microcomputer in 1979, was moved to a mini-computer and then to the PC in the 1980s. Others, including EViews (as MicroTSP), Gauss, and Stata, began to be developed on the microcomputer in the 1980s, joined by Betahat, EasyReg, Ox, and the present day incarnation of TROLL in the 1990s. Of all the recognized existing programs, only gretl began to be developed in the present century, albeit on the basis of "inherited" code, even if there are also certain Gauss and Ox-based special applications that have been created during the past few years. The packages just identified include those that have been surveyed and are evaluated in this monograph.

Inasmuch as the origins of most date from before 1980, their history and that of the electronic computer are intertwined. Econometric software spans the development cycles of hardware change from earliest times. For instance, in addition to the early use of the computer described in the first part of this chapter, the first use of the computer by economists at the University of Pennsylvania apparently involved some later use of the UNIVAC during the early 1960s, the immediate design successor to the EDVAC (Desai, 2007; Preston, 2006), although this is almost incidental. However, the connections to the second generation are quite meaningful. At least three of the existing packages began to be developed on second-generation computers, and several more on the third. The distinguishing hardware characteristic of the second generation was the introduction of the transistor, which occurred first in 1959 with the IBM 7090/94 (Rosen, 1969). Another machine, the IBM 7040, was effectively an IBM 7090 "lite." The IBM 1130 and 1620, used in several cases by economists, were second generation, small mainframes principally designed for scientific use. The CDC 6400, used in at least one case, can be described as a second generation mainframe, although it is architecturally compatible with the earlier CDC 6600, designed by Seymour Cray, which is generally regarded as the first supercomputer. Interactive, local area econometric computing began in 1970 at the Brookings Institution on a Digital Equipment PDP-10 (Renfro, 1970), another of which was later used by Bill Gates and Paul Allen (www.pdpplanet.com). The IBM 360 was a third generation machine and was used by econometric software developers, as were also its successors the IBM 370 and the 3090. Other econometric software developers, especially those in the United Kingdom, if they did not actually cut their teeth on the first EDSAC, can nevertheless date their earliest work to the use of the Atlas in the 1960s, particularly the machines at the Universities of Cambridge and London, or even the EDSAC 2 or Titan (Slater & Barker, 1967). More recently, econometric software has involved the use of Apple, Tandy, the Victor 9000, the RS/6000, several Sun machines, and multiple generations of the IBM PCs and compatibles. The inference to be drawn is that econometric software enjoys a long and rich hardware patrimony, one only partially described here.

However, until recently, this history has been part of the econometric deep background. Only certain individual developers have ventured into print to any significant degree (Belsley, 1974; Eisner, 1972; Eisner & Pindyck, 1973; Hendry & Doornik, 1999a, b; Hendry & Srba,1980; McCracken, 1967a, b; McCracken & May, 2004; Renfro, 1981, 1996, 1997a, b, 2004a, b, c, d; Slater, 1962; Stokes, 2004b, c; White, 1978). Furthermore, although the user guides and reference manuals commonly provided with individual programs often do give some information about their history, these accounts tend to be presented selectively, ordinarily without technical details. The most readily available, collective description of the existing econometric software packages, albeit somewhat limited, is found in the compendium published in 2004 (Renfro, 2004a, b, c, d). This collection comprises edited accounts by each of the current principal developers of each of the existing packages, although there are certain exceptions to this rule. The exceptions occur mainly in the case of historically significant programs that are today no longer maintained. Other, more selective, descriptions of particular econometric software packages, available in 1983 and earlier, can be found in an article of that date by Arne Drud (Drud, 1983), articles in a special issue of the *Journal of Economic Dynamics and Control* (Kendrick & Meeraus, 1983), and in minimally descriptive compilations of statistical software by Ivor Francis and others (Francis, 1981).

It is said that the past is a foreign country, but if the detailed, step-by-step record is now difficult to recover entirely, it is possible to determine the salient characteristics of these packages during modern times on the basis of an earlier interactive survey made in 2003. This survey was taken in conjunction with the publication of a special volume on econometric computing, published in 2004 both as volume 29 of the *Journal of Economic and Social Measurement* and a separate book (Renfro, 2004a, b, c, d). A number of the more general operational characteristics of the individual packages are documented there in the form of summary tables (Renfro, 2004a, b, c, d). In addition, the compendium just referred to (Renfro, 2004a, b, c, d) is included. It is interesting that the transition from desktop calculators to the electronic computer that began to take place in the early 1960s originally occurred in the form of a modal transfer: calculations previously made with the calculator began to be made instead using the computer, but initially without a significant change in mindset (Desai, 2007; Goldberger, 2004; Slater, 1962). After that first step, came the process of incorporating into this use both more comprehensive data management and more than particular parameter estimation methods. As mentioned earlier, the first recognizable econometric software commonly took the form of separate, single purpose programs classifiable individually as data management, data transformation, and regression programs, the latter in their original form not always easily distinguished from "statistical" programs of that day. To the degree that evident differences existed in the middle 1960s, the most obvious characteristic of econometric software was less of a tendency to include stepwise regression and more to include simultaneous equation techniques, such as Limited Information Maximum Likelihood and Two Stage Least Squares. It was only in the late 1960s, and even then only occasionally, that the programs became more than rudimentary in operating style and econometricians even began to think about something as conceptually sophisticated as software "design."

In contrast, during the past 25 years, reflecting the impact of personal computers, econometric software packages have become clearly categorically distinguishable, both from other types of software and from each other. Among themselves, as a general property, individual programs have become functionally more self-contained, combining parameter estimation capabilities with data transformation facilities and at least a minimal degree of more generalized data management and display capabilities, a number of packages increasingly integrating as well such capabilities as nonlinear multi-equation model solution facilities. Since 1995, there has also been a pervasive tendency to adopt the prevailing standards of the so-called "graphical user interface," associated with both Microsoft Windows and the Apple operating systems, although just as noticeable it has also been common for econometric software to continue to offer command line control, usually in the form of a scripting or macro capability. Most programs are today able to operate by manipulating econometric objects using a keyword-based command language, even if many operate primarily using menus and icons. It is also common to permit users to collect command elements into a text file, as a macro. The motivation is the repetitive nature of many of the operations performed during research; for example, requiring the ability to make data transformations repeatedly as new observations are acquired, or to rerun regressions. The ability to recycle commands, in order to perform previously executed tasks easily and repeatedly, is obviously a desirable trait.

The way in which the various specific econometric techniques came to be embedded in software during the past 50 years can also be outlined and usefully classified. Certain of these developments represent a widening, or broadening, in the number of econometric techniques, tests, and other operations implemented in software. Others represent a capital deepening process, in the sense of more sophisticated implementations that, in some cases, take the form of more complete algorithms that subsume the capability to perform any of a multiplicity of more elementary operations, including two or more econometric techniques in combination. In other cases, this deepening involves combining in the same program a sequence of operations that are mutually integrated, such as permitting parameters to be estimated as a first stage operation, followed by the very nearly automatic creation of model equations, and then linking these equations, as a next stage, finally causing the creation of a functionally complete model capable of being solved (Renfro, 2004a, b, c, d). Such broadening and deepening can be considered to be algorithmic in nature, although as also involving stylistic elements.

However, another aspect of this software development took the form of the creation of progressively more sophisticated interfaces, as discussed earlier. One of these is the human interface, the means by which the program user both controls the operations performed and either perceives or comprehends the results, which may or may not be the same thing. As mentioned before, in the 1960s, sometimes even in the 1970s, program control was effected by choices made using numbers located in fixed fields on punched cards or paper tape. This type of control has long since been replaced by the use of the WIMP graphical interface (Windows, Icons, Menus, and Pointing methods) and even earlier by the use of free-form, if still stylized command languages. The results generated may, in turn, be displayed in tabular form, or as graphs, or as other perceivable objects, such as an equation or a list of equations. Comprehension, as opposed to simple perception of the program's output, obviously can be aided by interface design, even if there has been considerably less attention paid by econometric software developers to this aspect of the human interface than to enabling simple perception.

Another interface type is the machine interface, the way in which a given computer either receives input from or sends output to one or more other machines. The idea of facilitating and then generalizing this interface, including its hardware aspects, so as to permit computers to intercommunicate effectively began to be implemented at the beginning of the 1970s, when it became progressively more desirable not only to connect individual users to machines remotely from a dumb terminal via a telecommunications link, either dial-up or dedicated, but also one computer directly to another. Peer to peer machine linkages were initially difficult to achieve, for computers in those days were originally designed to operate singly, not as either intelligent or co-equal correspondents. Connections then generally required some type of master-slave protocol. More recently, the machine interface has of course often taken the form either of a Local Area Network (LAN) or a Wide Area Network (WAN) connection, the latter including both the Internet and other machine-to-machine linkages. For econometric software developers, these were initially separated innovations, for ordinarily these developers were not involved in the establishment of machine interconnection protocols, as this is an operating system task. However, once these connections began to be possible, remote data retrieval and data base management, among other facilities, began to become important as ideas and in practice (Anderson, 2006; Anderson, Greene, McCullough, & Vinod, 2007; Harrison & Renfro, 2004; Renfro, 1980a, 1997a, b; Ritter, 2000), even if today it is still usual for econometric software packages to be designed simply to read in data from some type of text file or an Excel or some other spreadsheet file, rather than to query a relational or other remote data base system using SQL or other procedural language.

Aspects of the Evolution of Software Features

Mary Morgan (Morgan, 1990) and Qin Duo (Qin, 1993) have each described the process of the development of econometric theory and the way in which the ideas of Frisch, Haavelmo, and Koopmans, among others, and the work of Tinbergen, Klein, Goldberger and others during the early days of macroeconometric model building combined to establish both econometric practice and its received theoretical support at the beginning of the 1960s. Qin's assertion (p. 65) that "estimation can be seen as the genesis of econometrics, since finding relationships has always been the central motive and fulfilment of applied modeling activities" expresses well what can be regarded as a motivating thought behind the beginning efforts to more generally employ the electronic computer in the first few years of the 1960s. However, the operative philosophical position of those years was often that expressed in

1958 by Haavelmo (1958, p. 351), that "the most direct and perhaps most important purpose of econometrics has been the measurement of economic parameters that are only loosely specified in general economic theory." Of course, this measurement often took place without always sufficiently taking into account his clearly stated qualification (p. 352) that the quantification of economic phenomena had in the preceeding 25 years appropriately come to be interpreted to extend "not only to the measurement of parameters in would be 'correct' models, but to the field of testing, more generally, the acceptability of the form of a model, whether it has the relevant variables, whether it should be linear, and many other similar problems." The methodology debates at the end of the 1970s and into the 1980s stand as testimony to the continued lack of testing as a practice, which, as will be discussed in the next chapter, at least in part possibly reflected the slowness with which facilitating statistical tests became embodied in the software.

In the early 1960s, the electronic computer, as it became progressively more commonly available, represented to economists the potential to perform computations not feasible previously. Eisenpress's creation in 1959 of a program that implemented limited information maximum likelihood was followed in 1962–63 by the efforts of Zellner and Stroud to implement the Two and Three Stage Least Squares (Zellner, Stroud, & Chau, 1963a, b) and Seemingly Unrelated Regression Equations (Zellner, 1963a, b) techniques. This work by Zellner and Stroud marks the first time that particular estimation techniques were contemporaneously introduced in the literature (Zellner, 1962; Zellner & Theil, 1962) and implemented in software that could be used by others. A short time after that, in 1963–64, Mike Wickens programmed Full Information Maximum Likelihood, based upon a later-published formulation by James Durbin (Durbin, 1988) that, among other things, utilized Newton-Raphson convergence and demonstrated that the second iteration of the process generated Three Stage Least Squares estimates. Elsewhere, during this time, other econometricians also implemented estimation techniques in software; much of this work took place in Canada, New Zealand, the United Kingdom, and the United States (Bodkin et al., 1991; Klein, 1960). In most cases, these efforts can be seen to be motivated by the desire to make these calculations specifically for the sake of it. Other efforts in the middle to late 1960s – including follow on work in New Zealand (Bergstrom, 1967a, b; Phillips & Hall, 2004), as well as the program development that took place in Washington, DC at the Brookings Institution (Duesenberry et al., 1965, 1969; McCarthy, 1992), and that at the Wharton School of the University of Pennsylvania (Evans, 1969; Evans & Klein, 1967, 1968; Preston, 2006; Schink, 2004) – represented much more the need to support the estimation, construction, and use of macroeconometric models. However, as this was the take-off period of econometric software development, being the first dispersed attempt to create a software infrastructure, in almost all cases the initial effect was broadening, rather than deepening, as more and more estimation and even model solution techniques became embodied in software.

A broadening also took place in the 1970s that in many cases and in similar ways at first represented the efforts of individual econometricians, yet has since resulted in the general availability of packages such as AutoBox, B34S, BRAP, FP, LIMDEP, Microfit, PcGive, RATS, and SHAZAM. Recall that these programs appear to have originated either as individual reactions to the local unavailability, or simply the general absence, of appropriate software or else as solutions to one or more specific, perceived econometric problems, or, indeed, the combination of these circumstances. Sometimes, as in the case of MicroFit and PcGive especially, this software development increasingly over the years included the incorporation of misspecification tests and other evaluative features. But whatever its exact form, most of this broadening, beginning then and extending to the present day, constituted the addition of econometric techniques. However, these efforts did not simply represent an increase in the number of techniques to be applied in a given, possibly macroeconomic time series context, but, in certain cases, the development of software to be used instead in a cross-section or panel data, often microeconomic environment. The greater availability of survey data, both cross-section and panel, as well as econometricians' advocacy of Bayesian, Time Series Analysis, and other specific methodologies provided much of the initial broadening stimulus in the 1970s. In the 1980s and 1990s, the market possibilities provided by the microcomputer and, in later years, the Internet, added extra stimulus. However, some of these packages, even in their early days, also supported the applied research of economics departments and groups of economists at such diverse places as Auckland, the Brookings Institution, Chicago, Cambridge, Harvard, the London School of Economics, Minnesota, MIT, Pennsylvania, Princeton, and Wisconsin, so were not just being developed in isolation for their developers' personal research use.

The phenomenon of software deepening is both most evident and easiest to describe in the case of programs developed for research teams associated with large-scale econometric model projects. The need to manipulate and display substantial quantities of data in conjunction with the creation and use of such models, starting in the middle 1960s, led increasingly during the 1970s to the creation of large scale economic data base management systems, both separately and as subcomponents of such packages as DAMSEL, EPS, MODLER, Mosaic, and XSIM (Renfro, 1997a, b). From the 1960s to the later 1980s, data series often needed to be acquired in hard copy form and then keypunched. The associated expense obviously provided an incentive to develop ways to move the data, once in machine-readable form, from one context to another with a minimum of effort, as well as to manipulate the observations easily. Models containing 300 or more equations only became possible because of the computer hardware and software advances that began in the 1960s, although at first models of this size certainly strained the existing computational capabilities. Even in the early 1970s, to create a 200 equation model was commonly held to require a year's effort on the part of a team of 10–12 people (McCracken & Sonnen, 1972). In 1987, in contrast, one person working alone could estimate, construct, and successively solve a 300 equation model in a single week (Cooper, 1987; Renfro, 2004a, b, c, d).

The objects that are associated with macroeconometric models containing hundreds or even thousands of equations obviously include data series, which explains the development of data base management capabilities. However, somewhat less immediately obvious, they also include equations, multiple tables, graphical displays, macros used repeatedly to make transformations and updates, and other such items that also need to be managed effectively. These objects collectively constitute a significant data management problem that involves not simply classification and organization, but also a general problem of information management that includes the need to be able to search effectively. In addition, from the beginning there was a requirement to incorporate labor saving features; for example, the manual coding of individual model equations itself was time consuming, but in addition likely to result in transcription errors. Otherwise, in common with other types of software, deepening in this context also took the form of the creation of program components capable of performing a variety of selected transformative operations on a particular data input stream (Hendry, 1976). As indicated earlier, this intensification process can be considered both as an internal program phenomenon, as just briefly described, or else in connection with the development of human command interfaces that make possible the more sophisticated control of a program's operation.

The Development of the Human Interface

One of the evolutionary characteristics of econometric software – as discussed briefly earlier, and in greater detail elsewhere (Renfro, 2004a, b, c, d) – was the early development of explicit econometric modeling languages, which began in the late 1960s. The use here of the term "language" refers to the command structure as a human interface, which permits the user of this type of software to describe to the software the operations to be performed using an algebraic syntax and vocabulary, together with keywords and variable names; for example resulting in transformation commands (possibly simultaneously taking the form of identities) such as:

$$\mathrm{Y} = \mathrm{C} + \mathrm{I} + \mathrm{G} + (\mathrm{X} - \mathrm{M})$$

The variable names (Y, C, I, G, X, M) not only have an obvious mnemonic aspect, but as command elements each constitutes also a symbolic reference to a stored vector of observations. The use of the program's command language therefore not only directly invokes the retrieval of observations from an organized data base, and perhaps subsequently the storage of results there, but also defines and causes calculations and other operations to be performed that can be associated with the construction, maintenance, and use of an econometric model, a model that might contain even hundreds or a thousand or more equations. However, once created, such a program can also be used more prosaically to make simple data transformations, as shown above, as well as to perform regressions, execute a variety of analytical tasks, display tables, graphs, and the like, all in a relatively user friendly way. Consequently, as previously described, the development of econometric modeling languages in the 1970s was often associated with formation of economic consulting and forecasting firms, which then made available to a wider public both software services and economic data for analysis (Renfro, 1980a).

The IBM Personal Computer at its introduction in 1981, with its original DOS (Disk Operating System) command line user interface, can be seen to be immediately compatible with the type of command line operation associated with the econometric modeling languages developed for use with time sharing mainframes in the 1970s. Furthermore the interactive operation of time sharing operating systems, which normally provided the context of the early development of such modeling languages, was functionally (if only very locally) mirrored by the single user operating systems of the microcomputer. Therefore, from the first, the microcomputer provided a new, yet also quite familiar environment. What this machine in addition soon made available to each user, beginning in 1982, was a pixel-based screen display that permitted graphical displays of a superior type that involved a matrix of points, in the form of pixels, that were individually addressable. Such as screen can be described as being "all points addressable," rather than only line by line. Only rarely available previously to users of mainframe computers, this type of screen provided the environment for the development of the modern Graphical User Interface (GUI). Incidentally, the particular circumstance that caused the IBM Personal Computer and compatibles to be selected by almost all econometric software developers in the early 1980s, rather than the Apple, Tandy, or other microcomputers, may reflect the early availability for this machine of Fortran and other algebraically oriented compilers, in addition to the inclusion in its technical specifications of a numeric coprocessor chip, the 8087, which permitted faster floating point numeric calculations. For many years, the Apple machines, in particular, provided attractive frosting but almost no cake; with the exception of its stunning display, only in the present century has the Apple finally become hardware competitive with the PC.

Of course, the Personal Computer and modeling languages were independent developments, even if the microcomputer environment, taken together with the subsequent widespread use of this computer, caused a fundamental change in the degree of computer use worldwide. Considered alone, these modeling languages represent a logical extension of the development of high level programming languages that began in the middle 1950s. Both the parsed evaluation of alphanumeric commands and the translation of arithmetic/algebraic expressions, usually involving the conversion of infix notation (for example, a + b) into reverse polish (for example, ab+) or some other operative syntax that permits stack-based processing, constitute operations that are – or can be seen to be – common to both compiler design and econometric modeling languages. In turn, linker operation and the functional integration of a sequence of operations so as to marry the output of an earlier one to the input requirements of a later one are logically generally analogous in their essential characteristics.

During the 1970s, there was almost always a noticeable difference between the human interface of the econometric software packages typically used by academic economists and that experienced mainly by business and other nonacademic economists, who used the econometric modeling language type of interface just described. This difference in part reflects that, during the 1970s, batch processing mainframes and minicomputers were much more commonly available in academic environments than were computers with time sharing operating systems. The typical self-programming academic econometrician in the 1970s might, in any case, have had little incentive to develop a sophisticated language interface for a program, compared to the incentive to focus upon econometrically interesting algorithms, but in a card (or paper tape) oriented batch environment there was even less reason. AUTOREG, B34S, LIMDEP, and most other such programs were originally developed with an algorithmic focus, rather than on the interface. Programs such as DAMSEL, EPS, MODLER, and XSIM were more human interface-biased in their development. The combination of differences in developer incentives and their environments explain the particular diverse characteristics and almost bipolar orientation of econometric software development during the 1970s.

However, it is also pertinent that, until about 1978, much of the design and development of econometric software occurred under relatively isolated conditions. There was a time in the early 1970s that journals, in particular *Econometrica*, appeared ready to publish articles and notes about software, but for whatever reason this was a short-lived, Prague Spring. With certain exceptions (Belsley, 1974; Eisner, 1972; Eisner & Pindyck, 1973), it was only at the end of this decade that program descriptions and algorithmic details noticeably began to appear in the disciplinary literature (Dent, 1980; Hendry & Srba, 1980; Kendrick & Meeraus, 1983; Kirsch, 1979; Lane, 1979; Pesaran & Slater, 1980; Society for Economic Dynamics and Control, 1981). Otherwise, econometric software and its documentation ordinarily passed from hand to hand, even if user guides to statistical programs had begun to appear in university bookstores. Of course, in the days before microcomputers, software purchases were commonly made organizationally, usually by people who worked in computer centers and spoke of "statistical," rather than "econometric" software; in addition, it was decidedly uncommon for software of any type to be prominently marketed at the annual economic association and society meetings. Furthermore, even as late as 1983, computational methods were ordinarily investigated separately from any explicit consideration of their algorithmic computer implementation, and the citations that appeared in the formal economics and econometrics literature were often not directly related to any such implementation (Quandt, 1983), a practice not unknown today.

These circumstances of econometric software development before 1985 are relevant to the consideration of particular developments since. Furthermore, at the risk of stereotyping, it is useful to consider certain of the resulting properties of econometric software as the microcomputer began to be used widely, starting in about 1985. In particular, whatever the specific differences between programs in the 1970s, at that time it was almost universally characteristic of econometric software packages that they each offered specific, program dependent user choices. In the case of the econometric modeling languages, the user might be able to choose to create a possible variety of models, but the parameter estimation facilities were for the most part given. Some degree of flexibility might exist that would allow distinguishable techniques to be combined, such as Two Stage Least Squares and autoregressive corrections. Such packages also might offer greater capabilities to the degree an economist was willing to program, but essentially the typical user made his or her choices as if from a menu.

However, in 1985, a new type of software began to become available, the earliest example familiar to economists being Gauss (Hill, 1989). It is possible to argue that too sharp a distinction has just been made, that the econometric modeling languages already offered capabilities similar to those of these new packages, albeit described in the depths of thick manuals, but it is useful to ignore this particular fine point in order to focus on the difference *in orientation* of these two types of econometric software package. Packages such as EPS, MODLER, and XSIM are examples of econometric *modeling* languages (EML) (Renfro, 2004a, b, c, d), as has been discussed, but Gauss, Ox, and possibly other similar packages are effectively econometric *programming* languages (EPL). The critical difference is the object the user works with: an econometric modeling language characteristically has as its objects specific, well-defined econometric techniques, to include estimators with explicit names. Other objects take the form of time series variables, model equations, and models, but also a range of variable transformations, defined in terms of algebraic and arithmetic operators, and, as well, also implicit functions.

In contrast, an econometric programming language is defined by its mathematical and, in some cases, statistical objects. These objects include matrices, vectors, operators, implicit functions, a looping syntax, and a particular grammar, among other characteristics. As its name implies, an econometric programming language is a *programming* language, and one that is specifically oriented to the use of econometricians and economists. Generally, it is also a higher-level language than Fortran, C++, and other commonly recognized *computer* programming languages. An aspect of its high level nature is that the user is ordinarily not expected to be familiar with computer operating systems and other aspects of the particular use of a computer programming language. However, it is difficult to make hard and fast distinctions. Clearly, there is a potential classification question that could be raised concerning exactly how to distinguish an econometric programming language from any other programming language of a sufficiently high level. Similarly, as indicated earlier, an econometric modeling language can contain an econometric programming language as a sub category.

Suffice it to say that these are fuzzy sets. However, ignoring such categorical complexities, econometric software can today be classified into standard estimation packages that provide an economist with the ability to perform a given set of econometrically defined operations, operations that are specifically determined by the software developer. Notice that the operational characteristic in this case consists of the user *selecting* from a set of options, possibly using a menu. There is next a mid-range, which most obviously includes the econometric modeling languages, with the characteristic that the economist is required not only to make certain selections but also to determine how particular operations are performed: he or she must form equations, combining variables and operators and possibly implicit functions, and thereby *build* a model. These models can be *solved* or *simulated*. The results can be *plotted* or produced as *tabular displays*. Advanced Estimation Packages (AEP) or Generalized Estimation Packages (GEP) that both offer a selection of choices and incorporate a macro language capability should also be included in this classification, as offering a subset of capabilities and features. Finally, the econometric programming language in turn offers less in the way of prefabricated statistical and mathematical objects, but more scope to *create* new econometric, statistical, and mathematical forms. It also might be possible to infer that an econometric programming language is most suited to use by econometricians, as creators of emerging techniques, as opposed to applied economists, who are more likely to use established methodologies, hence another type of package. Obviously, these sharp distinctions are most meaningful when considering polar examples of these package types.

Considering the human interface aspects of the modern econometric software packages, the classifications just described can be considered to imply substantial progress, inasmuch as the ideal might be to present economists with the capability to perform their research in the most immediately intuitively obvious way. For certain analysts, interested only in the use of standard econometric techniques, it is clearly beneficial for econometric software packages to be available that are easy to learn to use and involve little effort to apply. For others, the capability to learn an econometric language that is language compatible with the material presented in textbooks and journal articles would appear to offer much, even if this capacity might also imply the need to specify explicitly the calculations made in each case.

More generally, it might seem possible to infer from this description that this apparent movement towards a complete econometric programming language represents for economists what the development of CAD/CAM (Computer Aided Design/Computer Aided Manufacturing) has meant for architects, designers, engineers, and others, namely the creation of a productive environment in which it is possible to both design a new entity and at the same time constructively establish its specific characteristics. In an engineering context, the CAD component can be interpreted to permit the production of a design for a particular object; the CAM component ideally then permits the design itself to control the machine, or machines, that then produce this object. Alternatively, it might be possible to see these econometric software developments as implying potentially much the same type of near term functional improvement in econometric practice as modern word processing software has brought to document production, namely, in this case, a screen representation that is effectively the same visually as the final printed document, a characteristic that usually goes by the name what-you-see-is-what-you-get, or WYSIWYG.

All this sounds good at the outset, but there are certain aspects of econometric software that make these concepts less than immediately applicable. In the first place, in the case of econometric software, there is no necessity for there to be a direct correspondence between what appears on the screen and the computations that are made. Users of this software generally do not and will not know the algorithmic details of the computations performed, for the simple reason that individual developers do not ordinarily publish these details. Furthermore, whatever the user specifies in the form of a command, there is no necessary relationship between this command and the specific calculations algorithmically performed by the software. At issue here is not only the user's ability to specify the characteristics of a particular arithmetic or algebraic operation, but also the specific way in which various conditions are evaluated, such as, for instance, the manner of convergence in the context of an iterative nonlinear process, or the user's freedom to set initial values and other control parameters (McCullough & Renfro, 1998, 2000).

Fundamentally, whatever set of commands the user provides will be *interpreted* by the software package, acting as an intermediating agent. The actual calculations then performed are determined and controlled in advance by the software developer. Obviously a choice that the developer can make is to permit the user's commands – whenever appropriate – to be implemented exactly as stated, but even this possibility is the developer's choice and therefore constitutes intermediation. The choice to allow the user algorithmic control is by no means necessarily the best. In at least certain cases, perhaps even most cases, it can be argued that it is desirable that the user of the package *not* be allowed to control precisely how the program does what it does inasmuch as that user cannot be presumed to be a knowledgeable numerical analyst, nor necessarily an experienced programmer who will also take the time to evaluate qualitatively the results obtained.

Directives Versus Constructive Commands

In certain respects, the discussion has come full circle since the introductory section of Chap. 1. Recall the argument made there that, over the past 30–40 years, specialization has occurred, with the majority of economists effectively ceding responsibility for the design and development of econometric software to a minority of econometricians. In contrast, one of the implications of an aspect of the modern development of this software, namely the creation of econometric programming languages, would appear on the face of it to provide any enterprising economist with the effective capability (once again?) to design and develop his or her own software, but now in a way that avoids many complexities, yet achieves the goal of allowing that person to determine the constructive characteristics of whatever applied econometric research project he or she might wish to undertake. An objection that has been made to this idea is the argument just posed that, in any case, the designer and developer of any econometric programming language actually remains in control as an intermediating agent, whether this control is exercised or not. A normative question that naturally arises is, to what degree and how should this designer/developer actually exercise this control given the inevitable complexities of the computational process?

In order to address this question properly, a certain amount of background information is necessary. It is useful to begin by considering exactly what distinguishes a directive from a constructive command. A *directive command*, or simply a *directive*, as this term will be used here, can take any of a number of forms. For example, in order to direct a program to perform an Ordinary Least Squares regression of a named dependent variable, such as CE, on one or more other named regressors, the user might in one case issue the commands:

Dependent: CE
Regressors: YPD, CELAG1

in another:

CE = F(YPD, CE(−1))

or, in a third:

Reg Command: CE = c1*YPD + c2*CE(−1) + c3

Each of these types of directives are found in the command languages of one or more econometric software packages. It is also true that in some cases, pull down or drop down menus will instead be used in order to identify progressively the dependent and regressor variables.

All these directives are constructively equivalent, inasmuch as none do more than direct that a certain type of operation be performed. In all cases, the command's meaning and the particular corresponding default operations will have been established by the program's designer. That is, the meaning of the directive is completely established by the syntax and vocabulary of the program used. However, as illustrated, a directive can in some cases seemingly or even actually have constructive features; for example, notice that in the second command above, the term CE(–1) itself constitutes the directive that the variable named CE is to be retrieved from the program's data storage component and then lagged by one period before the observations on this variable are used as one of the regressors in the implied regression. In the third case, a linear-in-parameters regression specification also appears to be explicitly indicated. Nevertheless, notice also that none of the directives considered are, except by default, linked to a particular regression method.

In contrast, a textbook consideration of the general linear model and Ordinary Least Squares regression will commonly begin with a statement like:

Consider the linear regression equation:

$$\mathbf{y} = \mathbf{X}\beta + \mathbf{u}$$

where:

y – a vector of T observations on a variable Y
X – a matrix of T observations on k regressor variables
β – a vector of k unobserved constant parameters
u – a vector of T unobserved disturbances

This opening set of definitions will be followed, by and by, with the statement that the ordinary least squares estimator is defined as:

$$\mathbf{b} = (\mathbf{X}'\mathbf{X})^{-1}X'\mathbf{y}$$

If this operation were actually to be carried out constructively using pencil and paper, given an understanding of linear algebra and the availability of a particular data set, the most direct way to proceed is to compute first the sums of squares and cross products of all the relevant variables and then load these into a matrix. As is shown in almost any modern econometrics textbook (Davidson, 2000; Greene, 2003a; Johnston & DiNardo, 1997), and even in many older ones (Goldberger, 1964; Johnston, 1963; Theil, 1971), if this matrix is formed so that the cross-products of the dependent variable with the regressor variables border those of the regressor variables alone, a result is obtained that can be characterized as:

$$\begin{matrix} \mathbf{X}'\mathbf{X} & \mathbf{X}'\mathbf{y} \\ \mathbf{y}'\mathbf{X} & \mathbf{y}'\mathbf{y} \end{matrix}$$

Constructively, the next step is simply to invert the interior matrix, $\mathbf{X}'\mathbf{X}$. Following this inversion, the estimated value $\mathbf{b}$ can then be computed by carrying out the matrix multiplications indicated by its above apparently constructive definition. Somewhere, in individual textbooks, at least historically, Cramer's Rule may be provided as a constructive definition of matrix inversion.

In contrast, if this estimation process were to be considered as a computer programming task, using some combination of a programming language such as Fortran or C++ and possibly Assembly language, the process of computing the estimates can instead be programmed so that, *as the computation occurs*, the rightmost column of the original bordered matrix *simultaneously* becomes the location of the estimated values of the parameters (Goodnight, 1979), denoted by $\mathbf{b}$:

$$\begin{matrix} (\mathbf{X}'\mathbf{X})^{-1} & \mathbf{b} \\ \mathbf{y}'\mathbf{X} & \mathbf{y}'\mathbf{y} \end{matrix}$$

where $\mathbf{b}$ *is* the set of estimated parameter values. The reason to make the calculations in this way, rather than to compute:

$$\mathbf{b} = (\mathbf{X}'\mathbf{X})^{-1}\,\mathbf{X}'\mathbf{y}$$

by making the implied matrix multiplications, once given $(\mathbf{X}'\mathbf{X})^{-1}$, is that such matrix operations, if carried out explicitly are in fact not efficient and may in addition result in greater rounding error, compared to the simultaneous generation of the inverse and the parameter estimates. However, as a matter of interface design, the program's developer is in no way constrained not to allow the program user to specify $(\mathbf{X}'\mathbf{X})^{-1}\,\mathbf{X}'\mathbf{y}$ *as a directive*. Neither is the developer constrained to compute $\mathbf{b}$ in any particular way, whatever the user's directive. But does it therefore follow that the developer should always act as a "correcting" intermediary?

In the beginning, specifically in the mid 1960s, it was common to find regression programs that replicated textbook calculations, as indicated in chap. 1. In those days, existing programs were commonly shared in the form of source code and an economist might therefore begin a research project by obtaining a deck of Hollerith cards onto which had been punched this source code. At first, because of time and effort constraints, it was natural in such cases to make only changes that had to be made and otherwise to leave well enough alone. However, in 1967, James Longley (Longley, 1967) evaluated a number of the existing statistical regression programs and discovered that, for at least certain data sets, they could be disastrously numerically inaccurate, essentially because of the problem of ill conditioning and the use of single precision values during calculations. The possibility of this type of computational problem occurring had, in fact, been known to human computers at least as early as 1943 (Hotelling, 1943; Rushton, 1951), if not well before, but it had been forgotten. It is not actually particular to electronic computers, but there is no point now in considering it any more generally.

When the matter considered is the calculation of linear Ordinary Least Squares parameter estimates, or linear-in-parameter estimates more generally, it is possible to regard the fundamental threat to be the degree to which the data used are collinear. As discussed towards the end of Chap. 1, the problem in this case is essentially due to the ease with which the calculations can result in intermediate values that have no precision whatsoever, possibly implying the need at the very least to compute and provide the $\mathbf{X'X}$ matrix condition number as a potential warning (Belsley, 1991; Belsley et al., 1980). More generally, an aspect of the use of finitely precise numbers is that number comparisons can only discriminate between those that differ by a certain minimum amount. It is not meaningful to ask if $\mathrm{x} = \mathrm{y}$, but rather only if $|\mathrm{x} - \mathrm{y}| \leq \varepsilon$, where ε is some suitably chosen small number and x and y are floating point real values. One of the implications is that, even in the linear case, computations such as matrix inversion must be carried out with due regard for the effect of the data used, as a matter of conditioning, as well as the fact that in the end the solution is always *approximate* rather than exact (Higham, 2002; Stoer & Bulirsch, 1993).

This inexactness has a number of practical consequences. For example, to the mathematical economist, there is a sharp conceptual difference between a linear and a nonlinear problem. In contrast, to the econometrician in the guise of a numerical analyst, the environmental computational difference between a linear and a nonlinear problem can be fuzzy. It can be that the latter involves the need to make additional, more open-ended calculations in a context in which each successive calculation could progressively involve additional rounding and approximation error, although it is also true that nonlinear problems can involve specific computational issues, some of which arise from such things as the need to compute derivatives as finite approximations, local versus global maxima, initial conditions, and stopping rules (McCullough & Renfro, 2000). Recall that error propagation is not necessarily particularly serious in the case of multiplication, division, or taking square roots, but simply adding operands of different sign can, in extreme cases, lead to catastrophic cancellation (Stoer & Bulirsch, 1993, p. 11–12). In addition, rounding error can be local to a given iteration sequence in the case of convergent iterative calculations (Ralston & Rabinowitz, 2001, p. 334). The relevant issues and aspects are varied, but in the end what fundamentally needs to be understood is that infinitely precise calculations do not occur within an electronic computer, and that the name of the game is the minimization of calculation error, not its absolute elimination.

When considering these matters, notice also that the devil is in the details. For instance, when considering the textbook expression

$$\mathbf{b} = (\mathbf{X'X})^{-1}\,\mathbf{X'y}$$

it is tempting to interpret it initially as being essentially constructive in nature – in part because of its familiarity to econometricians – but if the sequence of elementary calculation steps are set out carefully certain inherent ambiguities can become evident along the way, especially to the degree that k regressor variables are considered, rather than 1,2, or even 3. For instance, there is the issue of the best way to invert $\mathbf{X}'\mathbf{X}$, as well as the precise way in which other calculations are made, not excluding the construction of the matrix $\mathbf{X}'\mathbf{X}$ itself. The precision with which each of the numbers are calculated needs to be considered – and also the precision with which they are stored at each stage of the computational process. There are numerous opportunities to make serious errors in the calculations that might not be noticed at first, some of which can be due to the order in which the individual calculations are made. And if the final results are presented without the clear identification of the specific numbers used as original inputs to this computational process, it may be difficult for someone else to validate the results later, even given knowledge of each computational step. This example of course represents a relatively simple case in this day and age.

The original consideration of the numerical accuracy of regression programs by Longley (1967) brought to the attention of econometric software developers, among others, the problem of rounding error in the context of single precision floating point numbers. During the intervening years, there has been a significant amount of work done concerning the numerical methods that should be adopted, most recently considered by Stokes (2005), who also addresses data storage precision, as well as alternative matrix inversion techniques. The particular methods of matrix inversion that should be employed, generally speaking, are determined by the characteristics of the data: in particular, near singular matrices imply the need to dispense with the usual Cholesky factorization of $\mathbf{X}'\mathbf{X}$ and to use QR decomposition applied directly to the data matrix. However, before performing a regression, it is commonly not evident just how collinear the data are, which once upon a time created a conundrum: in earlier years, in the 1960s – in the case of mainframes – and the 1980s – in the case of the microcomputer – there was an issue concerning the demands accurate matrix inversion techniques placed upon the capabilities of existing CPUs as well as computer memory. Today, it is generally no longer necessary to worry about this as a matter of computer resource cost: few techniques likely to be employed will today require more than a literal fraction of a second, given the use of a modern computer, and high speed memory has become comparatively abundant. But even if the stakes associated with “capital intensive” methods of calculation are no longer what they once were, it is still true both that a design decision must be made and, if the wrong decision is made, that it may not be obvious to the computer user whenever the calculations involve excessive errors. A classic consideration of the computational problem of error accumulation is that by Harold Hotelling (Hotelling, 1943), but see also Wilkinson (1961), Belsley et al. (1980), and most recently McCullough (2004) and Stokes (2005). For a consideration of the algorithmic properties of certain simultaneous equation estimators, see for example Kontoghiorghes et al. (Foschi, Belsley, & Kontoghiorghes, 2003; Kontoghiorghes, 2000; Kontoghiorghes & Dinenis, 1997).

At present, it is not necessary to consider in further detail the particular numerical analysis issues associated with econometric computation, for the purpose of this exposition is not to identify and catalogue the specific provisions that need to be made for numerical accuracy in each case. The aim is rather to make clear that how calculations are performed does matter and that the provisions made are relevant to the way econometric theory is practiced – or should be. Numerical analysis issues need to be addressed not only behind the scenes, when the design and development of econometric software is specifically considered, but also in the mainstream literature when the putative properties of estimators, diagnostic tests, and other econometric topics are discussed. The reason is simply that the characteristics of the computations made can affect the validity of the inferences that are drawn by those who apply theory to the real world.

However, in addition to the operational need to produce sufficiently accurate results, there is also the matter of computational completeness, briefly discussed earlier. Textbook and journal presentations, jointly and severally, inevitably provide only partial coverage of the range of formulae and calculations that are pertinent. Implementing Ordinary Least Squares, or any other parameter estimation method, so as to produce a program that is generally reliable in its operation requires information that is sometimes difficult to discover: for example, what calculations to permit in a variety of special cases, such as when the program user chooses to suppress the constant term, which among other things affects the supplementary statistics that are ordinarily displayed next to, above, or below the parameter estimates? Alternatively, how should the program react if the user chooses to regress a variable on itself, or omits entirely all regressor variables? Or, what if the data set used exhibits missing values, either at the extremes or for observations in the interior of its range? The relevant consideration here is not that these events are each necessarily the result of sensible actions on the part of the program user, but rather that particular actions, if not dealt with properly, can cause the program to crash, or perhaps to produce (possibly without warning) results that are neither correct nor appropriate under the circumstances. Ideally, *any* action that the user is both permitted and chooses to make should result in a meaningful and graceful response on the part of the program, taking the form either of an intelligible error message or a sensibly determined result.

Still other aspects of the econometric computational problem could be considered. However, it is already self-evident, from the perspective of software design, that many (most?) of the seemingly constructive formulae that populate the econometrics textbooks and the general econometrics literature should in fact be considered to be simply directives. Furthermore, it is clearly quite justifiable to argue that, ideally, those econometricians who create the software used by economists generally – both in the case of standard packages, that involve user selection of options, *and* in the case of econometric programming languages and other contexts in which seemingly constructive commands might appear – should carefully take into account what the translation from the mainstream econometrics literature to software embodiment implies. However, it is not always immediately evident how this translation should be effected. Giving the typical economist what might be appear to be full control of the operations performed, in the case of an easy-to-use econometric programming language, yet monitoring carefully every command – so as to insure numerical accuracy, among other considerations, as well as to permit program control errors to be trapped and then dealt with gracefully – would seem to place the econometric software developer in a courtesan's role, not only assuming, in the famous phrase, "power without responsibility," but also flattering the user into thinking of him or herself as having accomplished something dangerously difficult or at least deeply satisfying, yet all the while with the training wheels still firmly attached.

But is it reasonable to suppose that econometric software of any type, including econometric programming languages, can be created so as to be quite so foolproof to use? An obvious aspect of the matter being considered here is the question whether the programming of econometric computations can ever be simplified in a way that easily permits any type of relevant calculation to be performed and insures the accuracy and completeness of the results? Very early in the historical development of the electronic computer it became apparent that the availability of labor saving "building blocks" would be helpful as programming aides, which lead to the development of subroutine and function libraries (Wilkes et al., 1951). It was evident that, ideally, the creation of such libraries as collections of elemental computational "tools" might both simplify the programmer's task and simultaneously provide algorithmic components that would presumably meet appropriate qualitative standards. One of the beneficial effects might be to permit programs to be created using prefabricated "parts," more or less in assembly fashion. The development of econometric programming languages can be viewed as simply a further evolution of the idea of providing any person wishing to perform calculations with a reliable set of tools from which a selection can easily be made. However, the collective programming experience of now nearly 60 years has shown that the assembly of computer programs can only be simplified to a certain degree, still requires the assembler to acquire skills, and implies the need for an information transfer to occur – in both directions – between the tool builders and the tool users. It also true that even if the programming process itself can be simplified to the point that each individual "tool" is separately easy to learn to use, the inevitable multiplicity of them is likely to make the learning process time consuming. Furthermore, the subsequent application of these "tools" to the applied economic research process involves complexity on this score as well.

Developers, Users, and Use

Certain implications are obvious and particular questions have been posed. It is now time to consider directly the $64,000 question: what then is there about economic or econometric computation that permits the argument to be entertained that, after many years of neglect, the production and use of econometric software are activities that should be consciously regarded as being fundamentally within-discipline, not only that, but as calling for open discussion in a disciplinarily central place? Is it not enough that economists should be content to use this software? Have its properties not already been evaluated enough? It is, of course, true that software reviews have appeared in the econometrics journals, actually more than 120, based upon the count by McCullough and Vinod, which they report in a recent issue of the *Journal of Economic Literature* (McCullough & Vinod, 1999). Unfortunately, almost all these have tended to superficiality in their treatment (McCullough & Vinod, 1999). In particular, among the defects of these reviews, only four considered the numerical accuracy of the software (Lovell & Selover, 1994; McCullough, 1997; McKenzie, 1998; Veall, 1991), just a few attempted to address the comparative suitability of different programs for particular applications (Mackie-Mason, 1992), and some fell short of being independent evaluations. Otherwise, as intimated earlier, to the degree that economists or econometricians have overtly considered the subject of econometric software, there has been a pronounced tendency to treat its characteristics as being at best of very limited disciplinary relevance. Except in exceedingly rare cases, econometric journal articles and textbooks noticeably have not discussed, indeed have seldom ever referred to, any substantive software design or algorithmic issues. Furthermore, as will be demonstrated, when applied research results have been presented, involving computational aspects, the particular software that may have been used to generate those results has seldom even been mentioned.

Use and Users

What can in fact be discovered fairly quickly about the revealed software preferences and use characteristics of the typical economist? The answer is, a few things, which derives from the existence of JSTOR, which has been developed as a full-text-searchable online archive of journals and is of course Internet-based. Among other things, JSTOR permits at least a portion of the economics literature to be keyword searched for such terms as "software package," "computer program," "econometric software," or even for the names of particular software packages, in order to examine the degree to which economists describe their software use in the context of published articles, including for example possibly the extent to which evaluative tests are carefully and knowledgeably used. Welcome to Bibliometrics 101; however, notwithstanding any appearance to the contrary, it is here being taught by a novice.

To start, there are certain important caveats. The first relates to the fact that JSTOR is made available to research libraries not just as a complete archive, but also as separable groupings, including economics journals, business journals, statistics journals, biological sciences journals, and so on. Although a large library might subscribe to all the groups, smaller libraries or specialist libraries can instead purchase selective access. One of the JSTOR characteristics is that these groups are not mutually exclusive, but instead, in certain cases, intersecting sets – as is in fact the case for the ostensibly separate economics and business journals groups. Therefore, it is not necessarily meaningful simply to contrast search results obtained from the business group with those from the economics. Another limitation is that the publishers of most journals impose a multi-year moratorium on the inclusion of their most recent volumes in this archive, thus establishing a “moving wall” that year-by-year inches forward and may vary from journal to journal and group to group; for a group, the results for the most recent years, going back as much as 5 years, may exclude volumes for particular journals. In addition, as an operational characteristic, as everyone knows from Internet search experience, keyword searches involve potential inferential pitfalls. For example, taken at face value, approximately 26,000 articles published during the mainframe period from 1960 to 1984 appear to include at least one mention of the program “GIVE” (or at least words consisting of various mixed cases of this specific sequence of letters, being the name of the predecessor of PcGive). This is no doubt a heart-warming thought for David Hendry, especially as, in contrast, other programs, such as EViews, MicroFit, MODLER, and TROLL, either individually or collectively, seem to be mentioned in only 330 articles during the entire period from 1960 through 2003, notwithstanding the possible appearance of rEViews. Finally, the particular results obtained in each case can also be affected by specific settings of the “Advanced Search” criteria, including “search full text content only” and limitations of the search to “articles” only, versus the alternatives. In addition, the number of journals included in JSTOR is increasing, so that the results reported here might not, even now, be able to be exactly duplicated. However, the good news is that anyone with access to JSTOR can play.

If due allowance is made for the effect of the use of common words, and for the fact that one person’s “package” is another’s “program,” what is easily discovered is that searches using the terms “software package,” “computer package,” “computer program” and “econometric software” together yield a total of less than 2,400 articles in which any of these terms appear during the 64 years from 1950 to 2003 – for this specific collection of terms, a total of 2,395 “hits” are obtained, the first in 1958. This number is reasonably large, but a subsequent article-by-article examination of a reasonable sample of the “hits” obtained reveals that this number represents a substantial overstatement of the degree to which economists have either identified any software used or revealingly discussed their use of the computer during the years since 1960 – especially if only marginally evaluative software reviews that appear as “articles” are excluded (McCullough & Vinod, 1999). Restricting the search to “Articles” alone reduces the total to 1641. Adding “Reviews” increases the total to 1851. The further addition of “Editorials” brings the total to 1853. The number of mentions in the “Other” category alone is 542. It is easily, if tediously, determined that the vast majority of the articles discovered by the search discuss economically relevant aspects of the use of the computer by economic agents or else aspects of the economic role of the computer during these 50 + years, but – for the most part – *not* how economists use the computer.

But if dreams have been shattered, there are still some interesting results. Considering the JSTOR findings in more detail, in the 52 journals identified there as “economic journals,” it appears that from 1960 to the end of 2003, a total of 2,176 articles were published in which the word “software” occurs at least once. The choice of this particular ending date is made because of the "moving wall." The first article in which the word "software" is used was published in 1962 and the second in 1965 (Diebold, 1962; Levy, 1965). The first 109 of these articles were published before 1979, with 122 of them published in the next 6 years; hence, all but 231 have been published since the beginning of 1985 – the first to be published in 1985 (January) happens to be a review entitled *Econometric Software for Microcomputers* that with the exception of MicroTSP (now EViews) and SORITEC focused entirely on *statistical* software packages. Looking further back, the first use of the term "computer program" occurred in 1958 (Moore, 1958), although the first related use of both "computer" and "program" in the context of an article occurred in 1955, appropriately enough one written by Herbert Simon (Simon, 1955). In the 1,098 times "computer program" appears at least once in an article before 1 January 2005, the first 446 occurred before 1981; the last time it appeared was in October 2003. The alternative term "computer package" scores 76 "hits", the first in 1976 and the last in 2003. Only 182 articles contain the more specific phrase "econometric software," arguably the most widely used category of "economic software," although this seemingly more global term itself actually receives a total of only 1 hit. "Econometric software" is of relatively recent coinage: of the articles mentioning it only 14 were published prior to 1987, the first one by Robert Shiller in 1973 (Shiller, 1973). In contrast, 94 have been published since 1995. Perhaps surprisingly, inasmuch as it is commonly considered to be a broader category, "statistical software" receives fewer hits, only 100, with the first article containing this term appearing in 1982.

Anyone who wishes to luxuriate in the innate practicality of economists can take heart from the fact that the composite phrase "theoretical software" is absent from the JSTOR economics database – or does this instead indicate something else? From 1960 through 2003, a total of 60,202 articles are found in which the word "theoretical" or "theory" (or both) appear least once, 55,234 in which "theory" appears, 31,709 for "theoretical," and 26,741 in which both these words appear. In contrast, "software" appears without "theoretical" or "theory" in only 1,136 articles. In order to determine the population of all English language articles published from 1960 through 2003 in economics JSTOR journals, a search for all the articles that contain the word "the" (which can be given the acronym ACTWT) results in 83,659 hits, which essentially provides a database population count, since surely no article can be written in English without using this word. This finding suggests that essentially 72% of all articles may have some theoretical content. On the other hand, the word "data" appears in 52,085, which is 62% of all articles. Unless this finding implies only that data is something economists like to theorize about – from 1960 through 2003, 39,136 articles were published in which "data" and either "theory" or "theoretical" appear – it is seemingly something of a mystery what exactly was being done with all that data.

The combination of these findings with other, independent use-evidence lends support to the idea that economists have often employed econometric software without much mention of that use. Given that people ordinarily use what they pay for, an indication of a significant degree of actual use is the reasonably substantial revenues collectively generated by econometric software programs such as EViews, MicroFit, MODLER, PcGive, and TROLL, in contrast to the approximately 190 combined hits obtained searching these names. This hit count is at least a slight overestimate, inasmuch as it might include such obviously noisy results as would be implied by articles on such things as collective bargaining in Pacific coast fisheries, because of the verb "to troll;" furthermore, none of these programs existed prior to 1968 and the names of certain programs date only to the 1980s or later. Restricting the time period to 1986 through 2003 results in a hit count of 130. Even TSP, which is apparently the individually most mentioned econometric software program (with 397 hits for the period between 1965 and 2003) and one of the longest existing, is subject to false positives, despite being a made-up name, inasmuch as, for instance, a 1967 article on the Theory of the Second Best attracts a hit, possibly because of a misprinted TSP for TSB.

However, economists do not live by econometric software alone. Examining the hit rate for the statistical software packages SAS and SPSS yields 632 hits for SAS and 132 for SPSS for the period from 1975 through 2003. The group that includes EViews, MicroFit, MODLER, PcGive, SAS, SPSS, TROLL, and TSP yields a total of 1,273 hits for the same period, which may imply a tendency for economists to use SAS or SPSS rather than any of the econometric software programs. It has previously been suggested (Renfro, 2004a, b, c, d) that SAS, SPSS and other statistical software packages were used disproportionately by economists during the mainframe era, in part because computer centers often leased software likely to be used by a broad range of disciplines. The fact that beginning in or about 1985, economists for the first time potentially could choose their own software, makes it interesting to consider whether any differences can be discovered between the time before 1985 and since. As it happens, if the search time period is limited to the period from 1985 through 2003, the econometric software packages on their own achieve 430 hits. Adding SAS and SPSS to the collection raises the number of hits to 1,034. Either SAS or SPSS (or both) are mentioned in 630 articles published from 1985 through 2003. Without trying to be too precise, it appears that economists were as likely to mention SAS or SPSS in their articles in the post 1985 period as before that time. From 1975 through 1985, SAS or SPSS achieved 148 hits; from 1986 through 2003 a total of 601 hits. What this finding implies about relative usage, and why it is that economists may have continued to use SAS and SPSS (dis)proportionately just as often since 1985 as before, are potentially interesting questions.

It is additionally potentially informative to combine in the searched population for the period from 1960 through 2003 the articles in the 23 journals currently identified by JSTOR as "statistics" journals. In this case, the number of articles found containing the word "software" increases to 5,619, and those containing "computer program" to 3,023. The term "econometric software," in contrast appears in only 196 articles in the combined population, obviously implying that only 14 articles in this particular "statistics" literature sample contain a mention of this term. On the other hand, "statistical software" achieved 748 hits in the combined sample, nearly eight times the number in the economics journals alone; the first affected article in the statistics literature was published in 1970, in the form of an article on statistical training and research (Patil, Box, & Widen, 1970). In the combined sample, 12 articles mentioned "useful software," but these were mainly in statistical journals; in economics journals articles, only Zvi Griliches and Paul Romer felt this usefulness strongly enough to mention it, other than those economists who apparently used this phrase in connection with the use of computers by Ohio commercial farmers or when dealing with the use of simulation models in the classroom. Anyone with access to JSTOR can easily generate additional results in order to examine the implications of other word combinations and to probe other sample configurations.

Of course, it is also true that even if economists had always been particularly careful to document their computer use in their published articles, the JSTOR results would still only provide a partial result, for the simple reason that the journals included are those that ordinarily publish the work of academic and academically oriented economists, who may also work in government and other research-focused contexts, but generally not that of business economists and others less likely to publish in these journals. Those less likely to publish in such journals nevertheless constitute a significant part of the market for econometric software. Packages like LIMDEP, MicroFit, PcGive, and SHAZAM, on the one side, and AREMOS, AutoBox, FP, MODLER, Stata, and TROLL, on the other, both categorically and individually appeal to distinct user groups and types among economists. For instance, academic economists probably use programs in the first group disproportionately. Some of these users may also have particular geographic characteristics: MicroFit and PcGive, for example, may be more likely to be used in the UK and Europe than in North America. In contrast, AutoBox, with ARIMA as a specialty focus, tends to be most used by business economists and others who would describe themselves as forecasters. AREMOS, FP, MODLER, and TROLL are most likely to be used by those interested in working with macroeconometric models, with FP and TROLL of perhaps greatest interest to those interested in simulations that incorporate the phenomenon of model-consistent expectations. Stata appeals to both academic and business economists, but within the latter grouping not those who primarily consider themselves forecasters.

Actually, remember that JSTOR has a category of business journals, in addition to economics journals, but as discussed earlier these are not mutually exclusive categories, nor is the journal selection in each category such that it would be possible to separate the academic from business economists by journal without making individual selections. For example, taking a simple minded approach, adding the business journals to the economics obtains the result that "econometric software" receives a total of 222 hits. Among the business journals alone, this phrase receives 217. Searching for a mention of any of the terms "EViews, MicroFit, MODLER, PcGive, TROLL" receives 221 hits among the business journals, and 226 when the business and economics journals are combined.

It is also possible to consider the research practices of others. What provides both an interesting and perhaps telling contrast to all the results just described is that if the 83 biological sciences journals in JSTOR are searched just for "SAS" for all its history, 1975 through 2003, there are 11,519 hits. If "SAS" and "SPSS" are jointly searched, for the period from 1975, the number of hits is 14,626. The ACTWT population count is 326,747. Most importantly, once these searches are made, although an article-by-article inspection reveals some false positives, it also reveals that a vastly higher percentage of the time an empirical research article is "hit," the software package being used in the reported research is explicitly identified within it, as compared to the case of econometric software packages and the economics literature. What an article-by-article inspection also reveals is that authors often report not only the software package used but also the specific procedures employed, the software version, and other significant details. Obviously, this quick comparison "test" has its flaws. For instance, it does not reveal how often those who contribute articles to these biological sciences journals omit any identification of the software used in their empirical research, particularly once adjustment is made for this type of article versus any other, or the effect if the full range of packages that potentially could have been used were to be included. Furthermore, no comparison has been made of the number of empirical research articles published in each case. Nevertheless, there is a possible object lesson here for economists.

The implication would seem to be that, among economists, econometric software design is commonly held to lack interest or relevance as a disciplinary topic and, furthermore, that it is generally thought that the use of a particular econometric software package or a particular version, rather than another, is of no material consequence. The conventionally accepted role of econometric software appears to be, as asserted in the introduction to Chap. 1, simply that this software makes operational, in an essentially slavish fashion, the formulae and related results that are to be found in the existing printed literature. As a consequence, certainly throughout the twentieth century, a reader of the economics and econometrics literature might well conclude that the development of this software has had no perceptible effect on the development of economics or econometrics other than to make faster calculation possible, essentially as a result of the speed of the computer as compared to alternative methods.

A possibly revealing sidelight on this particular conception of this software is provided by a short comparative description of a recent version of an econometric software package written by an economics graduate student and teaching assistant. The description was intended to be an evaluation of the salient econometric features of two quite different programs and in its entirety took the form:

> To be honest, I found Y's interface to be very 1990s (and early 1990s at that) and 16-bit retro. The browse features with files in 8.3 with tildes in a land of long file names is pretty old fashioned. Compared to X, the product is very clunky. No right click context menus and a very non-intuitive interface. I really only spent a few hours playing with the software before I went back to X. So I really did not explore very deeply as I found the browsing and non-drag and drop environment a real handicap compared to other products. In my opinion, the product really needs a complete overhaul to make it competitive in today's environment.

The writer exhibits no awareness that it was only the mid 1990s before Windows even began to be the common interface for econometric programs, and that it was only with the introduction of Windows 98 that right-click context menus first arrived as a standard feature – and that, because of its operating system origin, context menus are actually a feature of both programs X and Y, but perhaps not for all program features. This review makes no reference to numerical accuracy or other substantive issues. It is obviously concerned entirely with only certain aspects of the human interface, including an apparently requisite drag and drop environment, in a way that suggests that the earlier discussion in this chapter of macros and other algebraic language matters would be quite a foreign idea to this resident of the "land of long filenames." However, fundamentally, the question that this review raises is why the writer omitted altogether to analyze what each of the programs do, in comparison with each other. Perhaps the assumption the writer made is that below the surface they would be the same functionally, implying a consequent belief that the focus of any econometric software review should simply be the immediate intuitiveness of the interface.

The formation of this type of mindset during the late second half of the twentieth century and the first few years of the next appears to be a consequence of the circumstance that during the earlier years of this 50 + period those econometricians who developed econometric software did so in the context of their own applied research, or perhaps when acting in a research assistant role, using the electronic computer simply as a replacement for earlier computational technologies, hand or electromechanical. At that time, as discussed earlier, what first tended to occur was simply the application of well-known formulae in an essentially straightforward fashion, at least until it became known, essentially beginning in the later 1960s (Longley, 1967), that such an approach can result in highly inaccurate results. Of course, inasmuch as at this point in time the applied economist or econometrician characteristically self-programmed, the modifications required to ensure numerical accuracy could be made silently, with little need to communicate to others either how the particular algorithms used differed in their numeric analytic characteristics from econometric textbook formulae or the nature of any particular modifications. At that stage economists also hardly needed to tell themselves whenever they discovered that previous versions of their software exhibited specific computational faults.

However, by the middle 1980s, because of the widespread adoption of the microcomputer by people who had never programmed, most computer users became instead computationally vicarious, therefore ordinarily not particularly conscious of the specific computational techniques employed and seemingly all too ready to accept at face value whatever their chosen software provided. To be sure, there has always been some level of background recognition that it was possible for programming mistakes to be made, and economists have certainly been aware that such things could happen, but, except in the context of specialist works read almost exclusively by the computationally committed (Belsley, 1991; Belsley, & Kuh, 1986; Belsley et al., 1980), to date there has been a distinct reticence to consider openly this possibility and its implications (McCullough, 1997; McCullough & Vinod, 1999; Renfro, 1997a, b). A willingness to tolerate operational secrecy has long been characteristic of applied economic practice and, notwithstanding certain disquieting revelations from time to time (Dewald, Thursby, & Anderson, 1986; McCullough, Renfro, & Stokes, 2006; McCullough & Vinod, 1999), or occasional avuncular public scoldings (Griliches, 1985, 1994; Leontief, 1971), only recently have the most prestigious economics and econometrics journals, in their article submission requirements, noticeably begun to mandate the more careful reporting of the data used and of relevant computational details (Anderson et al., 2007; Bernanke, 2004; Renfro, 2006). But only a few have yet implemented such policies and even fewer have taken the necessary steps to insure their effectiveness (Anderson, 2006).

Each of the circumstances just mentioned – the possibility of software "bugs," that the formulae that appear in the econometrics literature may not exactly correspond to the specific computations performed by econometric software packages, and that software users and developers are normally different people – are each individually sufficient to establish the need for more public discussion of the characteristics of this software. In combination, they overwhelmingly establish this need. However, it is also true that various, somewhat more general, software design characteristics can also be shown to affect the success with which economists conduct their applied research, as well as the specific numerical results obtained (Renfro, 2004a, b, c, d). But what may have curtailed discussion among economists of the design implications is, as much as anything else, likely to be a general perception of the lack of any feedback effect from the process of software development to the development of either economic or econometric theory. Notice the specific wording here. It is not suggested that there has been no feedback, nor that this has been minimal, but rather that it has not generally been perceived, which may well be the result of the necessity to comprehend first how the design and development of econometric software can affect both applied economic research and the development of economic and econometric theory, before it is possible to perceive either the existence of that effect or its particular magnitude. Omitting to look can itself diminish perception. The Pollyanna problem that such indifference poses is that it is usually folly both to fail to encourage the development of those things that affect well being, yet still to depend upon and expect a successful outcome.

However, the winds of change are picking up. For example, quite recently, an article has been published in *Econometric Theory* (Kristensen & Linton, 2006) that proposes a particular closed form estimator, the need for which is there declared to be a direct consequence of reported, possibly inherent, computational problems encountered by econometric software developers when using standard numerical optimization procedures. This is a potentially interesting example of the type of coordinated investigation that could and should occur as a matter of course. However this degree of recognized symbiosis between the development of econometric theory and econometric software is nonetheless still exceedingly rare, with the development of theoretical econometrics so far seemingly only minimally inspired by workaday computational experience and, in addition, with little attention being paid by theorists to how *best* to implement their work computationally.

Econometric Software Developers

In all the preceeding discussion one actor has so far appeared in what might be perceived to be a shadowey, if not furtive role, namely the econometric software developer. Actually, this is not a dishonorable calling, nor does it require anonymity. There may be little need at this stage to proclaim the names of the culprits, one by one, for these are available in the published compendium and in the manuals and references cited there (Renfro, 2004a, b, c, d), but something about them should be said. The activity is interesting for the types of econometricians it attracts and now includes. It includes economic and econometric journal editors and associate editors. It includes textbook writers, theoreticians, and applied econometricians. Many of its members are academic economists. Others are employed professionally to design and develop econometric software. By motivation, it may be an avocation, an addiction, or even a hobby. But there is also the question, should this group be regarded as including only those who happen to be active today, and not those who have played a part in the past? If its ranks are expanded retrospectively, to include those who during their careers have spent a reasonable amount of time programming, using dyed-in-the-wool computer programming languages, the category of econometric software developer is arguably even quite distinguished in its membership, a secret army consisting of hundreds, perhaps thousands of economists, notwithstanding that most may have given it up upon receipt of their last earned degree. However, as a active grouping, it may not be numerically self-sustaining. This is potentially a troubling idea. But hold this thought for a while, for there is yet more to consider.

Chapter 3
Econometric Diagnostic Tests

It is evident from the previous chapter that more than half a century has passed since economists first began to use the electronic computer. During this time, the modern techniques of econometrics were developed and have evolved, to some degree coincidentally and to some interactively. The formation of econometrics as a sub-discipline of economics and the development of its constituent ideas are the respective subjects of the well-known books by Qin Duo (Qin, 1993) and Mary Morgan (Morgan, 1978) cited in the last chapter, each of which in its own way traces econometrics from its earliest history to its more formal modern development, which has occurred mainly during the period since 1943. Other authors have further elaborated various aspects of this history. The contributions include a history volume edited by Adrian Darnell, a methodological critique by Adrian Darnell and Lynne Evans, a history by Roy Epstein, and a special issue of the *Oxford Economic Papers* edited by Neil de Marchi and Christopher Gilbert (de Marchi, & Gilbert, 1989), as well as a history of macroeconometric model building edited by Ronald Bodkin, Lawrence Klein, and Kanta Marwah (Bodkin, Klein, & Marwah, 1991; Christ 1994; Darnell & Evans 1990; de Marchi, & Gilbert, 1989; Epstein, 1987; Gilbert, 1991; Qin & Gilbert, 2001). In addition, David Hendry and Mary Morgan have compiled a collection of foundation readings (Hendry & Morgan, 1995) and most recently the *Palgrave Handbook of Econometrics* (Mills & Patterson, 2006) has been published. Some years ago, J. J. Thomas, at the end of an article describing the early econometric history of the consumption function suggested that "now is the time for econometricians to write their own history of econometrics, particularly while some of the pioneers are still available to contribute to an accurate account of the early work"(Thomas, 1989, p. 145). His admonition may have been heeded.

This chapter is more specific. As the title indicates, it considers aspects of the development of econometric diagnostic tests and, because the present study apparently constitutes the first attempt to evaluate the particular way in which these tests have been implemented operationally in a generally available form, a brief historical review is also provided. However, both as consequence and characteristic of this brevity, the discussion will be much more sharply focused than in the more comprehensive, but also much more general studies of Darnell and Evans, Epstein, Morgan, and Qin (Darnell, 1994; Darnell & Evans, 1990; Morgan, 1990; Qin, 1993). This review will neither attempt to assess fully nor to examine critically exactly how these

C.G. Renfro, *The Practice of Econometric Theory*, Advanced Studies in Theoretical and Applied Econometrics 44, DOI 10.1007/978-3-540-75571-5_3,

tests have each been incorporated into econometric methodology over the years, nor to set them in this wider context – nor to carry that story forward to the present day. At least initially, the goal is simply to highlight certain historical circumstances and, by so doing, confirm that the procedures for specification selection and evaluation available for use today – with the exception of a few adopted from earlier use by statisticians – first began to be formulated and introduced in or about 1950. This date also happens to separate the estimation of the first macroeconometric models by Tinbergen (Bodkin et al., 1991; Tinbergen, 1939, 1940) and Klein (Barger & Klein, 1954; Klein, 1950) from the first disciplinary use of the stored program electronic computer at the Department of Applied Economics (DAE) in Cambridge. But what is most immediately interesting is the timing of the original introduction of particular diagnostic tests and their first appearance in the textbooks, followed by their implementation in econometric software and subsequent general application. The earliest relevant test statistics, particularly the well-known Durbin–Watson (Durbin & Watson, 1950, 1951) and the von Neumann ratio (Hart, & von Neumann, 1942; von Neumann, 1941, 1942) predate this use of the computer. A characteristic of these and other diagnostic tests is that they made their entry onto the econometric scene quietly.

In retrospect, one of the more striking features of the earliest econometrics textbooks, such as that by Tinbergen (1951), first published in English in 1951, or those of Tintner (1952) and Klein (1953), published in the following 2 years, is just how much they differ from their counterparts today. Reading them now, what is immediately noticeable is the very few and relatively weak evaluative statistics they contain – essentially only standard errors and the R^2 statistic. They do not consider even the Durbin–Watson, which had been introduced in 1950 (Begg & Henry, 1998; Durbin & Watson, 1950, 1951). This omission, not surprising under the circumstances, may at least partially reflect that although this diagnostic statistic ultimately came to be more or less universally adopted, that general use took years to achieve. Its initial rate of spread can be gauged from a full-text search of the combined set of JSTOR economics and statistics journals, which reveals that only 11 articles and reviews published in these journals prior to 1958 contain the phrase "Durbin–Watson," excluding those authored or co-authored by Durbin or Watson.

Other relevant tests began to appear only in 1958, such as the Aitchinson–Silvey that year (Aitchinson, 1962; Aitchinson & Silvey, 1958), the Chow tests in 1960 (Chow, 1960), and the Cox tests in 1961 and 1962 (Cox, 1961, 1962; Darroch & Silvey, 1963). These too were not adopted immediately to any noticeable degree. Neither the Johnston textbook of 1963 (Johnston, 1963) nor that of Goldberger in 1964 (Goldberger, 1964), mention either the Chow or these other tests – except for the Durbin–Watson, which is described in each. Johnston appears to be the first to mention and describe it in a textbook (Zellner, 2006). However, both Goldberger and Johnston, in contrast to earlier textbook authors, include a demonstration of the conditions under which OLS estimators are Best Linear Unbiased, Johnston's particular treatment possibly influenced by an earlier one by Stone (Gilbert, 1991, p. 292). Goldberger and Johnston also each integrate the concept of the analysis of variance. The presentations are generally similar otherwise, possibly because of the authors' earlier mutual contact (Gilbert & Qin, 2006, p. 138). These two textbooks of the early to mid 1960s have aged reasonably well and might be regarded, even today, to be at least early modern in approach.

To mark the introduction of other pertinent tests, it is necessary to fast forward to 1969. In that year Ramsey (1969) proposed his *REgression Specification Error Test* (RESET) and related tests. In 1970, Atkinson proposed a method for discriminating between models (Atkinson, 1970). These tests and the other tests and results cited above are those that in 1972 formed the corpus of the commonly recognized criteria for the within-sample evaluation of econometric specifications (Dhrymes et al., 1972). However, for perspective, it needs to be realized that they were then only potentially available and, what is more, only to the small group of computer-skilled applied economists who in those times directly used the electronic computer and had the capability to implement new procedures. "Potentially" is an important qualification: with the exception of the Durbin–Watson, the R^2 and standard errors, none of these tests or statistics appeared in econometric software packages as a matter of course. In fact, with the exception of these three, it appears that *none* of the (few) then generally available packages included *any* of the tests or statistics that have been cited here.

This last statement is possibly surprising and to assess its validity, in order to give benefit of doubt, remember that 1972 falls within the time period that computer-using economists ordinarily would roll their own software. The characteristic general lack of mention in the economics and econometrics literature of the properties or even the name of the software used in applied economic research, as considered in Chap. 2, makes it impossible to say with certitude that these other tests were not then at least occasionally applied by individual economists, even if it is possible to be skeptical about the extent. Of course, in terms of the ability today to replicate past published results (McCullough, McGeary, & Harrison, 2006a, b; McCullough, Renfro, & Stokes, 2006), it is almost always easier to evaluate aspects of the practices of economists pre 1960, in the dark ages of econometric computation, when authors usually took pains to describe the specific computations performed, than after. However, this opacity of the actual practices in the computer age since 1960 does not call into question the fact of the relative paucity of evaluative tests and statistics in the early days of that decade, quite apart from their degree of implementation and application. As a rule, it is the state of the computational technology at each of the stages of evolution since 1960 that, in the final analysis, is now the best evidence for what could have been done, or not done (as the case may be), rather than the testimony of almost any of the applied research reports published in the literature since then.

Other evidence is far more circumstantial. Consider the continuing tendency of economists to speak as if revealed knowledge immediately transmutes into practice, requiring no more effort than clicking the heels of a pair of red shoes, when in fact there are a number of implementation difficulties. There is some basis to suspect that conscientious applied research application of the known diagnostic tests has been at best occasional, especially as a customary practice. Furthermore, there is anecdotal evidence dating from earlier times that might provide support for this idea. The familiar bon mots of commentators such as Coase, Gilbert, Leamer, and Orcutt, said to apply to econometric practice in 1983 and before, sound plausible enough today as a catalogue of past venial, if not mortal sins. Among the most memorable are:

> "if you torture data long enough, nature will confess,"

> "…the econometrician's [typical] response to these pathological manifestations is to re-specify his equation in some way – to add or subtract variables, change the definition of variables, and so forth – until, eventually, he gets an equation which has all correct signs, statistically significant coefficients, and a Durbin–Watson statistic of around 2, a relatively high R^2, and so forth,"

> "In order to draw inferences from data as described by econometric texts, it is necessary to make whimsical assumptions," and

> "doing econometrics is like trying to learn the laws of electricity by playing the radio"

However, considered in terms of the rules of evidence, in a court of law these charges would no doubt be judged as being no more than hearsay, even if on the basis of them it seems convincing to characterize specification selection during those wild and woolly times as involving the heuristic selection of the "best" from among a multiplicity of computer-produced alternatives using dubious criteria.

To anyone young who today surreptitiously reads Leamer's reformist admonitions (Leamer, 1983) – especially in the original near-brown paper covers – it might seem that the almost apocryphal 1960s and 1970s represent a turned-on time of promiscuous econometric excess, a heady time of guiltless and heedless indulgence. But, actually, even in 1972, if not before, economists had already tasted of the fruit of the tree of knowledge and had recognized nakedness. There was then a general consciousness that economists of that era concentrated "on the estimation of many parameters, and the pure testing of relatively few hypotheses" (Dhrymes et al., 1972), which was already a source of some discomfort to thoughtful econometricians. Haavelmo's words, quoted in Chap. 2, asserting the need to test were actually voiced 15 years earlier in his 1957 presidential address to the Econometric Society in Philadelphia (Haavelmo, 1958), and indicate clearly that, essentially as a matter of course, econometricians already accepted the need for misspecification tests, in principle, if not practice.

It may or may not be sufficiently exculpatory to suggest that, in 1972, the typical applied economist practitioner was a member of that generation that came of age just as the computer was beginning to replace pencil and paper calculation, the abacus, and the desktop calculator, when national income accounts and a broad range of other economic data were for the first time becoming available for the asking, and the spirit of possibility and discovery hung in the air. However, it is also true that candidate specifications derived from "data mining" exercises were, even then, sometimes (perhaps even often) regarded as simply hypotheses, made to be evaluated subsequently by later out-of-sample testing, rather than held out as tested postulates. In those years, initial evaluative perfection was not necessarily expected, in part perhaps because of a sense that disciplinary progress was being made – even if much of the focus was on properties of estimators. Reflecting this emphasis, the relatively few extant econometric software packages tended to differentiate themselves from others simply on the basis of the set of estimators provided and other such features, not in terms of their capacity to provide the means to exhaustively test hypotheses. Was this tendency then a matter of not quite grasping exactly how both to implement and use these tests effectively? It is now hard to tell.

However, taking into consideration the rhetoric, finger pointing, and anguished brow beating that has gone on since about applied econometric research practices, sometimes from a rather detached and remote Olympian theoretical perspective (Ericsson & Irons, 1995), it is salutary to recollect, as indicated earlier, just how thin was the operationally available crop of evaluative tests in 1972. Viewed retrospectively, the need at that time for further development is now overwhelmingly obvious. However, only in the later 1970s, following most notably from the work of Sargan and his colleagues and students at the London School of Economics (Desai, Hendry, & Mizon, 1997; Hendry, 2003b), did either theoretical or applied econometricians begin to focus intensively on the issues that are now seen to be involved in the specification search process in a dynamic context (Davidson & MacKinnon, 1981; Dhrymes et al., 1972; Harvey, 1981; Krolzig & Hendry, 2001; Leamer, 1978; Pagan, 1990; Wickens & Pagan, 1989). Moreover, with certain important exceptions, there was then also a further significant time lag before those who wrote econometric software packages commonly began to incorporate other than the most basic supporting evaluative statistics into their packages, such as the R^2 (in both adjusted and unadjusted forms), the F-test, the residual variance, the Standard Error of Estimate and the Durbin–Watson (Renfro, 1997a, b). The situation calls to mind the adage, bricks without straw. Indeed, it was not until the 1990s (Renfro, 1997a, b) that it became usual for tests of disturbance properties, such as the Breusch-Godfrey, Box-Pierce, Ljung-Box, Jarque-Bera, and tests of structural stability and functional form, such as the Chow tests and the Ramsey RESET, began to be commonly included in econometric software packages, not to mention heteroscedasticity tests, which until recent times were themselves often seen to be almost solely cross-section related.

The inference consequently to be drawn is that the active operational acceptance of these evaluative tests, as well as the formation of any practitioner consensus on the appropriate criteria for their selection, has been slow to happen. Notice that characteristically two separate lag processes operated: there was, first, an appreciable time lag between original introduction and the first mention in an econometrics textbook. Then there was an additional, often also appreciable lag before software implementation. This inertia had several causes. On the one hand, as the above account to a degree indicates, the original creation of individual tests has historically occurred in what might almost be seen as random fashion, the order of appearance often reflecting the particular, even idiosyncratic, interests of individual econometricians. During this creation process, the contextual operational role of each of the particular tests has not always been immediately widely perceived, even if the individual purpose of a given test might have been well understood – or if perceived, that perception has not always led to action.

Of course, there are always exceptions to a rule. It is true that, once admitted into the textbooks, the Durbin–Watson statistic was almost immediately adopted by econometric software developers, which is perhaps an indication of its seemingly obvious applicability, as well as the original scarcity of other useful tests, but this adoption also reflects that the general development of software began well after this statistic had already begun to achieve iconic status. In contrast, and progressively, each new statistic has increasingly had to fight for its place among the existing crowd, with the rules of selection still not settled. How these rules might be appropriately formulated remains less than obvious: it is too simplistic to imagine the individual selection battles as being won or lost simply on the Darwinian basis of survival of the fittest, if for no other reason than that the context of use is often what matters most. It also the case that the end users, even now, are themselves not necessarily entirely convinced that the game so far has been worth the effort – including today those irrepressible scofflaws, the calibrationists, who appear to spurn outright standard econometric conventions.

During the time since 1960, as discussed in the two previous chapters, there has also been a change in the logic of the econometric software development process, which can be seen to be a consequence of the wider use of the computer, and perhaps to a degree the result of specialization as well, but more fundamentally reflects the skills required. As discussed earlier, software development originally occurred as self-creation, to suit the personal needs, practices, and beliefs of the individual investigator. Latterly, packages have commonly been developed to suit the perceived needs of a relatively wide audience. This tailoring can be seen to be the modern motivation irrespective of whether or not a particular program is classed as freeware or fee licensed. In either case, it is the audience that in the end conditions the performance, individuals voting with their feet if not their money. The theorist conceives and the developer provides, but it sometimes appears to be true that the typical user of these software packages – even excluding outright heretics – ignores all but a few of the included test statistics, perhaps not quite grasping the connections between each of them – nor their relative strengths and weaknesses as indicators of various circumstances and conditions. In fact, it is not entirely unknown for some who use these programs to ask how to hide the bulk of these numbers, the effect in their opinion being simply to clutter up the screen display. Such resistance has had an effect, for tests are quite often provided as options, rather than displayed by default. Unlike the theoretical econometrician, who may be able to adopt the lofty pose of a seeker of truth, the colleague in the trenches who chooses to develop software for the general use of the profession finds justification by that use and may need to bow in recognition that who pays the piper picks the tune. “Eat your spinach” is not always welcomed.

Nevertheless, it is by a combination of successful *implementation and use* that any given procedure operationally becomes state of the art. Not simply by pencil and paper are econometric techniques validated, but rather by the discovery of an apposite applied research context for their use (Granger, 1992). Implementation over the years has involved the need to make compatible selections, but this is an effort that has also to some degree been frustrated and retarded by the very development of econometrics as a subject, as well as by the changing dictates of taste and fashion. Although there has long been a tendency for economists to speak glowingly of the computer and its capabilities, almost as a requisite encomium and almost as if computer implementation is effortless, until about 1975 an initial retardant was the simple reality, discussed in Chap. 2, that the creation of econometric software packages was at first focused mainly on the development of algorithms to implement parameter estimators, organize the data input process, perform transformations, and other equally workaday issues – and only minimally on the calculation of evaluative test statistics (Renfro, 2004a, c). Of course, prior to the use of the computer, the primordial need to make calculations with desktop calculators, or even pencil and paper, stood in the way of any enthusiasm to implement more than a few basic test statistics (McCullough, McGeary, et al., 2006a, b; McCullough, 2006).

Preachers of hell-fire and damnation can rant and rave as they wish about econometric practices, but the mores of practitioners at each point in time are ordinarily more conditioned by three things: their laboratory training during studenthood, or lack thereof; the particulars of any requirements to provide as a part of journal submissions complete details about the data and software used, or not; and the capacity of the available software to provide the necessary tools of research, as best understood at each moment. Of course, in addition, the inherent difficulty faced by economists to make sense of observed economic processes, to fathom the properties of what is now often called the data generating process, must of course also be recognized. Among these considerations, the historical development of econometric computing, beginning essentially from a standing start, may also help to explain why it was not really until the early 1970s that econometricians even began to focus specifically on the choice of the appropriate set of evaluative statistics (Dhrymes et al., 1972). It may help to explain why econometric software developers only later began to implement any appreciable number of these. Recall once again that, in the early 1970s, the user and developer of the software were commonly to be found under the same roof. Furthermore, for publication, there was not then nearly the same degree of referee pressure for an author to provide a plethora of misspecification statistics.

Under more ideal circumstances, econometric software might have been developed from the beginning so as to provide a complete and coordinated set of facilities, as could, perhaps, have been originally defined by a careful survey of the theoretical literature. However, instead, this development has been rough and tumble evolutionary. It is almost as if, initially, a tacit agreement (possibly a bargain with the devil) was reached between the programmer and users whereby the software included a serviceable, casually agreed, quite basic set of techniques, and users in turn chose to ignore what was missing. Seen retrospectively, the software developer, pressed for time, offered as gold the copper and iron pyrites hard-scrabbled by the user from the machine. Of course, this is a simplified, somewhat whimsical retelling: the issue was not even then just a matter of the developer making a choice between one more estimation technique versus an additional evaluative test, but instead involved making competitive choices among a range of possible computational facilities that each took time to develop, in many cases for the first time. It was actually a choice between substitutes in a resource-constrained world.

However, recall also that what soon occurred, beginning in the later 1970s – and which initially had the effect of further obscuring this question of software content – is that the entire point and purpose of econometrics became subject to debate (Diebold, 1998; Gilbert, 1986; Hall, 1995; Leamer, 1983; Pagan, 1990; Sargent & Sims, 1977; Sims, 1980 Summers, 1991; White, Engle, & Granger 1999; Wickens & Pagan, 1989), perhaps in part because of prior implementation difficulties, albeit then unrecognized as a contributing factor. In any case, throughout the 1980s, one of the effects of the econometric and macroeconometric modeling debates (Hall, 1995) instigated by Leamer, Lucas, Sargent, and Sims, not to mention the particular necessitating effects of the work of Engle, Granger, Hendry, Pesaran, Sargan, and others, was both to broaden and in certain respects deepen the implied software requirements. Not only has there been a broadening and deepening within the context of econometrics, as its characteristics might have been defined in the middle 1970s, but in addition its techniques have been applied to support a wider range of applications, among them financial theoretic applications, microeconometric problems, and in recent years increasingly a variety of nonlinear simultaneous equation solution methods.

The collateral effect then and since has been a tendency to impose computational demands much beyond the immediate capability to respond of the relatively small group of econometricians who develop software, especially in the context of the microcomputer revolution that was under way at essentially the same time. It is a general programming truism that 90% of the job can often be done quickly, but that the last perfecting 10% may take many times as long. The implication of course is that the qualitative results obtained are not always of the same magnitude as the quantitative results, a dangerous combination in the context of any demonstrated tendency for users to pay little attention to the numerical accuracy of those results (Brooks, Burke, & Persand, 2001; McCullough & Renfro, 1998, 2000; McCullough & Vinod, 1999, 2003; McCullough & Wilson, 2002, 2005; McCullough, McGeary, et al., 2006a, b; McCullough, 2006; Summers, 1991). These home truths, although possibly hard-to-swallow, are admittedly somewhat tangential to the matter of the present study, but they need to be voiced if only to call attention to the fact that the number of techniques that have been implemented in software during the past 25 years is quite substantial, and involves so far some degree of papering over of the cracks. To have implemented these techniques thoughtfully, accurately, and fully, even in the absence of the hardware and operating system changes that have occurred, would have seriously overtaxed the capabilities of econometric software developers and has not so far occurred.

Does it therefore follow that situation alone is to blame? Forgetting any attempt to be fair-minded, a question that might instead be asked in summary is who is the appropriate scapegoat? Historically, it was the applied economist, cast as the practitioner, who was commonly taken to the woodshed for the way in which applied economic research was conducted, especially during the 1980s. However, it has just been argued that then – and more recently as well – the software "tools" operationally available actually did not provide the capability to perform more than a few, relatively weak diagnostic tests. This defense might be elaborated further by arguing that, in many cases, this practitioner made a good attempt to obtain the best result, given the established environment. Taking this argument into account tentatively, it might then seem that the truly guilty party is the econometric software developer, who bears the opprobrium for not having provided the right software notwithstanding the capacity to provide it. After all, the applied economist did not willfully refuse to apply a well-conceived and developed research tool and, apparently, there were well-known diagnostic tests that could have been implemented even as early as the beginning of the 1970s. The only mitigating defense available for this not-so-harmless-drudge, the econometric software developer, appears to be that the technological environment that now exists did not exist much before the start of the present century – but are the Internet and the graphical user interface that important – or other recent hardware improvements? Or is it that econometrics, as a set of methodological precepts, was in a sufficient state of flux that only in the past 10 years has there been the capacity to determine what should be done? Of course, if this argument is implausible, there is always the economic statistician, who might be assigned heaping blame for the quality of the data that has been supplied. Last, but not least, the econometric theoretician might be seen as the real culprit, the one who created the confusion in the first place.

Diagnostic Tests: A Division of Responsibilities

However, to ascribe blame in this way, even jestingly, is to miss the point that the applied economic research process should today be organized in a way that defines each participant's possible contribution and particular role in a way that is clear to all. That individual participants may lack sufficient information about specific computational aspects was considered during the discussion in Chap. 2, implying also the general need for greater information flow. But, it is additionally true that both applied economists and econometric theoreticians have, for the most part, left econometric software developers to their own caprices. From the early 1960s until the 1990s, applied economists seldom spontaneously exerted any pressure to improve the misspecification testing or other evaluative facilities of the software. Similarly, although individual econometric theorists have been known to suggest occasionally the desirability of including certain test statistics they have formulated, such direct admonitions are rare, even if they frequently occur implicitly as the result of published articles. Most software changes have occurred endogenously, *not* as a consequence of careful consideration by economists generally, even if certain software packages, such as REG-X, originated in order to make it more generally possible to apply particular methodologies, in this case the LSE method (Renfro, 2004a, c, pp. 382–383). But that a widespread examination of these issues – although desirable – has not yet occurred can be verified easily by even the most casual search of the journal literature, as well as by the absence of discussion there of the desirable properties of such software.

Of course, econometric theorists, by focusing their attention on the development of misspecification and other evaluative tests, have served to delineate the frontiers, but this effort on its own arguably had minimal operational significance during almost the whole of the second half of the twentieth century, reflecting the necessity for these tests first to be widely implemented and then used. Is it possible to discover and describe why in relatively recent times this implementation began to increasingly occur even if it might be harder to determine the degree of use? Possibly providing a parallel example, Gilbert and Qin have recently recounted (Gilbert & Qin, 2006, p. 130ff) how the Keynes–Tinbergen debate helped to stimulate Koopmans "to further clarify the logic of model specification procedures" by adapting them to Frisch's structural approach and asserting that the model should be regarded as a "general working hypothesis," in order that the "... Specification should then cover all the additional information required to allow conditional statistical inferences to be drawn." However, a supporting rationale was necessary and it was Haavelmo who "offered a forceful justification of probability theory and the stochastic specification of structural models from both the conceptual and operational aspects." Haavelmo's ideas were taken up and then extended by the Cowles Commission scholars.

The aspect of this historical process that especially merits consideration is the light it might shed upon the important change in mindset that occurred between the middle 1950s and today as regards misspecification testing. To begin, consider the mindset that is evident in the 1957 contribution by Theil (1957) that represents an early attempt to test specification errors and the estimation of economic relationships. He begins by stating that

> The usual first step in the estimation and testing procedures of economic relationships is the formulation of a 'maintained hypothesis.' As is well-known, this amounts to a specification of the general framework within which the estimation or testing is carried out. Most of the results in this field have been derived under the assumption that the 'maintained hypothesis' is correct. This is indeed in accordance with its name: the hypothesis is not subject to test, but 'maintained.'

He then indicates that his purpose is to analyze how to evaluate if the "maintained hypothesis" is erroneous. He sees his argument as a "generalization" (p. 41) of his earlier approach in the context of his monograph on aggregation problems (Theil, 1954). The particulars of Theil's argument in this specific article do not need to be considered now. What is most pertinent and particularly noticeable after 50 years is the absence of any reference to any misspecification tests, as this term might be interpreted today, but in addition – and more important presently – there is also an absence of an organized operational approach to the analysis of the problem addressed.

Possibly the first clear example of the introduction of a methodology designed to address the form of a specification occurred in the context of the contribution of Box & Jenkins, (1984), originally in 1970, with its emphasis on the determination of the "transfer function" of the system – "the determination of a dynamic input–output model that can show the effect on the output of a system subject to inertia, of any given series of inputs" (p. 1) – by a process that involves, among other things, an evaluation of the autocorrelation function of the process in order to "identify" the model. Box and Jenkins did not consider the model to have any necessary "economic" content and the issue is not at present the degree to which this approach should be regarded as "econometric" in nature. For the moment, the relevant consideration is the attempt to form an operative "model" by what can be perceived to be a testing process that involves aspects of specification and misspecification.

In contrast, if the focus is shifted to economic specifications, an appropriate modern starting point might be the words of Denis Sargan that have been quoted by others before (Johnston & DiNardo, 1997):

> Despite the problems associated with "data mining" I consider that a suggested specification should be tested in all possible ways, and only those specifications which survive and correspond to a reasonable economic model should be used.

The so-called LSE approach that is essentially founded on these ideas and is embodied in the ongoing work of a number of Sargan's students, which dates back to the mid to late 1970s, has been referred to earlier. Of course, this methodology has not been universally adopted, but, as mentioned, the progressive adoption of the idea of testing intensively as a principle has nevertheless caused the development of many of the time series-related evaluative tests found in econometric software packages today (Hendry, 1995). The general outlines of this development were clearly visible in the literature as early as the beginning of the 1980s (Harvey, 1981), more or less at the time of the introduction of the IBM Personal Computer, but it is also evident that, although the need for careful testing may now be widely agreed, in detail this approach has since been adapted in different ways by different participants. In addition, there is still the need to examine whether the research tools now available are sufficiently well fabricated.

It is possible to regard Sargan's point of view and much of the associated development of econometrics since as being sympathetic with and involving the refinement of earlier conceptual developments, proceeding from and based upon the work of Frisch, Haavelmo, Koopmans, Tinbergen, others affiliated with the Cowles commission, and yet others who at that time might have generally subscribed to the idea voiced by Karl Fox (Fox, 1956) that, in those early days of applied econometrics, there were among economists "sharp differences of opinion about how far one should go towards formalizing the concept of interrelatedness," and, in addition, that in 1956 there remained a still tenaciously held traditional view "that all economic relationships should be dealt with on an intuitive level – that no tangible mechanism should intervene between the raw material (individual time series) and the finished product (policy recommendations)." Standing against this conservatism the development of econometrics involved the hope and promotion of a "newer view, still struggling for recognition, ... that the policy implications of a host of raw time series can be made clear if they are organized into an econometric model – a system of equations which translates the concept of interrelatedness into an explicit, quantitative, reproducible form" (p. 128).

In contrast, there is today another point of view that at least at first sight appears frankly antagonistic to the serious consideration of formal statistical tests. For instance, Lawrence Summers (1991, pp. 129–130) has argued that:

> ...formal empirical work which, to use Sargent's phrase, tries to "take models seriously econometrically" has had almost no influence on serious thinking about substantive as opposed to methodological questions. Instead, the only empirical research that has influenced thinking about substantive questions has been based on methodological principles directly opposed to those that have become fashionable in recent years. Successful empirical research has been characterized by attempts to gauge the strength of associations rather than

> to estimate structural parameters, verbal characterizations of how causal relations might operate rather than explicit mathematical models, and the skillful use of carefully chosen natural experiments rather than sophisticated statistical techniques to achieve identification.

although he adds the important qualification (p. 130) that:

> ...my quarrel is not with the goal of estimating deep parameters and seriously evaluating precisely formulated hypotheses but with its feasibility. Attempts to make empirical work take on too many of the trappings of science render it uninformative.

These quotations appear in the introductory section of Summers' paper and somewhat convey its tone, but a close reading of the entire argument reveals that much of what he has to say on this topic is addressed to the separation that he perceives to exist between theoretical and applied economists. Even more pertinent to present concerns, Summers puts forward an argument that is strongly in favor of informative statistical tests; it is against *uninformative* tests. This argument is quite relevant to any consideration of the relative roles of the econometric theorist and those who develop econometric software, particularly when these roles are considered in the context of an examination of how the historically developed econometric specification tests have been utilized in the software that is today commonly employed by applied economists.

When the work of econometric theorists is considered specifically, it quickly becomes apparent that the historical focus has most often been the individual test, somewhat in isolation, both in terms of separateness from real world application and from tests of other aspects of a given specification, in much the same fashion as the parable of the three blind men and the elephant (Harvey, 1990, Chap. 5). However, this characterization may be less generally accurate today: there may now be a greater, and even still growing, appreciation among theorists of the defects of the so-called "one directional" test and of the consequent need to harmonize the battery of tests that might be applied, born from a higher degree of recognition that tests formulated to signal the presence of one pathology may not always be robust in the presence of another (Bera & Jarque, 1982; Davidson & MacKinnon, 1985, 1993; Hausman, 1978; MacKinnon, 1989; Plosser, Schwert, & White, 1982; Saikkonnen, 1989; Thursby, 1989). For instance, in the context of their consideration of the implementation of misspecification tests individually and collectively, Eastwood & Godfrey (1992) demonstrate awareness both of the range of tests in MicroFit, PcGive, and SHAZAM and that the interpretation of test outcomes often poses a problem for users. As an example, on the basis of earlier work by Consigliere (1981), they point out that a linear specification in which the parameters are constant but the disturbances are autocorrelated may cause one or more of the Chow tests to reject the hypothesis of parameter constancy; conversely, in other cases, they report that parameter variation may result in (apparently) correlated residuals. An obvious question that such circumstances raise is whether sufficiently robust individual tests can be formulated or if what is most often instead required are an entirely new battery of consciously harmonized tests?

A possibly less obvious question is whether collaboration between theorists and econometricians who develop software might help to shed light on the various theoretical and practical aspects of this problem. When considering practice, a healthy dose of empiricism never goes amiss, for as mentioned before there is always at least a degree of temptation for theorists in isolation to shape problems to fit solutions, rather than to grapple with practitioners' actual needs. Faced with limited practitioner feedback, and with little information otherwise available about practices, it is all too easy to embrace over-tightly the concept of asymptotic equivalence and also to consider too readily an idealized finite or small sample context of application. And, of course, not all theorists exhibit – or may themselves have – a conscious awareness of potential interactive effects. Because of the concept of asymptotic equivalence, some seem inclined, at least on occasion (Kiviet, 1986), to view the classic methods of forming tests statistics – including Lagrange Multiplier, likelihood-ratio, and Wald – as yielding a bountiful cornucopia of alternative test statistics with contextually desirable properties, that in addition have the protean quality of being able to be modified in a variety of ways so as

> to generate new tests which retain the original asymptotic properties, but may have improved small sample properties. The practitioner is thus faced with a proliferation of tests for the same null and alternative hypotheses. [However, b]ecause these alternative test statistics may have different power functions and different true significance levels in small samples, they may cause conflicting statistical inference and consequently confuse model builders.

On a literal reading, the principal problem would appear to be only that of joyous excess, an exciting prospect for practitioners, even if the possibility exists for conflicting statistical inference. Notwithstanding a possible sorcerer's apprentice effect, almost any practitioner, once made aware of this potential seemingly profusion, is sure to have both appetite whetted and hopes raised in the anticipation that, with luck, he or she might one day wake up in the midst of what could be a veritable econometric Lego village.

However, unless "practitioner" is applied restrictively to those few remaining applied economists who invariably write their own programs, including the code for the individual constituent algorithms, this village may be found to have certain, almost potemkin qualities. As argued earlier, in the case of most "practitioners" – in the more usual, much wider sense of the word – intermediation is almost always necessary, in the form of some combination of textbook authors and software developers, for only few who would not describe themselves as econometricians venture to read econometrics journals. But even in those cases that the prototypical practitioner might hazard his or her chances, and wander into the library or else download a pertinent article from the Internet, there are still problems of communication. In much the same way as economists are sometimes still given to writing about the "firm" as if it were invariably managed by an owner–entrepreneur, theoretical econometricians sometimes adopt one of two expository styles that effectively place a distance between themselves and the world of application. The first is an ex cathedra style that supposes the reader to be a cap-in-hand supplicant, a practitioner who is hanging on each word in the hope of receiving wisdom, and who often needs to have the results communicated informally in order to know how to apply them. But seldom do the papers that adopt this style provide the specific cookbook details that are required – certainly not at the level of universality that implementation in software almost inevitably mandates. The other is more entre nous, supposing the reader to be an econometrician, and in this case even the idea of implementation may be treated as if somewhat beside the point. In this case any subsequent attempt to implement will then first require a careful, possibly quite expert, even variant interpretation of what the theorist might have meant. Many mathematical proofs are notoriously non-constructive in nature, as was discussed in Chap. 1.

What is actually needed is a blend of skill sets. The pretense that the practitioner – or his or her interlocutor, the software developer – sits at the knee of the theorist-as-savant is not particularly useful, for in many cases successful implementation of a new technique or diagnostic test will require computation skills and numerical analysis knowledge the theorist does not necessarily possess, not to mention that the implementation process may expose particular aspects that he or she has either not considered or else discounted. Alternative formulations of test statistics may be asymptotically equivalent, but (as will be demonstrated later) the finite sample forms can differ in important ways that might affect inference; in any case, different software packages may present users with identically named statistics that exhibit distinctly different values, as already demonstrated. More generally, programming – because of the need for completeness, as discussed in Chap. 2 – often requires a level of close attention to detail that may not be perceived in advance. The computer is a hard taskmaster, so that the implementation process should be both collaborative and iterative, and, in the twenty first century, everyone involved needs to accept the modern realities, among them that intermediation is here to stay. An apparent present effect is often to restrict and somewhat standardize the workaday practitioner's possible choices: even casual study of the current crop of econometrics textbooks reveals that only in a limited sense is there actually a "proliferation of tests" on display. The issue is both the variety of distinct test statistics and the tendency for tests to be implemented in the form of a given test statistic, when in contrast the theoretician might argue that a given test may quite appropriately be associated with a range of statistics. An aspect of this situation may be a need for the practitioner to become progressively better educated concerning the concepts associated with a "test," raising the question of how best to provide this education?

As will be demonstrated, the existing econometric software packages each, or even collectively, offer as built in defaults or options only a limited subset of the tests that have been considered in the literature, particularly so if this set of tests is defined to include all the variants that have been investigated. Moreover, it may be necessary that this restriction of the econometric playing field should occur, lest the practitioner be overwhelmed by the specialist knowledge required to make the choice between tests and procedures, some of which may not yet be "ready for prime time." The question is, is it possible for these intermediaries to make an appropriate selection from among all relevant tests in such a way so as to not, by prior selection, mislead the timorous practitioner, but at the same time, by offering variants, not confuse? Furthermore, is this transmission process likely to be bidirectionally efficient, both in making available to practitioners the best work of the econometric theorists and at the same time providing theorists with a good sense of the immediate and future needs of practitioners? These are not questions that can yet be answered, for at this point in time we do not yet know in full detail either what is available to this still slightly mythical "practitioner" or, of what is available, what is actually used. In addition, it is also necessary to recognize what is implicit in any such consideration; in particular, econometrics is today a multi-faceted arena, not a circumscribed specialty. A close look at any modern textbook reveals a degree of disjointedness not present in earlier years. Such questions also occur as, whether it is possible to design an econometric software package to be used both for time series and cross-section applications? That is, in what ways might the tests applied in each case differ? Only to a degree is there an obvious answer.

Realism Versus Instrumentalism

Taking an overview, Tony Lawson has characterized the development of econometrics as historically involving two opposing points of view, *realism* and *instrumentalism* (Lawson, 1989). His purpose was to explore whether this type of categorization can be regarded as bearing "at all upon recurring themes in econometric analysis." The first of these labels is generally associated with the "doctrine [that] there exists a material and social world independent of (any individual) consciousness but which is knowable by consciousness – it is accepted that true theories can be obtained... and... that such knowledge – or true theories – should be pursued" (p. 238). In contrast, Lawson identifies Instrumentalism as the "methodological doctrine that predictively successful theories is all that is required." Although as stated here these definitions essentially represent polar philosophical positions that may not correspond to any given econometrician's deeply held point of view, it is nonetheless clearly possible to identify the Cowles Commission methodology, and structural econometric modeling more generally, with realism and, in contrast, statistical time series modeling with instrumentalism. However, Lawson's argument goes beyond this type of categorization. It is also made with reference to what might be called "hard core" realism, the point of view that originally raised a fundamental question concerning the historical application of probabilistic methods to the representation of empirical economic phenomena. In his consideration of this position, Lawson not only raises such questions as how to evaluate, in the context of the 1930s and 1940s, Keynes' ideas or Haavelmo's probabilistic approach, but also the relative positions in the later 1940s of Koopmans and Vining in the "Measurement Without Theory" debate (Koopmans, 1947; Vining, 1949), pitting structural modeling versus the Burns and Mitchell business cycle measurement methodology, a tension that somewhat extenuated still exists. The modern opposition of the structuralist and the statistical time series approach implies the need to consider still the degree to which inferences about specifications can be considered in terms of a given, common set of statistical tests and criteria – and of course there is as well the classical/Bayesian split, among other possible divergences.

It is certainly characteristic of the modern economist, since the beginning of the twentieth century, to wish to analyze and explain economic phenomena in a way that permits some type of policy formulation. From this comes the need to make inferences and also to determine which of these are valid. Mitchell's work on business cycles was not simply a passive research program, but rather appears to have been driven explicitly by a desire to discover how to ameliorate the amplitude of cycles. Similarly Keynes, notwithstanding his famous response to the Tinbergen model, was a notably policy-active economist, involved throughout his career in what might be termed "back of the envelope" model building, but in a way that provided quite definite prescriptions. Of course, these economists are simply examples from relatively recent times. The idea that the role of the economist is to make sense of empirical economic phenomena towards an end can be traced back as far as William Petty, who declared in his *Political Arithmetick* in 1690, that his goal was to "express [himself] in terms of *number*, *weight,* or *measure*; to use only arguments of sense, and to consider only such causes as have visible foundations in nature." In the eighteenth century, Quesnay carried this idea further as a basis for analysis, with his Tableau Economique. In modern times, Leontief inherited this mantle. It is evident from the fact that he placed Quesnay's words:

> Les sciences qui admettent le calcul ont donc la même base de certitude que les autres. Cette certitude, il est vrai, peut s'étendre par le calcul sur les quantités qui ne peuvent être supputées que par le calcul, et dans ce cases il est toujours en lui-même essentiellement infallible, c'est-à-dire qu'il presente toujours infailliblement et conséquemment ou des erreurs ou des réalités, selon qu'on l'applique à des réalités ou a des erreurs. D'où suit que, dans la recherché de la vérité par le calcul, toute la certitude est dans l'évidence des données.

prominently on the page opposite the Introduction of *The Structure of the American Economy*, as well as later direct testimony, that he aspired to improve significantly upon Quesnay's work as an embodiment of economic theory. He declared this volume to describe "an attempt to apply the economic theory of general equilibrium – or better, general interdependence – to an empirical study of interrelations among the different parts of a national economy as revealed through covariations of prices, outputs, investments, and incomes" (Leontief, 1941, p. 3).

However, not only do economists have predilections, possibly as delineated by Lawson, but they also are influenced by their historical technological context. For instance, even Alfred Marshall – although perhaps unlikely to approve of the modern probabilistic approach that almost any type of modern econometric representation embodies – nonetheless recognized the need for and role of quantification. In 1907, almost exactly a century ago, Marshall (1907) summarily assessed the development of economics during the nineteenth century as having achieved "general agreement as to the characters and directions of the changes which various economic forces tend to produce," but he was also quick to indicate that "much less progress has indeed been made towards the quantitative determination of the relative strength of different economic forces. That higher and more difficult task must wait upon the slow growth of thorough realistic statistics" (pp. 7–8). These words can be considered independently of their time, but they gain force when placed in their historical and technological context. At the moment that Marshall spoke he would have known of the existence of automobiles and early airplanes, each perhaps by direct experience, if unlikely to be able to gauge well their subsequent development and full impact. He would have known about and almost certainly have used both the telephone and the telegraph, at least to a degree, but would have known nothing of radio nor of television. He is likely to have known something about the punched card information processing technology associated with Herman Hollerith and his tabulating machines, first used in the 1890 US population census, as described in Chap. 1, but it was of course only in 1924 that Hollerith's Tabulating Machine Company merged with two others to become what is now known as IBM. However, even this development only distantly presaged the ability to organize and employ economic statistics effectively: as recounted, it was not until 1943 that the first electronic computer began to be successfully created and not until 1945 that the design of the stored program electronic computer was first conceived. It was even later that such a machine began to be used by economists or, for that matter, was made by IBM. Of course, the larger forces that established the overall twentieth century technological environment were also unknown to Marshall in 1907. He may have sensed something of the possible future impact of the social and political developments of his time, but he did not then know of the First World War nor of its catalytic impact on the development of the economic statistics of which he spoke. He knew nothing of the second war to follow and the combined effects of both these conflicts to accelerate the development of all these technologies. Nevertheless, the very fact that Marshall thought to mention the relative backwardness of the "quantitative determination of the relative strength of different economic forces" certainly demonstrates something about his mindset as an economist.

Of course, the present context suggests realism as a leaning: to be concerned about the variety and characteristics of misspecification tests provided by the existing econometric software packages implies prime facie a realistic turn of mind. But it is also necessary to be careful not to prematurely categorize. As indicated, Lawson discusses briefly the Koopmans–Vining debate of the late 1940s (Lawson, 1989, p. 253), sparked by Koopmans' review of Burns and Mitchell's *Measuring Business Cycles* (Burns & Mitchell, 1946), that Lawson suggests was cast as a debate about "theories and models, the conditions of prediction, and the grounds for adopting a probability approach in econometrics," and makes the point that this debate represented the opposition of two analytic positions, one that focused on the behavior of economic agents versus the Burns and Mitchell approach that took "cyclical behavior per se as a unit of analysis." Yet, as will be argued later, it is also possible to regard Mitchell's work as being conditioned and bounded by the technological circumstances in which it first began in or about 1912. Koopmans, in contrast, reflected in his point of view both his education as a physicist and the later time at which that occurred.

Approximately 20 years thereafter, in 1970, Leontief chose the occasion of his AEA presidential address (Leontief, 1971) to consider the topic "Theoretical Assumptions and Nonobserved Facts." However, it is significant that he argued not that economists were too otherworldly, choosing to investigate irrelevant impractical problems, but rather that economists tend to be ineffectual, decrying the "palpable inadequacy of the scientific means with which they try to solve" (p. 1) what are in fact quite worldly problems. In this context, Leontief also quoted from Frank Hahn's Econometric Society presidential address (Hahn, 1970, p. 1): "...the achievements of economic theory in the last two decades are both impressive and in many ways beautiful. But it cannot be denied that there is something scandalous in the spectacle of so many people refining the analysis of economic states which they give no reason to suppose will ever, or have ever, come about... It is an unsatisfactory and slightly dishonest state of affairs." Ten years later, in 1980, Christopher Sims (Sims, 1980) more specifically attacked the use of macroeconometric models as being fundamentally flawed, because of a lack of identification, implying at the same time a general recognition by economists, based upon experience, of these models unsuitability: "It is still rare for empirical research in macroeconomics to be planned and executed within the framework of one of the large models" (p. 1). Sims argued then in favor of a second best type solution, advocating the consideration of "an alternative style of macroeconometrics," a "... dynamic system... estimated without using theoretical perspectives" (p. 2). As will be discussed, each of these is a questionable statement.

Three years further on, Edward Leamer (1983) advocated "Lets Take the Con out of econometrics," based upon earlier work (Leamer, 1978), suggesting Leontief-like a fundamental weakness in the economist's scientific method, using the characterization: "The applied econometrician is like a farmer who notices that the yield is somewhat higher under trees where birds roost, and he uses this as evidence that bird droppings increase yields. However, when he presents this finding at the annual meeting of the American Ecological Association, another farmer in the audience objects that he used the same data but came up with the conclusion that moderate amounts of shade increases yields. A bright chap in the back of the room then observes that these two hypotheses are indistinguishable [because of the identification problem], given the available data" (p. 31).

In contrast, 3 years after that, in an article in the *American Economic Review*, William Dewald, Jerry Thursby, and Richard Anderson (1986) reported on their study that investigated empirical studies published in the *Journal of Money, Credit and Banking*. They indicated that their research provided "new and important information about the extent and causes of failures to replicate published results in economics. Our findings suggest that inadvertent errors in published articles are a commonplace rather than a rare occurrence" (p. 587) and they also suggested that there might be questions about scientific method, although generally in the form of sloppiness. During just the past few years, at the beginning of this twenty first century, B. D. McCullough and H. D. Vinod (2003), Richard Anderson, William Greene, McCullough and Vinod (2007) and others (Breusch & Gray, 2006; Chapman & Gray, 2006; McCullough, McGeary, et al., 2006a, b; McCullough, 2006), in a series of reports in the *American Economic Review*, the *Journal of Money, Credit and Banking*, *the Journal of Economic and Social Measurement*, and other journals have reported on the situation today or have provided commentary (King, 1995; Mirowski & Sklivas, 1991), generally finding more confirmation of the 1986 Dewald et al. results.

In an associated development, during the past 10 years, both the accuracy of econometric software and the circumstances of its development and use have been the subject of an increasing number of studies (Brooks et al., 2001; Bruno & De

Bonis, 2004; Greene, 2004; McCullough, 1997, 1998, 1999a, b, 2004; McCullough & Renfro, 1998, 2000; McCullough & Vinod, 1999; Stokes, 2004a, 2005). Generally, the discovery has been – in the case of linear parameter estimators and related basic statistics – that econometric software compares quite well in relation to other software, but in the case of nonlinear estimators, in the few cases that studies have been made, potentially serious problems have been exposed (McCullough, 1998, 1999a, b; McCullough & Wilson, 1999, 2002, 2005). Against this should also be set the JSTOR findings on software use described in Chap. 2.

The way in which each econometrician will respond to these reported circumstances will of course depend upon background, beliefs, and preconceptions. However, considering them in the order they have been introduced here, it is evident that the common thread throughout is continuing concern on the part of the various investigators and commentators about either the circumstance or the formalized practice of econometrics during the past 100 years, with more emphasis on the past 50. Furthermore, it is important to consider this span of time and this variety of circumstances, not just particular expressions of points of view that date from the late 1970s and early 1980s. The choice of a full century as context makes it clear not only that quantitative economics and econometrics has had a combined adolescent and adult lifespan less than that of the airplane, automobile, radio, or television, which provides some perspective against which to view the comparative degree of operational development of econometrics. These circumstances may also suggest the degree of ambition of economists, in terms of their attempt to use as fast as possible, perhaps even too quickly, tools that have been evolving rather rapidly, so as to make seemingly verifiable, even verified, statements about the behavior of economic agents or the results of that behavior, including business cycles. However, the simple fact that these tools have been evolving creates in itself a serious question concerning the inferences that can be drawn from past research reports, irrespective of the quality of that research otherwise. Not only have the computational technologies been changing, but so also have the quality and characteristics of much of the data used. It is only very recently that anyone has begun to consider making available so-called "real time" data bases that in principle permit the reconstruction of datasets observation by observation as of some given reference time (Anderson, 2006; Anderson et al., 2007; Renfro, 1997a, b); relatively little, if any research has been done concerning the impact of alternative datasets, although at the edges preliminary estimates and data revisions have been considered (Howrey, 1978, 1996).

At the same time, it is quite clear that today differs technologically from 1907. Originally, in 1907, there was comparatively little data available, weak computational facilities, and no econometric theory. Wesley Mitchell, starting his measurement of business cycles about 1910, had no choice but to collect what data he could and use these in his inquiries, and this circumstance needs to be recognized whichever side of the Koopmans–Vinings debate one happens to vicariously choose (Foss, 1983; Kendrick, 1995; Kenessey, 1994; Vanoli, 2005). It is possible to criticize the quality of the data now available, as well as the quantities of particular types, and perhaps even the way that observations are constructed, particularly in the case of macroeconomic data, or the circumstances of their collection, especially in the case of microeconomic data. But it is also true that there has been a fundamental change from the time of Alfred Marshall, even if considerable improvement is desirable. However, what possibly should give the modern econometrician greatest pause is not in fact the quantity and quality of the data available today, but rather the continuing evidence of a widespread mishandling of these data, even separately from the particular computations made using them. There is very little evidence from the investigations that have been made recently that economists and econometricians have, in general, taken particular care to carefully document and organize the datasets used. Leamer's fable of the farmers may ring true, but is this necessarily because of the identification problem and faulty statistical inferences, any more than bad data handling practices?

Originally, in 1907, computational and data management facilities were severely limited. In 1972, computational facilities still remained comparatively poor for the typical applied economist, even if an econometrician who was a skilled programmer could produce qualitatively quite good results given sufficient time – although the time involved could, starting from scratch, actually be measured in months if not years. For example, for the typical applied economist in 1980, Sims' proposed six equation model was quite possibly far too demanding, in terms of the computer skills required, so that in fact the argument he made at that time could have been reduced to one that focused just on computational issues. For instance, if the focus is the typical applied economist, it is unquestionably true that in 1980 it was "rare for empirical research in macroeconomics to be planned and executed within the framework of one of the large models;" indeed, except in the case of those who were full time users and builders of such models, it was actually almost unheard of: practically no one else ever attempted to use these models. Before 1984, it was virtually impossible for any but a handful of economists to successfully solve the Chase Econometrics, Data Resources, MPS, or Wharton models. The reason is that the composite of the skills required was too great, not to mention the practical aspect of relatively restricted access to the mainframe computers on which these models were mounted. However, these particular models were nevertheless actively used: corporations, government agencies, and other organizations paid literally millions of dollars for studies to be made using them. But prior to the autumn of 1984, when for the first time a given model could be found on any of more than 100 widely dispersed Personal Computers, none of these models were ever successfully mounted and generally used on more than 3–5 computers each, if that, and certainly not on any regular basis (Dewald et al., 1986, pp. 596–597).

Even today, it remains a formidable project for an economist to construct and manage a large data base independently, containing a thousand or more time series, notwithstanding that each of these series can now be downloaded almost instantly from a government, trade organization, or other standard website (Renfro, 1980a, 1997a, b). But what may be most troubling is not the problems that the typical applied economist might have, or have had, working with a large macroeconometric model, but instead the trouble he or she might be having today working effectively with a two, three, four, five, or six equation model. Very little published information exists about the present day applied research practices of economists that would permit an objective, independent observer to validate that work. Characteristically, when an economist publishes today he or she does not provide, in particular, the computational details, including the specific version of the software used, and what if any precautions were taken to insure the computational accuracy of the reported results. Until reporting practices improve, it is not evident that extended debates about the fine points of methodology are worth pursuing, for often important details are left to the imagination of the reader. Consider in contrast the fact that it remains possible to reproduce and evaluate step by step the Klein models of 1950, the Klein–Goldberger model of 1955, and (with more work) probably even the Tinbergen model of 1939; the data and the computational details were then sufficiently well reported (Klein, 1950; Klein & Goldberger, 1955; McCullough, McGeary, et al., 2006a, b; McCullough, 2006; Renfro, 2003; Tinbergen, 1939).

Specific Survey Characteristics

At the cost of some repetition of earlier discussion, this section is intended to provide in one place a clear statement of the survey methodology of this study. As indicated in the Preface and at the end of Chap. 1, the survey that stands behind this volume was not conducted simply by sending a single questionnaire to each of the original or current developers of the included econometric software packages, which they filled out and returned once and for all. Instead, after the original responses were tabulated, both the tables and successive versions of the text were then sent to these same people several times for review, correction, and amplification, whenever necessary, in an effort to insure that the offerings of each of these software packages have been correctly represented. Furthermore, the numeric results that are shown in the figures, generally diagnostic statistics, have been independently calculated, in each case using two or more of the software packages in the survey, and of course the same underlying data sample. When the results have agreed, they have been included in the figures. When they did not, a further investigation was made in each case to determine why.

In most cases of original disagreement, it was discovered that the differences observed were the result of decisions intentionally made by individual developers and were often caused by the use of variant formulae, rather than because of manifest computational errors. However, although it is informative to mention that observed differences do not necessarily imply error, more stress should be placed upon the idea that much can be gained from an ongoing mutual consideration of the way in which econometric techniques, tests, and statistics should be implemented algorithmically. After initial independent attempts to implement, the progressive discovery of the reasons for observed numerical differences achieves both a better general appreciation for where the pitfalls lie and greater understanding of the specific effects of implementing a procedure one way rather than another. Of course, it does not follow that, for efficiency, this agreement instead should be sought in advance of an attempt to implement. To the contrary, such efficiency might ultimately result in predetermined stultification, rather than new knowledge.

The numerical outcome of the process just described is the set of figures that appear in Chap. 4–6, which identify all the discovered variants of the relevant statistics that are generated by the included econometric software packages. Although the presentational style of these figures is generally uniform and might be regarded as seemingly produced by a single package, each number reported, as indicated, has been independently produced by at least two packages. Many numbers are produced in common by all or most of the packages. Values uniquely produced by a single package have not been reported, not because they are necessarily incorrect or insufficiently precise, but simply because they could not be crosschecked and any differences fully evaluated. As indicated earlier, the initial, intermediate, and final versions of the displayed results were sent to the original respondents to confirm their accuracy and completeness.

As to the characteristics of the overall presentation, it offers among its findings a set of tables that are constructed from the information supplied by each of the developers of the existing econometric software packages – or in certain cases those currently responsible for maintaining them. These tables identify which statistics are produced by which packages. The packages surveyed are drawn from those described in the recently published compendium of (Renfro, 2004a, c), as discussed earlier. As indicated there, the primary qualification for inclusion is that the developer or proprietor of the program has (reasonably) presented it as being fundamentally econometric in orientation. Qualifying the programs was neither difficult nor contentious and, collectively, they can be regarded as defining the current state of the art of *operational* econometrics.

However, even at this relatively late stage in the analysis it still might be asked whether it is entirely reasonable to thus seemingly seal off the practice of econometrics from statistical influence? Econometricians still employ statistical concepts day-to-day and the boundary between econometrics and statistics is sufficiently permeable, and likely to remain so, that any attempt to mark a strict division is at some level impossible to justify. Furthermore, not only do economists continue to use SAS and other "statistical" software packages, they sometimes prefer them – in some cases possibly because of their greater tendency both to support more generally the use of cross-section data and to offer a broader range of potentially useful statistical facilities and techniques, some of these all but unmentioned in the econometrics literature. In turn, developers of statistical software packages still actively encourage use by economists; in particular, SAS explicitly includes the SAS/ETS (Economic Time Series) procedures and has for many years. Yet the manuals for SAS/ETS (Allen, 1982; Hill & Griffiths, 1993; Sall, 1979; SAS, 1993, 1999; SAS Institute, 2000) also emphasize that these "procedures are a part of the SAS system and behave like other SAS procedures." It is for this reason that it is useful to have drawn the line: there is an intersection between econometrics and statistics, but mutually recognized differences nevertheless exist. However, insofar as specific numerical calculations are concerned, certainly in the case of commonly defined operations, distinctions between econometric and statistical software are more tenuous. Therefore, as indicated once or twice in later chapters, some cross-checking of econometric results produced by SAS was done systematically. But it is nonetheless true that the motivation to perform a particular range of certain types of statistical tests is econometric in origin, so that the direction of influence considered is that of econometrics on statistics.

Together, the tables in Chap. 4–6 provide a reasonably detailed, if summary view of the results of the survey that stands behind this volume. However, as might be inferred from the description of the way it was carried out, the findings do not consist simply of these tables: more aptly, the word "provide" in the last sentence might be replaced by "ostensibly provide," for these tables should be regarded as being *only a starting point* for a consideration of what has been done. Considered more reflectively, they might be viewed essentially as pointers to the particular econometric pathologies that are individually and collectively seen by econometricians as being the most important to guard against. Many of the test statistics that are implemented are those that might be expected, but the selection made is nonetheless revealing as a statement of applied practice. And, of course, they represent simply a selection of those that have so far been proposed by econometricians, notwithstanding the interpretations made by individual software developers.

Furthermore, although these tables show that there is considerable commonality among the included tests, just as important a finding, albeit not immediately obvious, is that identifying names can hide as much as they reveal. As suggested by the earlier discussion of intentional differences, for many of the test statistics considerable variation was found among the programs. In certain cases, the differences consist of one package producing a statistic such that smaller values of it essentially imply "better," whereas for another program, bigger is better – but otherwise the test statistics are ostensibly operationally equivalent. In other cases, statistics have been constructed so that, in the case of one program, a particular example is constructed using a proper sub-sample of the observations that would be used by another program, but in each case the statistics produced are otherwise constructively the same; that is, the tests statistics are asymptotically equivalent, but differ for finite samples, all other things equal. In still other cases, there is a more fundamental constructive difference between the statistics that are offered by two or more programs, yet these are nominally presented as being the same test. These differences, which at times – but not always – appear to reflect an intentional and independent choice on the part of each developer, may in fact simply be the result of individual developers having originally acted independently and in the process differing somewhat in their interpretation of the essential constructive characteristics of these tests.

Whatever the individual reasons, because of such differences, it is necessary to be expositionally precise when characterizing the implementation characteristics of each of the test statistics. The terms "constructively the same" or "constructively equivalent" are intended to convey that the test statistic in question is the implementation of particular, arithmetically specific formulae found either in one or more textbooks or in an original paper (or papers, for there are certain instances of simultaneous independent formulation) that introduced the one or more statistics. Following from this qualification, a test of constructive sameness, given the same test dataset, would seem, at least at first sight, to involve an evaluation of the capability of a program to produce the same value of ostensibly the same test statistic as that produced by another (Bruno & De Bonis, 2004; Greene, 2004).

This concept of "constructive equivalence" is useful, and the just-described manner of evaluating the statistics produced by each of the programs has generally been adopted, for in the conduct of this survey there have been very few associated difficulties, particularly given the ability to repeatedly interview program developers in order to resolve apparent differences. However, the issue of how *best* to formalize a test of this property in more complex cases does lie reef-like just below the surface. In particular, constructive equivalence is not, except in special cases, simply a matter of inspecting the source code of each program in turn so as to determine algorithmic identicality, if for no other reason than that, algorithmically, there are often two or more equally good ways to code a particular procedure, even when using the same programming language. There is also the obvious issue that any such test will need to be one that takes into account that what is often involved is the evaluation of the secondary output of the program; for instance, if the computation of the statistic being considered involves first computing the residuals of the original regression, then a prerequisite clearly will need to be that each program tested must first possess the capability to produce the same, or effectively the same, sample of residuals.

The use of the word "effectively" is the result of the circumstance that another question to be considered is the degree of "sameness" required, which at least in part is a matter of representational precision. Recall from the discussion in Chap. 1 that, computationally, real numbers cannot be compared on the basis of strict equality, even if the rounded machine numbers that are displayed seem to imply that they can. In addition, there is also the usual matter of "under repeated sampling," which to some extent might be seen as a question of whether, for every possible sample dataset, "matching" values can be produced by two or more alternative programs. In particular, when nonlinear techniques are considered, matching across datasets is not assured (McCullough & Renfro, 2000). Simply in order to clarify the issues that are involved, this discussion has rather quickly begun to lead to a consideration of how to test software. In the process, it has led somewhat away from the most important initial concern, which is to discover the particular intended and after-the-fact characteristics of existing econometric software.

Focusing now on this goal, the most pertinent immediate question is: what have econometric software developers actually intended to do and how successful have they been in achieving their goals? Presumably the general intention has been to provide economists with a set of well understood, reliable, and familiar test statistics that can be used effectively to evaluate OLS equation specifications. Stated in this way, the question is rather open ended: of all possible tests that could be offered, are users of these programs actually being provided with those that are most appropriate in each situation that is likely to be or is in fact encountered? Of course the complete answer to this potentially dichotomous question can also quickly lead to a rather deeper background discussion, like that earlier in this chapter – well before beginning to consider explicitly the degree of developers' success. Such a discussion, on the one hand, can be seen as being necessarily econometric in focus (for instance, are the implemented tests properly conceived?) and, on the other, as leading back into the dense thicket of software testing (Altman, Gill, & McDonald, 2004).

Because of these evident complexities, for the moment it may be best simply to begin to consider what can be learned about what econometric software developers *think* they have accomplished. As indicated earlier, there is minimal past evidence in published form. At the outset of this survey, it was known from the grapevine that a variety of offerings has been available for some time and that not all developers implemented either the same tests or the same forms of a given test. For example, in the 1984 edition of his textbook, Johnston casually mentions in passing (p. 326) the use of two (today still existing) programs for various purposes, if mainly for parameter estimation. More recently, at various points in the 1997 successor edition (Johnston & DiNardo, 1997), Johnston and DiNardo suggest that particular tests and procedures are provided in one or other econometric software packages. These authors offer (p. 197), with reference to a particular package, the comment that "A comprehensive set of estimation and testing procedures is available...." However, they provide no additional specificity nor any evidence of what was then available. Greene, in his 2003 and later editions, somewhat more darkly notes a few specific differences between packages, providing at one point (p. 269) in his 2003 edition:

> A warning to practitioners: Current software varies on whether the lagged residuals are filled with zeros or the first P observations are simply dropped when computing [the Breusch–Godfrey] statistic. In the interest of replicability, users should determine which is the case before reporting results.

but otherwise there is little cumulated evidence readily available to guide either the applied economist, the econometric software developer, or the theorist who might be interested to know the operational state-of-the-art. As mentioned earlier, the most complete, existing statement of characteristics is the recently compiled compendium of econometric software packages (Renfro, 2004a, c), but even that lacks sufficient specificity.

Chapter 4
The Basic Statistics

The spirit of the present, somewhat econometrically jaded age is different from that in the immediate post-war years of the 1940s at both the Department of Applied Economics in Cambridge (Gilbert, 1988; Qin & Gilbert, 2001; Smith, 1998) and the Cowles Commission in Chicago (Christ, 1994; Morgan, 1990; Qin, 1993). These were then the two organized centers of applied and theoretical econometric research and the aspects of their published findings that deserve some modern attention are not only the ways in which the particular promulgated techniques shaped econometric practice subsequently but also the associated degree of intellectual excitement. Among the specific results emerging from the DAE in the late 1940s and early 1950s were the techniques and statistics now associated with the names Cochrane and Orcutt and Durbin and Watson (Gilbert, 1988; Qin & Gilbert, 2001). In addition to other research conducted by Stone and others, ignored here because of its present tangentiality, there was also the exploration by Geary, Stone, Tintner, and others of the connections between Instrumental Variables, Principal Components and Canonical Correlation (Begg & Henry, 1998; Gilbert, 1991; Qin, 1993; Smith, 1998). These investigations lead to later seminal work on Instrumental Variables by Denis Sargan, then a graduate student at Cambridge (Desai, Hendry, & Mizon, 1997; Hendry, 2003). As discussed to a degree in Chap. 3, the Cochrane-Orcutt technique and the Durbin-Watson statistic have of course been included in most econometric software programs of the past 40 years. Sargan's work, including further extensions, is embodied in the LSE method – but also in PcGive and other software packages (Gilbert, 1989).

The more broadly focused Cowles Commission ideas, documented in a still-imposing series of readings and monographs, are of course more widely recognized today as a distinct body of results. However, these too stem from a shared common heritage. Recall that, during the late 1930s and early 1940s, reflecting several circumstances, the joint dependency of economic variables was quite generally thought to be an important phenomenon, initially attracting the attention of Frisch, Haavelmo and Tinbergen, among others (Qin, 1993). The new Keynesian macroeconomics, with its stress on the joint dependency of Consumption and Income, then urgently implied the need to come to grips with it. But, at the microeconomic level, the independent requirement to discriminate between supply and demand curves implied the need to deal with sets of related equations as a general phenomenon in

C.G. Renfro, *The Practice of Econometric Theory*, Advanced Studies in Theoretical and Applied Econometrics 44, DOI 10.1007/978-3-540-75571-5_4,

empirical research. As Qin has pointed out [p. 68, 73], the immediate econometric consequence, as a stimulus to thought, was an initial rejection of the primacy of Ordinary Least Squares as an estimation method and, instead, a growing fascination with the principle of Maximum Likelihood. Specifically, by the end of the 1940s, the findings reported earlier by Haavelmo, (1943, 1944) and Mann and Wald (1943) had resulted in a more general statement of the estimation problem, which still admitted Ordinary Least Squares as a special case, yet allowed variable joint dependency to be formally included. This estimator emphasis-shift simultaneously focused the attention of the Cowles investigators on a series of issues: Qin [p. 68, 76] points out that these included such considerations as the population from which the observations were drawn, hence the attention paid to asymptotic properties; consistency and efficiency as focal issues, rather than bias; errors-in-equations, rather than errors-in-variables; and the joint distribution of both the random disturbances and the observable variables. These considerations underlie the so-called "Cowles Commission Method" as a response to the parameter estimation problem. Because of the lack of computing resources, this method was of course operationally rather restricted at that time, certainly much more so than today. However, the fortuitous discovery of Limited Information Maximum Likelihood (LIML) provided a computationally feasible, if partial solution, notwithstanding that it could still be somewhat arduous to implement using a Monroe or other desktop calculator (Goldberger, 2004).

As a direct consequence of the availability of LIML, in the description of their model Klein & Goldberger, (1955) stressed maximum likelihood, as did other Cowles Commission-linked presentations, as well as later performance studies of the KG model (Adelman & Adelman, 1959; Bowden, 1972; Howrey,1971). Earlier, in the description of his first three models, published in 1950, Klein similarly gave primacy of place to the LIML estimates (Klein, 1950, pp. 71, 80–122), although he also presented ordinary least squares estimates for Model I (p. 75), as well as for other particular equations. Inasmuch as the estimates reported were calculated using desktop calculators – albeit, in 1955, based to a degree on moment calculations using an early computer, as briefly described in Chap. 2 – this use of LIML obviously necessitated substantial extra work, compared to the use of OLS throughout.

In contrast, in his review of the Klein-Goldberger model in 1956, Karl Fox called attention both to the potentially severe computational demands of LIML in those years and to the difficulty the typical economist of that time might have had to understand its motivating precepts, as is perhaps also illustrated by Waugh (1961). In order to demonstrate an alternative, Fox then provided a complete set of OLS estimates for this model using the same data (Fox, 1956; McCullough, Renfro, & Stokes, 2006). He argued that inasmuch as "the general theory that underlies the limited-information method often leads one to single-equation, least-squares estimation as a special case" (p.130) there is not that strong of an argument to be made for it, for "…In addition, there are many practical situations in which it can be shown that the bias in least-squares estimation (relative to that in full-information maximum-likelihood estimation) will be small. While I do not believe that important issues of statistical methodology should be glossed over, it will be unfortunate if economists conclude that the study and use of economic models is closed to them unless they achieve an overpowering mastery of statistical technique" (p.130–131). However, as FIML was not actually implemented as an operationally feasible method until 1964, as described in Chap. 2, the question can be raised parenthetically, how could Fox be quite so sure when making his *quantitative* bias comparison?

Ease of computation and the dictates of pedagogy were shortly thereafter joined at the hip as econometrics began its widespread entry into the economics curriculum in 1963 on the coattails of the first edition of Johnston's textbook (Johnston, 1963). Possibly a more student-friendly, if meatier volume than earlier textbooks, Johnston's also had the good fortune to be published in time to catch the first wave of the "baby boom" generation. The exposition began with a methodological discussion of OLS. In contrast, the first reference to maximum likelihood, on page 20, consists of the bare statement that "If we postulate a normal distribution, then we can obtain maximum-likelihood estimators." Who knows how many students, albeit logically challenged, read this as a universal affirmative and simply converted it to: all maximum-likelihood estimators are OLS estimators when the errors are normally distributed? Thirty-four years later, the 1997 Johnston-DiNardo textbook (Johnston & DiNardo, 1997), continues to give maximum likelihood backseat treatment, pointing out the correspondence of Maximum Likelihood estimators to the since derived Generalized Method of Moments method, declaring that "MLE can be viewed as a GMM estimator" (p. 342). The authors then go on to ask rhetorically (p. 343), "if this is the case, then why do some researchers prefer GMM to MLE?" The answer they first give is "... tractability. Sometimes when the maximum likelihood estimator is difficult to compute, there is a GMM estimator that, although less asymptotically efficient than the ML estimator, is still consistent and easier to compute." The second is that "... sometimes, although not enough is known about the data generation process to specify the likelihood function completely, enough is known to specify moment conditions for a GMM estimator."

The inference that at this point should be drawn from these considerations is not to begin to suspect that what econometricians have done overall, or each step of the way, is necessarily wrong or misguided, but only that the historical computational situation has been an important conditioning factor, both in terms of what economists and econometricians have done and the manner in which econometrics has been taught. Indeed, it can be argued that the way that econometrics has been (and is) taught is critically important, including which results are presented, in which particular order, and with what emphasis. Furthermore, as a sidelight, it is interesting to consider the misgivings recently expressed by Clive Granger in his *ET interview* (Phillips, 2001) about the "whole evaluation process – both in econometrics and in economics generally." He goes on to say, "I want to know how people evaluate the theory and how people evaluate a technique and how to value the model." He then observes that "a lot of the literature is losing the viewpoint that we are here to learn about the actual economy," and furthermore he suggests that at least some economists are "... playing games when they write papers." Later (p. 62) he reflects that he "would like to see much more use of relevant economic theory in model building" and notes that [he is] "worried that the Econometrics Society includes both economic theorists and econometricians and that in their meetings the two sides never talk – or virtually never talk. There are few joint sessions involving both theorists and econometricians." In such an environment, it has been easy for the profession to ignore such seemingly mundane matters as the design of econometric software and its impact and to push all considerations of such issues into the deep background. Yet Granger's words imply also the degree of underlying importance of a better general understanding of the computational process.

It should be recognized specifically that, with a few notable exceptions, beginning in the 1960s and into the 1970s, there was a general predisposition, in sympathy with textbook presentations, for the early econometric software packages to offer predominantly OLS and those of its extensions most commonly found in those textbooks. Perhaps just as important in influence, the statistical software packages created in the 1960s and 1970s by non-economists, but which for many economists worldwide provided their first introduction to the use of the electronic computer, seldom – if ever – in those days offered LIML, but almost always OLS and even such problematic methods as stepwise least squares (Hocking, 1983). Otherwise, during this process, economists learned to keypunch data in quantity, rather than necessarily thoughtfully. The general effect in the end, amidst a declining immediate requirement to understand data construction methodologies, was possibly to create a research practice that was unthinkingly based upon the use of OLS and which may have consequently caused economists to justify this use reflexively. Of course, the tantamount consideration is not per se the favored choice of estimation method, but might be instead the implications of the placement of a license to compute into the hands of at least some who did not bring to the task solid research methodology training and a strong sense of research purpose; the most appropriate analogy might be a Jeep Cherokee as a teenage birthday present. As suggested earlier, such circumstances need to be considered when reading the words of Leamer (1978, 1983) and other commentators, written some 15–20 years later. The weak empirical research methodology in the 1970s and into the 1980s, decried by these econometricians, may be at least in part attributable to the way in which economists were introduced to computing and thus applied research from the early 1960s.

The Historical Display of OLS Regression Results

For whatever reasons and for better or worse, the workhorse parameter estimator for economists since 1956 has been the OLS estimator (Waugh, 1961). Certainly from 1963 to the present, it has been the first encountered parameter estimator for each generation of economists – a relatively intuitive and computationally tractable method now become a familiar old friend. However, the inferential problem this estimator poses, particularly in this modern age – now that the joint distribution of the observed variables is no longer the first terror of the econometric nursery it once was – has become that of marshalling the right set of associated statistical tests, including misspecification tests. Or so it is possible to argue. Of course, there are in addition several obvious questions that at some point also need to be raised, among these being which tests *should* be applied and in which order – and with which choice of a priori specification. An aspect of this evaluation is that of the "right" set, and this topic will be considered later in this chapter and in those that follow.

The fundamental question to be asked is, what set of statistics do the existing econometric software packages actually provide? To answer it might seem quite straightforwardly empirical, but there are several subsidiary normative questions that need to be considered on the way to this answer, for they bear on its aspects. The immediately most important of these has to do with the order in which each statistic is presented. Next is the question of the progressive elaboration of the information about the estimation problem at hand that the statistics potentially provide, including the choice of supplementary tests the user might be offered. But considering the statistics as a group and at first abstractly, various individual distinctions might be made. For instance, some statistics can be considered to stand equal in relation to each other and primary among all; these presumably should be provided by default. Others will be offered optionally, because their use is inferentially conditional, but this conditionality could also cause them to appear to the user to be subordinate in status. However, whatever the categorical groupings, the statistics and tests that are judged to be co-equal will still have to be presented in a particular order, which might affect how they are interpreted and possibly the esteem in which they are held, even if this ordering consists of nothing more than displaying a given statistical value on the first line of a screen or printout and another on some successive line. An obvious question is, what should be the hierarchy of the existing misspecification tests?

At this stage, to consider in detail the ideal placement of statistics line by line is too much of a good thing, but in a more general sense such matters are very much the subject of the present chapter and the two following. This ordering will be investigated initially in terms of what might be called the "basic" or "core" statistics, and then subsequently the other statistics. As just indicated, the sequence in which each group and its individual members are encountered is a potentially important topic, for the layout chosen is not simply a matter of esthetic design, in the trivial sense. Instead the order may be understood – by at least some users of a given econometric software package – to be indicative of the relative importance of each set of statistics, and perhaps that of the statistics individually. Casual groupings, even if arbitrary from the perspective of the software developer, can of course be interpreted by others as forming distinct, purposefully presented sets. And even if not always judged to be consequential, it is evident upon momentary reflection that, whenever something is displayed in a particular way by default – on the computer screen, on a piece of paper, or in the context of any other display medium – what is shown and how it is shown is always a matter of designer choice, either in the negative sense of no design or in the positive one of having been put there in a particular way for a particular reason. In this era of the iPod and the iPhone, this idea applied to econometric software might not now appear quite so strange as it might have only 5 years ago. The benefits of good design are becoming much more generally appreciated and the purpose of it more self-explanatory.

In the case of econometrics, an obvious fundamental circumstance is the knowledge and experience of the person practicing it. Someone who brings long experience and a strong sense of purpose might not be quite as affected by the ordering and placement of numbers, or which are shown by default and which are conjured optionally. This lessened sensitivity may occur in much the same way that quite generally a skilled software user faced with a new program, but knowing the use goal, immediately begins to explore for ways to attain it. Of course, in the end, if necessary facilities are lacking or the interface design is flawed or if the software is computationally unreliable, even the experienced user can be frustrated. However, it can still be argued that, for this user, in addition to inherent software reliability, it is desirable that the software be designed in a way that allows him or her to achieve each research objective as quickly and easily as possible. It may be difficult, even after some reflection, to state more specifically the actual design characteristics, inasmuch as these may be individual to particular users or else to users typed by research interests, but the need for at least some attention to design is evident in this context and the general benefits obvious. It is likely that, if questioned, many econometric software developers would claim their software to be designed for the knowledgeable user and with such an objective in mind, even if they also might admit that their existing packages still represent a work in progress.

In contrast, when faced with this type of package, the novice can find it somewhat challenging, given the need to understand implicitly both the econometric issues and how to make the software work. Software designed for the novice also needs to take into account the potential for it to be used pedagogically, which may imply the requirement for a variety of extra-econometric features (Cuff, 1980). For instance, to best support the software's use, such a person might need for it to provide not only some type of optional template, or a set of keyed templates, together even with examples and possibly extensive context help, but in addition a verbose mode to confirm at least a subset the various actions performed. The first time a software package is used, such confirmation is reassuring, even if it can become a nuisance to press the OK button the 10,000th time. Of course, although it might be possible to provide further examples of desirable pedagogic features, just as in the case of a package designed for expert use it is likely to be difficult, even on further reflection, to identify more specifically the ideal design characteristics, one of the reasons being that not much collective thought has yet been given to this matter. Few econometric software developers seem to have considered the practical implications, even if some will claim, as does Hendry, a pedagogic motivation, adding that "we see our 'PcGive' books as textbooks with a strong computational bent (and use them that way), rather than "manuals" (Hendry, 2003). Another of the potential design problems is the context of at least some of the student use of econometric software. It is not uncommon, even today, to hear a professor of economics or econometrics – particularly one approaching middle age – say blithely something to the effect, that inasmuch as he or she knows little about the computer, "I simply give the software to the students at the beginning of the course and let them sort it out among themselves." This sorting out by students certainly occurred in the 1960s and appears to have continued unabated since. Just how extensively, it is difficult to judge, but the conclusion is easily reached that it is an argument for the need for well-designed econometric software, if also a circumstance that makes that design quite difficult. It is the rare student who is a competent autodidact.

The operative distinction that has just been made is between econometric experience and inexperience. However, it is equally important in the final analysis to consider also the characteristics of the research process as a design criterion, which might be classified either by the research interests of the user or the particular research problem chosen. The research interests and preconceptions of economists are quite varied today and although it may be tempting as a sales management technique for econometric software developers to present individual packages as meeting all the needs of all comers, considered more objectively to treat each user at a given experience level as being interchangeable with any other is an idea that is difficult to sustain. There is an immediately obvious distinction between time series, cross-section and panel data. Within these categories further classifications are easily made, as was to a degree discussed in Chap. 2. For each individual category it might be possible to identify an ideal choice set of the statistics and tests made available to the user.

However, in the presentation of the survey results here, these use differences have generally been ignored, both because it is interesting at first simply to discover what econometric software developers individually and collectively make available and because of the difficulties associated with attempting to classify each of the packages at this initial evaluative stage. There is also the consideration that one of the purposes of this discussion is to be consciousness-raising and, when so engaged, there may be a virtue in not trying too hard to seem to have a ready answer. But, at some time, a proper classification of econometric software packages by user category ideally will need to be made and the design implications examined. Going forward, there is clearly a need to consider all these matters. There are a variety of general questions that might be asked. For instance, should the design of software incorporate the goal of raising the computer literacy of the profession? Or will it be necessary to wait a further 20 years for those who gain their general computer literacy in childhood to begin to become seasoned economists and econometricians? It is possible, of course, that this characterization of the situation is too pessimistic. But it is not the most pessimistic possible. More stress could be placed upon the possibility that when econometric software is used, the numbers displayed may lie, if the computations are badly made or if the input observations involve serious errors of measurement or do not reliably correspond to economic concepts or if users do not quite understand what they are looking at.

Of course, in the beginning, the econometricians who first began to program the computer seldom gave any thought to the matter of "design," to how, what, and the way something is displayed might affect the interpretation of results. At first, the focus necessarily was simply upon generating any result at all amidst an inevitable series of computer crashes – such is the early nature of computer programming. Even today, the seeming lack of a sense of design may be a badge of honor among the econometric cognoscenti. If put to a vote, the idea that design plays any part whatsoever in specification evaluation might still be greeted with at least a modicum of disbelief and bemusement. Nevertheless, whenever the layout of a regression display is prospectively considered, the argument just made suggests that there must in principle be some bare minimum amount of information that can be defined as an ideal "core" set, presented by default. Of course, it may be difficult to determine what properly constitutes this ideal set, so much so that, in order to approach this topic in a way that provides an effective means to proceed, it might be best to start historically. The very unfamiliarity of the distant past allows it on contact to have a potentially refreshing impact.

The obvious first design question is, what should be displayed? To those economists who first used the computer to perform a regression, possibly involving the prior transformation of the time series used, it was patently obvious that the parameter estimates themselves must be displayed. Certain results were familiar already from the use of desk calculators: during this pre-computer age, the elements of the principal diagonal of the inverse of the cross products regressor variable matrix were generated, essentially as a by-product. Other values could be generated, as Arthur Goldberger has recently described (Goldberger, 2004), including the innovation provided by a mathematician that more or less automatically gave the observation count as a memory aid. It is not known if the first economists to write a regression program necessarily each began their careers using a desk calculator and only then wrote computer programs (Desai, 2007; Preston, 2006; Renfro, 2004). But in whatever particular way all this came about – if anyone today can remember precisely – unquestionably the earliest regression computer "printouts" ordinarily contained a vector of parameter estimates, a matrix of cross products, and the inverse of the regressor cross products matrix. Most also included a number representing the number of observations and, as relevant, some might show the date range. Recalling this information set requires no feats of memory, inasmuch as almost all these numbers were needed as a debugging aid during a program's original creation.

The most notable subsequent innovation is likely to have been the first inclusion of a few misspecification tests. Here the textbooks of Goldberger, Johnston, Klein, and others (Desai, 2007) came into play to provide the ideas, but not in so many words. "Misspecification test" is a modern term for a modern concept and exactly what constitutes such a test in the original context is a little difficult to determine from a standing start. Recall the discussion in the last chapter about the contents of the early textbooks. It is best not to try to apply a modern day concept at this point. Instead, it is instructive to start by considering actual historical examples of the primitive econometric art form known as a regression display. These displays did not in fact change a great deal from 1968 to 1990, and this circumstance is of greater importance than might be commonly thought.

As indicated earlier, the earliest regression displays were sometimes, indeed often, nearly indecipherable except by the program creator. It was only once programs began to be created for other people to use that easy reading of results necessarily became a concern. However, this first creation for use by other people actually occurred rather soon, albeit in isolated cases: one of the earliest econometric software packages of this type is RGR, written by Arthur Stroud. It became available to economists at the University of Wisconsin in 1962 and its user guide was written by Arthur Goldberger & Verona Hofer (1962). Reflecting the fact that in 1962, in the absence of such software, the user would have been responsible for every aspect of the computations made, it was natural then to display both the final and intermediate results. In particular, RGR presented its results by first displaying the "raw moments matrix," followed by the correlation coefficients of the variables. Then in order to present the information that might finally appear in a journal article or a book in the form:

$$\begin{array}{l} \mathrm{H_g} = 363.9 + 1.070\ \mathrm{Y} - 6.745\,\mathrm{P} - .5825\,\mathrm{L/V} \quad \mathrm{R}^2 = .4666 \\ \qquad\quad (415.4)\ \ (.3517) \qquad (5.172) \qquad (.2367) \end{array}$$

the program user was instead given, as a "printout," the information replicated below as Fig. 4.1. The equation just above displays the same regression results for the time period 1920–1929, where the values shown in parentheses are standard errors and R^2 is adjusted for degrees of freedom (so as to yield the modern R-Bar Squared).

```
REGRESSION EQUATION  1

   DEPENDENT VARIABLE 1
   4 INDEPENDENT VARIABLES
   6 DEG FREEDOM

   MULTIPLE   CORR   COEFF
    1-R2       R2        R
   .3556    .6444    .8028
   .5334    .4666    .6831    DF CORRECTED


   ORDINARY EQUATION

            SUM OF SQUARES

          TOTAL        REGR         ERROR
         3.331E+04   2.147E+04   1.185E+04

            STANDARD ERROR OF ESTIMATE

            SE2            SE
        1.974E+03      4.443E+01


              0           2            3           5
     B    3.639E+02 1.070E+00 -6.745E+00  -5.825E-01
    SD    4.154E+02 3.517E-01  5.172E+00   2.367E-01
  B/SD    8.761E-01 3.042E+00 -1.304E+00  -2.461E+00
  PCYI    3.368E-01 7.788E-01 -4.699E-01  -7.088E-01

   NORM OF D IS 13869.2   NORM OF I = AD IS .259E-05   1  ITER


            F            VARIABLES
          4.9746            2  5
```

Fig. 4.1 RGR regression display

The printout in Fig. 4.1 is not a complete replication of what a user would have seen; not shown is the so-called "INVERSE OF RAW MOMENT MATRIX WITHOUT DEPENDENT VARIABLE," which then followed these results. This matrix has been omitted simply because it is not necessary for present purposes. The results shown were generated using a CDC 1604 computer and from a notation at the bottom left corner of the original printout (not shown here) the calculations appear to have taken 11 sec. From a modern perspective 11 sec is a long time, although it is not clear precisely what this time count might have meant; that is, whether it referred to "real" or execution time? Remember that in the days of shared computers, "jobs" could be prioritized by their computer resource requirements, including the amount of memory ("core storage") used. Furthermore, during execution, jobs might be swapped in and out of memory multiple times, as is actually still true in a Windows or other interrupt-driven operating system environment, the difference today being the possible use of the term "active memory," to imply virtual swapping.

Notice that the "printout" displayed in Fig. 4.1 presents a strong contrast to the "printed book" equation displayed above in more or less conventional standard form with the R^2 to the right and the standard errors below. To decipher the printout requires additional effort. In the figure, the parameter estimates are given on the line labeled "B." The numbers 0, 2, 3, 5 on the line above these estimates refer respectively to the constant term and the relative location of each of the other variables in the raw moments matrix of the original data set, as read in. Apparently, this matrix was computed first and then the variable selection was made. With reference to the estimated equation above, variable 1 is H_g, variable 2 is Y, variable 3 is P and variable 5 L/V; 0 refers to the constant term. The SD row consists of the standard errors, B/SD the t-statistic values, and PCYI the partial correlation coefficients.

The F statistic shown is defined as:

$$\mathrm{F} = \frac{\Delta SSR/H}{SE2}$$

where ΔSSR is the decrease in the regression sum of squares that would occur if the "H independent variables, whose [identifying] numbers are given, were dropped from the list of independent variables."

With the guidance just given, it is not difficult to interpret this "printout," and of course to anyone who used RGR daily its conventions would soon become second nature. However, the example does convey some of the flavor of that time. In particular, it was evidently not thought necessary then to "design" a regression display and it is obvious that the information provided by RGR falls short of what would today be considered to be sufficient. Generally speaking the program user was offered very little display flexibility, although there was some ability to include optional title lines, which could be used to give also the sample period date range and other pertinent information. In fact, as the user guide to RGR makes clear, the process of performing a regression was sufficiently daunting in those days that the reader was advised that a "programmer" might be consulted during the process of setting up a "run," a process that involved using paper coding sheets – in a manner that is in some respects familiar even today, on those occasions that forms are filled with information that will be keypunched or optically read.

```
1:   GC = C1*GC(-1)+C2*GYD+C3*DEL(1 : GDC)+C4

NOB = 66    NOVAR = 4
RANGE = 1955 1 TO 1971 2
RSQ   =   0.99958     CRSQ = 0.99956     F(3/62)  =    4.91E+04
SER   =   2.5084      SSR = 390.109      DW(0)  =    1.79

COEF        VALUE       ST ER        T-STAT

C1         0.71129    0.06411     11.09540
C2         0.27188    0.05602      4.85281
C3         0.46347    1.50555      0.30784
C4         0.87164    1.43087      0.60917
```

Fig. 4.2 1979 TROLL regression display

Of course, easy reading is a relative description. To choose an alternative example, a user of TROLL in 1979, or even earlier, would have seen a display essentially like that shown in Fig. 4.2.

This particular example is taken directly from the February 1979 *Troll Primer, Third Edition* (Center for Computational Research in Economics & Management Science, 1979). It is shown here because of its particular features and the fact that this display (or something much like it) could have been encountered in the wild at almost anytime from the early 1970s. The layout has a number of characteristics standard for that time, although actually, even for its time, this display exhibits a substantial degree of sophistication in certain of its aspects. An indication of its refinement – and which illustrates that it is representative of developments during the 1970s, even if traceable to work that began in the 1960s – is the first line, the equation display itself. The equation shown very nearly replicates the directive command a user would employ to specify the regression to be performed, including the embedded transformations.

It is useful to consider, in contrast, the prevailing style in the late 1960s. A printout display of essentially the same information, not generated using TROLL, might instead have looked more like the regression display shown in Fig. 4.3. Notice, in particular, that in this example, the labels associated with the regressor variables (GC,GCLAG,GYD,DF1GDC) refer, in some cases, to transformed variables, rather than being original variables embedded in implicit functions. TROLL in 1979, and earlier, was able to perform a series of interpretative operations in response to a user's command. That is, it could parse a command that included embedded arithmetic operators and implicit functions, then extract time series observations on the variables from a data bank, make the specified transformations, perform the regression, and display the results. In the absence of a program's ability to operate in this fashion, it was ordinarily necessary for the user to make variable transformations in advance and then to issue a regression command that identified the dependent and regressor variables after transformation, usually employing user-supplied name labels. Because of this use of labels, which might easily be confused with each other, such renaming can also lead to inadvertent mistakes being made, mistakes that are then no longer obvious when viewing a regression display, since

```
DEPENDENT VARIABLE: GC

NOB = 66     NOVAR = 4
RANGE = 1955 1 TO 1971 2
RSQ   =   0.99958     CRSQ = 0.99956     F(3/62)  =    4.91E+04
SER   =   2.5084      SSR = 390.109      DW(0)  =    1.79

LABEL        VALUE       ST ER       T-STAT
GCLAG       0.71129    0.06411     11.09540
GYD         0.27188    0.05602      4.85281
DF1GDC      0.46347    1.50555      0.30784
CON         0.87164    1.43087      0.60917
```

Fig. 4.3 Mock up of a late 1960s regression display

the transformations and original variables are themselves no longer shown. Such differences can of course be operationally significant.

However, considering all the above displays together, what also should be noticed is the relative sparsity of evaluative information about the regression performed. The parameter values are displayed, together with their standard errors and t-statistic values. The date range is given, as well as the number of observations. Otherwise, even in the case of Figs. 4.2 and 4.3, the only evaluative information is that contained in the adjusted and unadjusted R^2 values, the F-test values, the Durbin-Watson statistic, the Standard Error of the Equation, and the sum of the squared residuals. Arguably, the most informative numbers, of those shown, are the standard errors and t-statistics on the parameter estimates. The implied bias of the Durbin-Watson towards 2.0, because of the lagged dependent variable, limits its information content. Even the given original names of the variables provide relatively little information, although the names are sufficient, after just a little thought, to permit this regression to be interpreted as consumer expenditures on disposable income, itself lagged one period, and the first difference of durable consumer expenditures. Given this macroeconomic context, the F-test is therefore not particularly informative; a significant F-test is of course to be expected for such a regression. Is it any wonder if the typical applied economist might have been tempted to maximize the (adjusted) R^2?

The examples just shown have been presented in order to call attention to some of the characteristics of regression displays over the years, essentially as a consciousness-raising exercise. It would be potentially interesting, of course, to be able to go back in time and recreate a variety of regression displays, simply to get a feel more generally for what used to exist. This will be done in a later chapter, using actual historical examples. However, at the moment, that set of displays represents a tangential consideration. From this point on, except for the results shown in Chap. 7, the numeric results displayed in this volume will be modern examples, intentionally generated using easily obtained, previously published data. Furthermore, as indicated earlier, the observations used to generate these results are included in this volume's appendix. Therefore, anyone who is interested in replicating them should be able to without difficulty, using any econometric software package, to the extent of course that it provides the range of statistics shown in this volume.

Survey Results: The Core Statistics

In his 1971 textbook, Henri Theil considered an application that implements an investment theory originally proposed by Grunfeld (Grunfeld, 1958; Theil, 1971). Theil there briefly describes and applies Grunfeld's theory using annual data on two corporations, General Electric and Westinghouse, during the period 1935–1954. He first displays the Ordinary Least Squares estimates he obtained. Figure 4.4, below, shows those for General Electric. As stated earlier, in order to insure the possibility of replication, all the values exhibited have been independently re-estimated using several modern packages, all of which agree numerically. In his treatment, Theil then went on to consider Zellner's Seemingly Unrelated Regression Equations technique (Zellner, 1962; Zellner & Theil, 1962; Zellner, Stroud, & Chau, 1963), but this aspect of the Theil example is ignored here, although clearly this un-followed next step potentially carries with it an implication for the OLS estimates to be shown.

As indicated, one of the primary considerations behind the choice of the Theil example has been the easy and general availability of the underlying data. Other published OLS values for this data set can also be found in Maddala (1977) and Kmenta (1986). But if this data set provides a means of generating parameter estimates and supporting statistics, there is not actually a consensus among econometricians what the bare minimum set should be, as the discussion earlier in this chapter began to address. However, for the moment, it may be possible to continue to finesse this circumstance to at least some degree, for the values shown below in Fig. 4.4 are among those most commonly automatically displayed by the existing econometric software packages. The comparative table, Table 4.1, which follows on the next page, explicitly confirms the degree of commonality.

```
Equation Number:    1
  Sample Period:    1935-  1954
  Number of Observations:   20
  Ordinary Least Squares

  IGE=C1*FGE(-1)+C2*CGE(-1)+C3

           Parameter   T-Statistic     Std Error      Mean
  C1)       0.02655      1.70571        0.01557     1941.325
  C2)       0.15169      5.90155        0.02570      400.160
  C3)      -9.95631     -0.31734       31.37425        1.000

  Variance=          777.446299     Dependent Mean=          102.290
  Standard Error=    27.882724      Percent Error=            27.26%
  R-Square=             0.7053      R-Bar-Square=             0.6706
  Schwarz Criterion=   6.9429       Akaike Criterion=         6.7935
  Max LLF=            -93.31373     F-Statistic: F( 2, 17)= 20.3435
  DW Statistic=         1.07210


Glossary of Variables
    CGE     General Electric capital stock
    FGE     General Electric Market Value at end-of-year
    IGE     General Electric, Annual Investment Expenditures
```

Fig. 4.4 The basic statistics

Table 4.1 Core regression summary statistics for OLS time series regression

	Standard error of estimate	Variance	F-test	R^2	Adj R^2	Durbin-Watson	Durbin h	Max LLF	Von Neumann ratio	Akaike criterion	Schwarz criterion
Independent software packages											
AREMOS	X	X	X	X	X	X	(X)				
AutoBox	X	X	X	X	X	X				Log form	Log form
B34S	X	X	X	X	X	X		X	X	Both forms	Both forms
Betahat	X	X	X	X	X	X	(X)	X		Log form	Log form
EasyReg	X	X	X	X	X	X				Log form	Log form
EViews	X	(X)	X	X	X	X		X		Log form	Log form
FP	X	X		X		X					
gretl	X	X	X	X	X	X		ML		Log form	Log form
LIMDEP	X	X	X	X	X	X		X		Log form	Log form
MicroFit	X	X	X	X	X	X	(X)			(Log form*)	(Log form*)
Modeleasy+	X	X	X	X	X	X		X		(Log form)	(Log form)
MODLER	X	X	X	X†	X	X*†	(X)	X		Log form	Log form
NLOGIT	X	X	X	X	X	X		X		Log form	Log form
PcGive	X	X	X	X*	X	X				(Log form)	(Log form)
RATS	X	X	X	X	X	X		X		(Log form)	(Log form)
REG-X	X	X	X	X	X	X				(Log form)	(Log form)
SHAZAM	X	X	X	X†	X	X	(X)	X	(X)	(Both forms)	(Both forms)

SORITEC	X	X	X	X	X	X				
Stata	X	X	X	X	X	(X)	(X)	(X)	(Log form)	(Log form)
TROLL	X	X	X	X	X	X		X		
TSP	X	X	X	X	X	X	X*	X	(Log form*)	Log form*
Wysea	X	X	X	X	X	X				
Econometric programming language applications										
GaussX	X	X	X	X	X	X	(X)	X	Log form	Log form
TSM 4	X	X	(X*)	X	X	(X)	(X)		(Log form*)	(Log form*)

() – indicates that the statistic can be produced at the option of the user
ML – implies that the statistic is produced automatically, but only when technique performed is Maximum Likelihood
R^2 – The * indicates that the R^2 is not reported if the regression constant term is suppressed. The † indicates that instead the uncentered R^2 is reported if the regression constant term is suppressed.
F-test – The * indicates that the test performed automatically retains any included lagged dependent variables and dummies under H0, and so is a test for joint significance of the exogenous regressors only.
Durbin-Watson – The * indicates that when quarterly, monthly or other non-annual frequency data are used, a second, supplementary statistic is shown (the so-called Wallis-DW) constructed using a lag set by the maintained frequency. The † indicates that when the regression constant term is suppressed so is this statistic.
Durbin h – The * indicates that the program also produces the so-called alternative Durbin statistic, sometimes called the Durbin h alt. Note also that TSP, in particular, will produce the Durbin h by default when the equation specification explicitly includes a lagged dependent variable.
Schwarz and Akaike – The * indicates that the criteria are constructed by simply subtracting the degrees-of-freedom correction from the LLF, such that "bigger = better" as described in the text.

Considering the values shown in Fig. 4.4, with the exception of the Akaike Information and Schwarz criteria, and perhaps the maximum Log Likelihood, many of the values in this group would be familiar as a display to users of a typical econometric software package even as early as 1971 or perhaps just before, as is evident from the examples displayed in the previous chapter section. However, when what is displayed in this figure is compared with the 1979 TROLL regression display shown in Fig. 4.2, or the mock up shown in Fig. 4.3, one obvious difference is the degree to which the numbers are more clearly identified in Fig. 4.4. For instance, it is no longer necessary to decipher SSR or SER or CRSQ. As long as the viewer immediately understands the econometric characteristics of the numbers, Fig. 4.4 is relatively revealing.

Considering this regression display, the constructive characteristics of many of the statistics shown vary relatively little among the modern day packages, virtually none for the most basic of them. In particular, in the case of the Theil-Grunfeld data, from package to package the variance is constructed as the sum of the squared residuals over the period from 1935 to 1954, divided by the number of observations (20) less the number of regressor terms (3). The Standard Error of Estimate is computed as the square root of the variance. The R^2 statistic consists of the Explained Sum of Squares, divided by the Total Sum of Squares, in each case measured as mean deviations. Inasmuch as it is a constructive property of any OLS regression that includes a constant term that the residual mean is zero, the residual values are automatically measured as deviations from zero in this case. Alternatively, and exactly equivalently, the R^2 can of course be computed as 1 minus the Residual Sum of Squares divided by the Total Sum of Squares. In addition, as will be shown, the adjusted R^2 is obtained by making an explicit degrees of freedom adjustment. For the moment, ignore the remaining supplementary statistics.

The Core Statistics: The First Group

The results just considered can of course be presented more formally: As is familiar from almost any textbook (Davidson, 2000; Greene, 2000, 2003; Johnston & DiNardo, 1997; Theil, 1971), given the linear specification:

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{u}$$

where:

$\mathbf{y}$ – a vector of T observations on a variable y
$\mathbf{X}$ – a matrix of T observations on k regressor variables
$\boldsymbol{\beta}$– a vector of k unobserved constant parameters
$\mathbf{u}$ – a vector of T unobserved disturbances

and $\hat{\mathbf{u}}$ the corresponding calculated sample values vector, the historically most time-tested of the statistics exhibited in Fig. 4.4 are those defined by the particular formulae:

$$\text{Variance} = \hat{\mathbf{u}}'\,\hat{\mathbf{u}}/(\mathrm{T}-\mathrm{K})$$

$$\text{SEE} = \text{SQRT}[\hat{\mathbf{u}}'\,\hat{\mathbf{u}}/(\mathrm{T}-\mathrm{K})]$$

$$\text{TSS} = \text{ESS} + \text{RSS} = \hat{\beta}'\mathbf{X}'\,\mathbf{X}\hat{\beta} - \mathrm{T}\,\bar{y}^2 + \hat{\mathbf{u}}'\,\hat{\mathbf{u}}$$

$$\mathrm{R}^2 = \text{ESS}/\text{TSS}$$

$$\bar{R}^2 = 1 - \frac{T-1}{T-K}(1-R^2)$$

$$\mathrm{F} - \text{Test} = \frac{ESS/(K-1)}{RSS/(T-K)}$$

$$\text{Percent Error} = (\text{SEE}/\bar{y}) \cdot 100$$

$$\text{Max LLF} = -\frac{TK}{2}\ln 2\pi - \frac{T}{2}\ln\left(\frac{\hat{u}'\,\hat{u}}{T}\right)$$

such that

T = the number of observations
K = the number of regressor variable coefficients and **X** matrix columns
TSS = total sum of squared deviations from the dependent variable mean
ESS = explained sum of squared deviations from the dependent variable mean
RSS = residual sum of squared mean deviations
$\bar{y}$ = dependent variable sample mean

An inference that can be drawn from this statement of the constructive characteristics of these results might be that the method of calculation is only too completely obvious: how else could these statistics be computed? However, do not forget that a distinctive aspect of the present description is that it is focused on the way that econometric software packages actually calculate relevant statistics. When econometric results are considered from this perspective it is important that nothing should be taken for granted. Anyone who has even a passing acquaintance with computer programming cannot help but be aware that it is often only too easy to think that a particular piece of untested code implies a specific result but to discover later that this is not necessarily true. In this context, it is always mandatory to test rather than to assume. Nevertheless, these results are standard and they are also quite easily programmed, by applying the above formulae in a straightforward way, provided that the parameter estimates have been computed in a numerically accurate manner in the first place. In general, these results numerically depend on the numeric properties of the parameter estimates. However, as will be discussed shortly, there are also certain presumptions incorporated into these particular formulae.

Because all the packages considered here have been successful in these computations, the various associated numeric analytic computational issues will not be discussed in any detail. However, when self-programming is done, it is generally necessary, even in the case of the above statistics, to make the computations using reasonably high precision number representations (for example, using so-called double precision), as well as to make use, to the extent possible, of the implicit hardware representations of numbers such as π. More generally, for other data sets, numerical accuracy should not be assumed simply on the basis of the ability to replicate the results shown in Fig. 4.4; the Grunfeld data are *not* computationally challenging in the way that other data sets can be (Longley, 1967; McCullough, 2004; Stokes, 2005). Similarly, a program's ability to generate these results implies nothing about that program's numerical accuracy when performing other, particularly non-linear, parameter estimation (McCullough & Renfro, 1998, 2000). Furthermore, widely available, non-econometric programs that can be used for parameter estimation will not necessarily be numerically accurate (McCullough 2000; McCullough & Vinod, 1999, 2002, 2005). A final caution is that the above formulae pertain to a subset of the values shown in Fig. 4.4; there are certain special cases that now need to be considered separately.

Known Special Cases

The form of the above-stated statistics can of course vary somewhat in special cases. For instance, if and when autoregressive corrections are made, the autoregressive constants that are estimated during this process will increase the number of coefficients estimated. The making of such corrections has become less common in the past 25 years, inasmuch as the prevailing general sentiment among econometricians today is that disturbance serial correlation, when discovered, generally implies misspecification – hence the need to rethink the original specification's characteristics and form, rather than to apply an essentially mechanical correction; Qin and Gilbert have recently discussed certain aspects of this point of view as it applies to both the VAR and LSE methodologies [69, p. 425]. However, from the perspective of the software developer, politically correctness is not necessarily the deciding factor: simply as a textbook-described technique, autoregressive corrections still remain a part of the economist's kit bag; therefore in practice most packages continue to permit these to be made. Consequently, the question remains: how should each of the diagnostic statistics be computed when such corrections are made?

Interestingly, this is one of a number of similar topics that have not attracted much attention in the literature. It is one of those small issues, so small that it is unlikely to be considered even as a note in an econometric journal. Almost any econometrician, if asked, will be able to provide an opinion, as will almost any econometric software developer, but variations in treatment between packages do occur. One of the very few reviews of econometric software to actually make a numeric test is that by Lovell and Selover in 1994 (1994). This review was only *one of three*, out of the more than 120 software reviews published during the years 1990–1997, as counted by McCullough & Vinod (1999), to report on the numeric accuracy of the software considered; a fourth was published in 1998. An element of the Lovell-Selover review, entitled *Econometric Software Accidents*, was its consideration of autoregressive corrections. One of the findings, using a Cochrane-Orcutt AR(1) correction, was that each of the four packages they tested differed in their ρ estimates for each of three data sets; only in certain cases could these differences be satisfactorily explained, as reflecting known differences in specific AR(1) techniques.

Arguably, the appropriate design policy in such cases should be that, *if a procedure is permitted by the software*, whenever it is performed a specific result should be obtained. Such a dictate makes sense, but actually it is not known to what degree, when performing autoregressive corrections, each econometric program that offers a particular option will generate essentially the same result. What *is* known from the Lovell-Selover review is that the parameter estimates themselves can differ between programs. Furthermore, as yet, there is even some (mild) controversy about the construction of the R^2 statistic in the presence of such corrections, Theil, for example, argued in his 1971 textbook that inasmuch as it is possible to consider the residuals in either of two forms, with or without the autoregressive parameter(s) a penalty should perhaps be constructively assessed. In the general case that the constant term is omitted, Theil also points out that there is an open question concerning the construction of this statistic (Footnote 4, p. 178). Because of prevailing econometric sentiment, to deal with autoregressive corrections is beyond the scope of the present monograph, but the matter has been raised in order to make it clear not only that the present investigation is limited but also that somewhat further out, where the waves grow larger, there be dragons lying in wait.

Another OLS related technique that can be encountered, and which was referred to earlier, is stepwise regression; this too is out of favor, but not necessarily in the case of *statistical* software packages (Foster & Stine, 2006). Of course, one of the issues that the provision of such options raise is whether they, by their presence, encourage what might be termed "bad research practices?" On the other hand, who is to say what economists and econometricians should be allowed to do? As discussed earlier, econometric software developers inevitably play a gatekeeper role, but to what extent should this be seen to be intentionally censorious? This is a question that obviously needs to be considered carefully, from a variety of perspectives.

Variant Special Cases

Two of the statistics shown in Fig. 4.4 have not yet been defined. The Akaike Information (Akaike, 1973) and Schwarz (1978) criteria can respectively be stated in log form as:

$$\mathrm{AIC(K)} = \ln\left(\frac{\hat{\mathrm{u}}'\,\hat{\mathrm{u}}}{T}\right) + \frac{2K}{T}$$

$$\mathrm{SIC(K)} = \ln\left(\frac{\hat{\mathrm{u}}'\,\hat{\mathrm{u}}}{T}\right) + \frac{K}{T}\ln \mathrm{T}$$

or in original form as:

$$\mathrm{AIC(K)} = \mathrm{s_y^2(1 - R^2)e^{2K/T}}$$

$$\mathrm{SIC(K)} = \mathrm{s_y^2(1 - R^2)N^{K/T}}$$

where

$$\mathrm{s_y^2} = \Sigma(\mathrm{Y_t} - \bar{y})^2/\mathrm{T}$$

for t = 1,2,... T. These statistics play a role similar to the adjusted R^2 inasmuch as in each case, the appearance of the T and K values, obviously mainly the K values, causes a penalty to be assessed as the number of regressor coefficients increase. The trade off is between the lowered residual sum of squares, due to any increase in the number of regressors, versus the K value penalty that each of the statistics simultaneously assesses. The Akaike Information and Schwarz criteria obviously can each be stated in either form, although today (as of 2008) econometric packages seem to display the log form uniformly, even if this is not instantly obvious to the casual user – and certain packages permit either or both forms to be displayed. Concerning the relative behavior of the statistics, as Greene (2000) points out, the Schwarz criterion imposes a heavier penalty for degrees of freedom lost, compared to Akaike, and will therefore argue relatively in favor of specifications containing fewer parameters.

The observed numeric value of the Akaike Information and Schwarz criteria can also vary from package to package depending upon whether or not the "inessential constants" are excluded, as well as other somewhat similar formulaic differences. Akaike originally used the full log-likelihood, and that specification compared to the log form above involves the addition of:

$$\ln 2\pi + 1$$

EViews, for example, displays the two log form statistics as:

$$\mathrm{AIC} = 9.631373 \tag{4.1}$$

$$\mathrm{SIC} = 9.780733 \tag{4.2}$$

instead of the values shown in Fig. 4.4, which were generated by MODLER. Inasmuch as:

$$\ln 2\pi + 1 = 2.837877$$

the values for each program are exactly comparable, once this subtraction is made. Another example is provided by Stata, which instead displays the two statistics as:

$$\mathrm{AIC} = 192.6275$$

$$\mathrm{SIC} = 195.6147$$

Initially, at a very quick glance, these values might be interpreted as being in antilog form, except if one happened to divide them by 20 (the number of sample period observations), in which case it immediately becomes clear that they are by this transformation numerically exactly equivalent to the EViews results: upon division, the values obtained are 9.631175 and 9.780735 respectively, the precision differences reflecting the rounding error involved in this side calculation. Furthermore, as shown in Fig. 1.1 (displayed in Chap. 1), B34S employs a slightly different degrees of freedom correction compared to Stata. Which all goes to show that it is not only with legal contracts that it is necessary to read the fine print.

However, there are yet more variations to come. Whereas most econometric software packages that include the Akaike and Schwarz statistics appear to use one or the other of the formulae just considered, MicroFit, TSM 4, and TSP adopt an alternative form for each statistic, in certain cases based upon arguments made by their authors (Davidson, 2000; Pesaran & Pesaran, 1987,1991, 1997), namely:

$$\mathrm{AIC} = \mathrm{Ln}(\theta) - \mathrm{p}$$
$$\mathrm{SIC} = \mathrm{Ln}(\theta) - \mathrm{p}/2 \ln \mathrm{n}$$

The specific values obtained by TSP are:

$$\mathrm{AIC} = 96.3137$$
$$\mathrm{SIC} = 97.8073$$

And by TSM4:

$$\mathrm{AIC} = -96.3137$$
$$\mathrm{SIC} = -97.8073$$

which obviously differ only by sign. The relationship between these values and those shown earlier can be expressed by:

$$\mathrm{AIC} = -\frac{T}{2}(1 + \ln 2\pi) - \frac{T}{2}\mathrm{AIC(K)}$$
$$\mathrm{SIC} = -\frac{T}{2}(1 + \ln 2\pi) - \frac{T}{2}\mathrm{SIC(K)}$$

Comparing these formulae to those previously shown, the fundamental issue is whether or not one wishes to measure the effect of additional regressor terms as a matter of "bigger = better," compared to "smaller = better," inasmuch as the results are operationally identical between the alternatives, in terms of their use characteristics except for this aspect. James Davidson, in particular, the author of TSM 4, is "willing to proselytize for this style!" (Davidson 2006, 2008). However, once again, the particular scale of the values produced reflects also the lack of division by the number of observations. The variations just discussed are summarized in Table 4.2.

Table 4.2 Treatment for the Akaike and Schwarz statistics

	Full log likelihood	Omits constants	Divided NOBS	Subtract DF	Change signs	Akaike criterion	Schwarz criterion
Independent software packages							
AREMOS							
AutoBox	X					Log form	Log form
B34S	X					Both forms	Both forms
Betahat	X					Log form	Log form
EasyReg	X					Log form	Log form
EViews	X		X			Log form	Log form
FP							
gretl	X					Log form	Log form
LIMDEP	X					Log form	Log form
MicroFit	X			X		(Log form)	(Log form)
Modeleasy+							
MODLER	X	X	X			Log form	Log form
NLOGIT	X					Log form	Log form
PcGive	X					(Log form)	(Log form)
RATS	X					(Log form)	(Log form)
REG-X	X					(Log form)	(Log form)
SHAZAM	X					(Both forms)	(Both forms)
SORITEC							
Stata	X	X				(Log form)	(Log form)
TROLL							
TSP	X			X		(Log form)	Log form
Wysea							
Econometric programming language applications							
GaussX	X					Log form	Log form
TSM 4	X			X	X	(Log form)	(Log form)

() – indicates that the statistic can be produced at the option of the user.

Of course, the Akaike and Schwarz statistics are not the only Information Criteria to be displayed by the existing econometric software packages. Others that have been implemented in particular cases are the Hannan-Quinn (1979), by EViews (beginning in 2007), PcGive, MicroFit, SHAZAM, and TSM 4, and the Akaike Final Prediction Error (Akaike 1969), by PcGive and SHAZAM. The first of these takes the form:

$$\mathrm{HQC} = \ln \frac{\hat{\mathrm{u}}'\,\hat{\mathrm{u}}}{T} + 2\mathrm{K}\left(\frac{\ln(\ln T)}{T}\right)$$

and the second

$$\mathrm{FPE} = \frac{\hat{\mathrm{u}}'\,\hat{\mathrm{u}}}{T}\,\frac{T+K}{T-K}$$

In the case of PcGive, in the context of the application of the General-to-Specific methodology, these are used to choose between alternative specifications in a class. SHAZAM also offers, in addition to these, the Craven-Wahba generalized cross-validation (Craven & Wahba, 1979):

$$\mathrm{CW} = \frac{\hat{\mathrm{u}}'\,\hat{\mathrm{u}}}{T}\left(1 - \frac{K}{T}\right)^{-2}$$

Rice (1984):

$$\mathrm{RC} = \frac{\hat{\mathrm{u}}'\,\hat{\mathrm{u}}}{T}\left(1 - \frac{2K}{T}\right)^{-1}$$

and Shabata (1981):

$$\mathrm{SHC} = \mathrm{RC} = \frac{\hat{\mathrm{u}}'\,\hat{\mathrm{u}}}{T}\left(\frac{T + 2K}{T}\right)$$

To consider all these choice criteria in sufficient detail so as to bring out their full range of characteristics would require too much space to be attempted here, for the possible benefit obtained (Amemiya, 1980; Judge, 1985; Mizon, 1984). However, it should be recognized in passing that there are a number of aspects that need to be considered by the practitioner in order to choose between them. For instance, in contrast to the comparison of the Akaike and Schwarz criteria that is made by Greene in his textbook, which simply considers arithmetic differences, Davidson (2006, 2008) more fastidiously points out that, when considering the Schwarz and the Hannan-Quinn in comparison with the Akaike, that the first two are consistent selection criteria, implying that if the "true" model is included in the comparison, it will then be selected with probability one as $\mathrm{T} \rightarrow \infty$. Such are the subtleties that lie in wait for the neophyte user, who may or may not be a voracious consumer of textbooks.

Disturbance Properties Tests

Turning attention now to a more familiar evaluative statistic, the Durbin-Watson is of course defined as:

$$\frac{\sum_{t=2}^{T}(u_t - u_{t-1})^2}{\sum_{t=1}^{T} u_t^2}$$

Its limitations are well-known, whether in terms of its strict validity only in the case of nonstochastic regressors, or its bias in the presence of lagged dependent variables, or what Johnston (1984, p. 314–317) has characterized as the "awkward" problem of inconclusive range. In addition, there is also a problem posed when the regression constant term is omitted. As shown by Farebrother (1980), the fundamental pitfall in this case is that the standard Durbin-Watson critical values tables cease to be valid. Specifically, whereas the upper bound is still valid, the lower bound values must be replaced. It is not clear that all programs either automatically omit the test when the constant term is suppressed or else generate and display new lower bound values, but, ostensibly, GaussX, SHAZAM, and TSP each produce exact p-values for the DW statistic. As a special case, this statistic can also function in the context of a co-integrating regression as the CRDW (Co-integrating Regression Durbin-Watson) test statistic; inasmuch as such a regression can be performed as if an Ordinary Least Squares regression (Sargan & Bhargava, 1983), this aspect is marginally relevant here, but certain caveats apply (Banerjee, Dolado, Galbraith, & Hendry, 1993, pp. 206–208).

An additional test statistic, only sometimes found, is the DW-Wallis (1972). It was conceived originally for regressions that involve the use of quarterly frequency data; in this specific case, in contrast to the Durbin-Watson, the constructive difference is that the lag used is instead t-4 and the summation is over $t = 4, 5, \ldots$T. Obviously, it is computationally straightforward to extend its use to other more-frequent-than-annual frequencies, such as, for instance, monthly. However, the appropriateness of this more general use has apparently never been formally investigated. This absence of investigation poses a decision problem for the software developer that can be considered as an example of a common conundrum: tests such as this may be formulated originally for particular contexts, but their use immediately poses the question of what to do more generally; obviously in this case, of what to do when the observational frequency is not quarterly. The choice would appear to be either to hide the statistic or else to compute it with the order defined by the prevailing observational frequency, except of course for annual frequency, in which case it duplicates the Durbin-Watson. When, for example, monthly frequency data are used, it is not at all obvious that maintaining a 4 month lag is generally defensible, since the original motivation was to compare periods a year apart, which might suggest in the monthly case a 12 month lag. However, as indicated, there has been little discussion so far of what is the best approach.

Similarly, Durbin's later test (Durbin, 1970), the so-called Durbin h, developed for use in the presence of lagged dependent variables, does not seem to be as generally implemented as might be expected a priori. The same apparently can be said of what is called Durbin's alternative procedure, which Davidson & MacKinnon (1985) some years ago characterized as "curiously neglected by applied workers," although TSP produces it, as well as the Durbin h, when a lagged dependent variables is included among the regressors. This alternative test was also introduced in the Durbin paper just cited. Furthermore, it can be shown to be a "relation" of the Breusch-Godfrey test, described in a later chapter, which is, under specific conditions, asymptotically equivalent (Johnston & DiNardo, 1997).

There are in addition a number of incidental statistics commonly encountered as what can be regarded as members of the group of core or basic statistics. For example, in the case of the parameter estimates, many if not most packages will optionally produce partial correlation coefficients, a measure of the mean value elasticity and sometimes a standardized parameter estimate, most of which are shown in Fig. 4.5.

Some packages will also display the RSS, the sum of squared residuals; TSS, the total sum of squares; or ESS, the explained sum of squares; or some subset of these, often as a reflection of the tastes of the original program developer or perhaps the particular context in which a given program was originally created and developed. Certain packages will, in addition or instead, implement specific forms of standard statistics, as in the case of TSM 4, which implements the "F-test of the Regression" (testing for zero values of the slope parameters) so as to exclude lagged dependent variables and trend and seasonal dummies (Davidson, 2006, 2008).

Equation Number: 1
Sample Period: 1935- 1954
Number of Observations: 20
Ordinary Least Squares

IGE=C1*FGE(-1)+C2*CGE(-1)+C3

	Parameter	T-Statistic	Std Error	Mean	Elasticity	Partial
C1)	0.02655	1.70571	0.01557	1941.325	0.5039	0.382273
C2)	0.15169	5.90155	0.02570	400.160	0.5934	0.819758
C3)	−9.95631	−0.31734	31.37425	1.000	−0.0973	−0.07674

Variance=	777.446299	Dependent Mean=	102.290
Standard Error=	27.882724	Percent Error=	27.26%
Explained SS =	31632.0321	UnCntrd R-Square=	0.9480
R-Square=	0.7053	R-Bar-Square=	0.6706
Schwarz Criterion=	6.9429	Akaike Criterion=	6.7935
Max LLF=	−93.31373	F-Statistic: F(2, 17)=	20.3435
DW Statistic=	1.07210		

Glossary of Variables

CGE	General Electric capital stock
FGE	General Electric Market Value at end-of-year
IGE	General Electric, Annual Investment Expenditures

Fig. 4.5 Expanded regression display: Incidental statistics

A general consideration that the foregoing discussion raises is that of the proper forum in which to consider matters such as these as time goes on. There is no obviously sanctioned place. Seen from the point of view of the editor of an econometrics journal, each of these individual issues is unlikely to be considered as worthy of even a note, especially whenever considered alone. Likewise, econometrics textbooks have not, as yet, begun to consider evaluatively the design decisions that have been made by econometric software developers and might never. On their part, software developers have characteristically made these decisions very quietly, often without calling users' attention to the particular choices that have been made. In turn, software reviewers have usually passed over such issues in silence. However, as this discussion demonstrates, not only are some of these matters possibly individually significant, but the cumulative effect of all these decisions made implicitly by developers does determine what the applied economist can do and therefore has an effect on the practice of applied economic research. Furthermore, only some of the possible examples have been considered here. There are other bodies buried in the orchard.

Provisions for User-Constructed Tests

The foregoing presentation of the so-called core statistics has focused specifically on their role as a set of basic regression statistics for a candidate economic specification, which in this role display certain properties. However, they do not necessarily have only this function, for the linear regressions that may be performed by a knowledgeable user are not necessarily limited to only those that can be classified as economic-content equation specifications. It is also possible to perform supplementary linear regressions – sometimes called *artificial regressions* – that cause these basic statistics to be interpreted in a wider evaluative context. Their use in this role carries with it certain software design implications for the future.

In particular, whenever a program provides the capability to save such generated values as $\hat{\mathbf{y}}$ and $\hat{\mathbf{u}}$:

$$\hat{\mathbf{y}} = \mathbf{Xb} \quad \text{and} \quad \hat{\mathbf{u}} = \mathbf{y} - \mathbf{Xb}$$

It is in principle possible for the user to employ these values purposefully to form diagnostic tests. The textbook by Davidson & MacKinnon (1993) is a particularly relevant reference. There they discuss in some detail (in Chaps. 6 and 11, specifically) a number of circumstances in which it is desirable for the user of a software package to be able to perform this so-called Gauss-Newton regression (GNR), using values generated previously, in order to achieve potentially a variety of distinct diagnostic goals, among them being, as these authors point out:

1. In the case that the original regression is a nonlinear regression, to verify that the first order conditions for a minimum or maximum are satisfied sufficiently accurately
2. To calculate estimated covariance matrices
3. To calculate a test statistic after a specification has been estimated subject to restrictions, without ever estimating the unrestricted form
4. To calculate one-step efficient estimates
5. As a key part of procedures for numerical optimization that are used to find nonlinear least squares and other types of estimates
6. To form tests for the equality of two (or more) sets of regression parameters
7. To form non-nested hypothesis tests, in which a regression specification is tested against the evidence provided by one or more non-nested alternatives
8. To form tests based on comparing two sets of estimates, where generally one set is consistent under weaker conditions than the other
9. To form tests for heteroscedasticity of known form

However, to achieve these goals effectively may impose certain design demands. In particular, when considering an economic specification, core regression statistics such as the joint parameter F-Test and the R^2 can each be viewed as being only rather minimally informative. In contrast, in the context of an auxiliary regression, these statistics take on an enhanced role that makes it pertinent to consider them in a somewhat different light, and which because of this role also makes their provision a constructive software design issue. For example, to permit auxiliary regressions to be performed most usefully the core statistics produced should include the uncentered R^2 statistic, R_u^2, defined as:

$$\mathrm{R}_\mathrm{u}^2 = \frac{y' X(X' X)^{-1} X' y}{y' y}$$

and which obviously uses as the denominator the total variation in the dependent variable, relative to the origin. In contrast the (more common) centered $R^2$is defined by:

$$\mathrm{R}^2 = \frac{y'\,X(X'\,X)^{-1}X'\,y - (i'\,y)^2/T}{y'\,y - (i'\,y)^2/T}$$

where i' is the transpose of the unit vector. Evidently, if the dependent variable of the GNR regression has a zero mean, then the centered and uncentered R^2 values will be identical; however in general these two statistics will differ.

Given the appropriate definition of the R^2 statistic for the particular case, when a GNR is performed as a supplementary regression, it is well-known that the value:

$$\mathrm{TR}^2 \sim \chi^2(\mathrm{p})$$

provides, in a number of important cases, a test statistic the precise interpretation of which depends upon the particular dependent and regressor variables used. Alternatively, when the regression includes a constant term, it is possible to form:

$$\frac{R^2}{1-R^2}\frac{T-K}{K-1} \sim \mathrm{F}(\mathrm{K}-1,\ \mathrm{T}-\mathrm{K})$$

inasmuch as it is straightforward to show that this expression is algebraically the same as:

$$\frac{ESS/(K-1)}{RSS/(T-K)}$$

When relevant, this alternative statistic may behave better in a small sample context. Of course, from a software design perspective, it needs to be considered also that whenever a user must make any such transformations the number of significant digits post-transformation will depend upon the precision of the program's displayed statistics that are used.

In addition to the Davidson and MacKinnon chapters (Davidson & MacKinnon, 1993, an earlier paper by MacKinnon (1992) in the *Journal of Economic Literature*, about the formation of statistical tests based upon auxiliary regressions, demonstrates the method's generality inasmuch as the test statistics thereby generated are exactly equivalent, in a number of cases, to those created by other differently conceived evaluative approaches. As a follow on from such considerations, what is presently particularly relevant is that if a program automatically provides an appropriate set of core statistics this can obviously enhance its usability, possibly permitting the econometrically knowledgeable user who is not particularly adept as a computer user to formulate diagnostic tests beyond those offered by the program. Of course, there is still the issue that formulating tests in this fashion, one after the other, will always take the user's time, compared to the capability simply to review a comprehensive set of displayed diagnostic statistics. There are also additional software design issues (Renfro, 1997). It is nevertheless important to have touched on these several points simply to provide a reminder that the core statistics potentially have a role wider than just a set of passively displayed statistics of an economically-defined specification.

Chapter 5
The Failure of Assumptions

The point was made late in Chap. 4 that the facilities a software package might provide to perform OLS regressions extend beyond the capability to estimate the parameters of economic specifications to include also auxiliary regressions, executed specifically in order to permit the user to perform various misspecification tests. However, if the possibility for the user to construct these is for the moment ignored, the supporting statistics considered as of the end of that chapter can be viewed as displaying certain properties of the specification and the parameter estimates, but without necessarily providing the means to evaluate to any comprehensive degree the appropriateness of either the estimation technique or the chosen specification. The properties of these statistics are nonetheless predicated upon certain assumptions.

More formally, let it be assumed in the usual way that:

(A1) $\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{u}$
(A2) $\mathbf{u} = \mathrm{iid}(0, \sigma^2\mathbf{I})$
(A3) $\rho(\mathbf{X}) = \mathrm{k}$
(A4) $\mathrm{E}(\mathbf{X}'\mathbf{u}) = 0$

where:

$\mathbf{y}$ – a vector of T observations on a variable y
$\mathbf{X}$ – a matrix of T observations on k regressor variables
$\boldsymbol{\beta}$ – a vector of k unobserved constant parameters
$\mathbf{u}$ – a vector of T unobserved disturbances

Given the last four definitions, it is immediately evident that the first assumption, A1, incorporates not only that the given specification is linear-in-parameters, but also that the parameters are constants. The second assumption, A2, incorporates not only that the expected value of the disturbances are zero and that the disturbance variance-covariance matrix is diagonal, but also that the disturbance variance is homoscedastic. The third assumption, A3, states that the regressor variables are not linearly dependent. The final assumption, A4, states the distributional independence of the regressors and the disturbances.

However, before considering specifically each of the sets of tests that are provided by the existing econometric software packages in order to permit their

C.G. Renfro, *The Practice of Econometric Theory*, Advanced Studies in Theoretical and Applied Econometrics 44, DOI 10.1007/978-3-540-75571-5_5,

users to probe the possible failure of these assumptions, it is important to consider briefly a fundamental computational requirement. One way to look at these assumptions collectively is to say that they state the specific characteristics of the particular parameter estimates that, with qualification, are *computationally imposed* by the OLS estimator, assuming that in practice it is actually possible to invert the **X'X** matrix. This statement incorporates a distinction that is similar to that between the average of a set of numbers and the estimated mean of the distribution from which these numbers are drawn: that is, computationally, any set of observed numbers can be averaged, but it is a further conceptual step to say what that average value actually represents; the reason of course is that the mean, as a measure of central tendency, may or may not exist for the distribution from which the sample is drawn. However, in the case of the OLS parameter estimates, the issue is not simply the meaning of the calculation, but also whether it can be performed at all: an additional particular, computational issue is that in order to compute OLS parameter estimates uniquely – whatever their conceptual properties – the fundamental requirement, given a data sample, is the capability to invert the **X'X** matrix.

This inversion is not always possible: not only may the particular sample not permit it, as a matter of principle – stemming from the characteristics of the population from which the data sample is drawn, but in practice even if the rank requirement *is* met – inversion may not be possible in a way that provides unique and numerically accurate estimates. This last qualification is not usual in the theoretical econometrics literature, because of the implicit assumption that with probability one, if the rank of **X** *is* k, then it will be computationally possible to invert this matrix. Although it has always been recognized that the sample may involve a rank deficiency, the potential impossibility of meaningfully inverting the **X'X** matrix *when the rank of X is k* has been glossed over as merely computational, with econometricians secure in the knowledge, as demonstrated by Theil (1971), that under the conditions of the standard linear model, the matrix [**y X**] has the "rank $k + 1$ with unit probability, provided that $T > k$ and that the disturbances have a continuous distribution" (p. 166).

However, *as a matter of computation*, the determination that this matrix can be meaningfully inverted is fundamentally a consequence of the precision tolerances set by the computational context. In particular, recall that Longley (1967) demonstrated that the ostensible ability to invert this matrix neither guarantees the computational accuracy of the parameter estimates obtained nor insures that the observations used are not highly linearly dependent. Much more recently, Stokes (2005) has considered the effects of both the way in which the calculations are performed and the precision with which the observations are stored in the computer. The two computational issues essentially involved are, first, the degree of rounding error associated with the computational methods used and, second, the representational precision of the computer-stored machine numbers as approximations to the set of real numbers. These are issues even in the absence of measurement error, but of course they are greatly accentuated in the event that the economic data exhibit a low signal to noise ratio, as discussed in the last sections of Chap. 1. The study that this volume describes did not specifically evaluate the numerical accuracy of each of the individual econometric software packages surveyed, although other studies have evaluated at least certain of them, with increasing frequency (Brooks, Burke, & Persand, 2001; McCullough, 1997, 1999, 2000, 2004; McCullough & Renfro, 1998; McCullough, Renfro, & Stokes, 2006; Stokes, 2004a). Numeric accuracy is nonetheless a fundamentally important, if implicit, issue here, not least because of all the possible evaluative tests that might be incorporated into an econometric software package to test assumptions about the specification, the assumption of numeric accuracy is one for which no unassailable test is possible that involves only a computational procedure within that program. Recall the argument made in Chap. 1, although it is of interest that Stokes (2008) has recently demonstrated that the degree to which $\mathbf{X}'\hat{\mathbf{u}} = 0$ does provide a test that can be used in order to confirm the quality of the inversion of the **X'X** matrix.

As Ozturk & Akdeniz (2000) have pointed out, even short of the possible inability to invert the regressor cross products matrix, because of the solution instability in the face of small changes – that is, because the outputs are unduly sensitive to small changes in the inputs – ill-conditioning actually poses a three-stage problem. In this circumstance, as a first-stage issue, it potentially becomes critical for the distribution(s) of the data measurement errors to be known, or at least to a significant degree understood. As a second-stage issue, there is the problem that, in this context, the results become much more sensitive to the error characteristics of the specification, including the statistical optimization criteria used. Finally, the third-stage issue is that just discussed, namely the numerical analytic problem associated with the specific calculations that are made. Recall from the discussion in Chap. 1 that, in the presence of ill-conditioning, measurement error that might otherwise cause the estimates to be inefficient, relative to the situation in their absence, possibly could cause the parameter estimates to be catastrophically in error.

Furthermore, as was mentioned, numerical accuracy is not only an issue with respect to parameter estimates, but also as regards the statistical tests that are the subject of the survey reported here. A general question exists always whether the supporting statistics displayed by a given program are correctly implemented, both in the sense of a proper implementation of the formulae associated with a recognized test statistic and the numerical accuracy of the underlying algorithmic representation. Actually, one of the findings of the present survey, referred to earlier and to be discussed later, is that there is sufficient ambiguity concerning exactly how particular statistical tests should be computationally implemented that different packages commonly present variant forms of what, by name, are ostensibly the same test. The decisive circumstance in these instances is not necessarily computational, but can instead be constructive. It has, however, been demonstrated by McCullough & Renfro (1998) and subsequently confirmed by Brooks et al. (2001), specifically in the case of nonlinear estimation, that seemingly difficult implementation problems can occur that are directly related to the numerical accuracy of the algorithms sometimes used by econometric software packages and, as well, to the particular user provisions each program makes. These potential problems are perhaps less threatening in the cases considered here, because of a lesser computational complexity, but they are not altogether absent.

Heteroscedasticity

In earlier years, heteroscedasticity was seen as being possible but somewhat unlikely in a time series context. More recently, in an ARCH, and indeed GARCH-aware world, the idea that it is fundamentally to be found in the context of cross-section or, in some cases, grouped (panel) data applications has come to appear quite dated. Even as late as the 1984 edition of Johnston, or in the case of Amemiya (1985) the next year, the old mindset was dominant, although as early as 1971 the sometimes iconoclastic Theil (1971) opened his initial discussion of testing for heteroscedasticity with a direct reference to "a time series regression whose disturbances are to be tested against the hypothesis that that variances are not constant..." (p. 196). Today, reflecting the recognition that this condition can occur as a consequence of misspecification of the relationship among variables or arise even from log transformations (Ermini & Hendry, 2008), most econometric software packages now present tests against it more or less automatically. This change is indicated by Table 5.1 and it is evident from this table that the most popularly implemented tests are the Bruesch-Pagan, Goldfeld-Quandt, ARCH process tests, and White's, with some packages offering at least two of these automatically.

Of these tests, Goldfeld-Quandt (G-Q) is the granddaddy. Introduced in 1972 (Goldfeld & Quandt, 1972), this test essentially presupposes that a particular regressor variable is responsible for the heteroscedasticity and relies on the ability of the practitioner to divine or otherwise determine which one might be the culprit. Given that the particular variable can be identified, the sample regressor data set is then reordered so that the observations on all the variables are ordered in conformity with the numerical values observed for this variable. Following this reordering, some integer number $c \geq 0$ of the central values of the reordered data set are excluded and two separate OLS regressions are then performed using the first and last $(T - c)/2$ observations of this newly ordered data set, or sometimes the T_1 first and T_2 last observations such that $T_1 + T_2 + c = T$. The test statistic, a ratio, is calculated using as the numerator the residual sum of squares from the first regression and the denominator that of the second regression. The power of the test depends in part upon the choice of an appropriate value for c. The Glesjer test (Glesjer, 1969; Goldfeld & Quandt, 1972) is at times coupled with the G-Q, since it can be used to assist in the discovery of a particular regressor variable as the source of heteroscedasticity.

The Breusch-Pagan test (Breusch & Pagan, 1979; Godfrey, 1978), in contrast, is based upon a characterization of the disturbances u_t as being independently, asymptotically normally distributed (or even normally distributed for all sample sizes), such that the variance is some function $f(\mathbf{z}_t'\boldsymbol{\alpha})$. Commonly, as described by Johnston and DiNardo (p. 46, 167ff), the test can involve constructing a standardized variable using the residuals from the original regression which are then regressed on some if not all the original regressor variables, and perhaps others. The explained sum of squares, divided by 2, is asymptotically Chi-square distributed, and this value is therefore used as the test statistic. More generally, the Breusch-Pagan test statistic can be expressed either as a Lagrange Multiplier test or in the form of an F-statistic, although not all packages display both. It can also be applied more extensively than just when the original parameters are estimated by OLS; that is, provided that

Table 5.1 Heteroscedasticity tests

	Breusch-Pagan	ARCH process	Goldfeld-Quandt	White's test	Glesjer	Box-Cox	Tsay test	Harvey	Cameron Trivedi IM test	Szroeter's rank test	LM Het
Independent software packages											
AREMOS											
AutoBox						(X)	(X)				
B34S	(X)	(X)	(X)	(X)	(X)		(X)				
Betahat	(X)	(X)			(X)			(X)			
EasyReg	X	(X)									
EViews	(X)	(X)		(X)	(X)			(X)			
FP											
gretl	(X)	(X)		(X)							
LIMDEP	X	X	(X)								
MicroFit	(X)			(X)							(X)
Modeleasy+	(X)	(X)	(X)								
MODLER	X	X	(X)		(X)						
NLOGIT	X	X	(X)								
PcGive		(X)		(X)							
RATS	(X)	(X)	(X)	(X)							
REG-X		(X)									
SHAZAM	(X)	(X)	(X)	(X)	(X)	(X)		(X)			
SORITEC											
STAMP											
Stata	(X)	(X)		(X)					(X)	(X)	
TROLL											
TSP	(X)	(X)		(X)							X
Wysea											
Econometric programming language applications											
GaussX	(X)	X	(X)	X							
TSM 4	(X)	(X)#									

() – indicates that the statistic can be produced at the option of the user.
ARCH process test – The "#" indicates that the corresponding M-tests are also optionally produced.

appropriately constructed residuals are used, the test can also be applied in the context of Instrumental Variables, Limited Information Maximum Likelihood and other asymptotically equivalent estimators (Godfrey, 1988; Kelejian, 1982). Even from this thumbnail description, it should be clear that only to the degree that two or more econometric software programs permit their users to control the selection of the $\mathbf{z}_t$, or else predetermine the selection in exactly the same way, does the possibility exists of constructional equivalence. In fact, even for a single program, there can be considerable variety, with some offering their users optional variations. Stata, for example, will perform the test for each of the regressors or any other variables, and will optionally adjust the p-values reported for each variable to account for multiple testing. Sometimes, users can also choose among several multiple testing adjustments.

The ARCH process tests of course originate in the observation by Robert Engle (1982) that heteroscedasticity in some contexts can be a time series phenomenon due to a clustering of errors with similar characteristics, particularly in the case of prices and other variables in the context of a speculative market. One way to construct an ARCH process test is to regress the square of the residuals from an original regression on lags of itself, and then test the joint significance of the parameters from this regression. Even in this case, programs can generate different test results either due to the choice of the number of lagged squared residuals used or because of the omission of lags of particular orders. The ARCH process tests also can generally be stated in Lagrange Multiplier or F-test form.

The original White test, introduced in 1980 (White, 1980), is formulated by regressing the squared residuals from the original regression on a constant and the set of unique variables in the cartesian product $X \times X$. The test statistic is obtained as TR^2 from this regression or can be transformed into F-Test form. The test is obviously quite general, requiring little in the way of an a priori statement as to the source of the possible heteroscedasticity, but this generality is often seen as a shortcoming (Greene, 2000). Commonly perceived specific defects of this test are that it may have low power and that it is econometrically nonconstructive, providing in itself no strategy for further testing if the hypothesis of homoscedasticity is rejected. Reflecting such problems, packages that include White's test sometimes offer supplementary facilities in order to buttress its use.

As shown in Table 5.1, other more occasionally implemented heteroscedasticity tests include the Box-Cox, Harvey (1976), and Tsay tests (Tsay, 1986, 2002), and Cameron and Trivedi's decomposition of White's Information Matrix (IM) test (Cameron & Trivedi, 1990, 1993; White, 1987), as well as Szroeter's rank test, the latter of which can also be implemented in Lagrange Multiplier form (Szroeter, 1978). Another uncommon test is the so-called LM Het test implemented by MicroFit and TSP. It was apparently originally introduced in the MicroFit user guide. The basic model, such that the variance is proportional to the square of the fitted values, can be traced to Prais & Houthakker (1955, p. 53ff). Amemiya has also studied the model (Amemiya, 1973). Both Davidson & MacKinnon (1993) and Godfrey (1988) provide useful general discussions of heteroscedasticity tests and related issues.

Benchmark values are shown in Fig. 5.1 for the tests presently implemented in existing econometric software packages. These are calculated using the Theil-Grunfeld data. For several of the named tests, groups of values are displayed, the members of which demonstrate the effect of specific variations in the way in which the particular statistic, or variant statistic, can in practice be calculated. The notation "using t," on the right, refers to the use of the values $t = 1, 2, \ldots, T$. In each case, the test can be seen as involving conceptually a regression of the chosen regressand (which itself may vary) on these particular values. The notation "using X" refers to the use of the regressor variable values of the original regression. The notation "using X^2" refers to the use of squared values of the original regression regressor variables in the calculation of the test statistic. In turn, "using X, X^2" refers to the use of both the original regressor variables and their squares and, finally, "using X, X^2, and cross-X" refers to the use of the *unique* cross-products of these variables in addition.

It may be pedagogically helpful, in each case, to view the test statistic as being generated by an auxiliary regression that uses one of these sets of values as the

Original Form:				
Breusch-Pagan Test:	F(1, 18)=	0.0808, Chi-Sqr(1) =	0.0894	using t
Breusch-Pagan Test:	F(2, 17)=	0.1242, Chi-Sqr(2) =	0.2881	using X
Breusch-Pagan Test:	F(4, 15)=	1.2353, Chi-Sqr(4) =	4.9560	using X,X^2
Koenker-Bassett Form:				
Breusch-Pagan Test:	F(1, 18)=	0.0958, Chi-Sqr(1) =	0.1059	using t
Breusch-Pagan Test:	F(2, 17)=	0.1475, Chi-Sqr(2) =	0.3412	using X
Breusch-Pagan Test:	F(4, 15)=	1.5574, Chi-Sqr(4) =	5.8687	using X,X^2
White Test:	F(4, 15)=	1.5574, Chi-Sqr(4) =	5.8687	using X,X^2
White Test:	F(5, 14)=	1.3924, Chi-Sqr(4) =	6.6425	using X,X^2 and cross-X
Engle ARCH Test:				
ARCH Process Test:	F(1, 17)=	0.4098, Chi-Sqr(1) =	0.4472	Lag 1
ARCH Process Test:	F(2, 16)=	0.1881, Chi-Sqr(2) =	0.4405	Lag 2
ARCH Process Test:	F(3, 15)=	0.0663, Chi-Sqr(3) =	0.2560	Lag 3
ARCH Process Test:	F(4, 14)=	0.0913, Chi-Sqr(4) =	0.5143	Lag 4
Goldfeld-Quandt (Sorted on FGE):				
Goldfeld-Quandt Test:	F(8, 8)=	1.4450		c=0, T_1=10, T_2=10
Goldfeld-Quandt Test:	F(7, 7)=	1.7171		c=2, T_1= 9, T_2= 9
Goldfeld-Quandt Test:	F(6, 6)=	2.3811		c=4, T_1= 8, T_2= 8
Goldfeld-Quandt Test:	F(5, 5)=	3.3203		c=6, T_1= 7, T_2= 7
Goldfeld-Quandt Test:	F(4, 4)=	2.3179		c=8, T_1= 6, T_2= 6

Fig. 5.1 Heteroscedasticity tests

regressors. However, this auxiliary regression should be regarded as being potentially surrogate, inasmuch as the calculation of the test statistic in the context of a given econometric software package can be carried out without employing any such regression, yet yield the same test statistic value. All references made here to auxiliary regressions should be understood to have been made simply in order to provide a conceptual aid, not as an indication that any software package necessarily generates the given test statistic via such a regression.

Considering individually each of the tests that are implemented, and reading from the top of the Fig. 5.1 display, the Breusch-Pagan test is obviously the initial one and is shown in two forms. The first is that described by Greene in his text book (Greene, 2000, p. 510ff) as *the* Breusch-Pagan test. The second is that used by MODLER, RATS, and TSP in particular, and involves a modification that removes the need for the normality assumption and may yield, at the cost of a certain degree of power, large efficiency gains when the distribution is not normal. It is referred to by Greene as the Koenker and Bassett variant (Koenker & Bassett, 1982) – notice in particular that the χ^2 form of this test statistic is computed as T times the *centered* R^2 from the appropriate auxiliary regression. Furthermore, the test in this form is sometimes labeled as a White test (White, 1980), inasmuch as it has been shown by Waldman (1983) that if the variables in $\mathbf{z}_t$ are the same as those used by White, then the two tests yield the same values; Fig. 5.1 explicitly demonstrates this equivalence. Finally, it is also true that in the literature the Breusch-Pagan test is variously referred to by this name (Herwartz, 2006) or else as the Breusch-Pagan/Godfrey (Greene, 2000, 2003) or even the Godfrey/Breusch-Pagan (MacKinnon, 1992).

When White's test is presented, it is generally a particular variant that is provided, although EViews provides the full range. The first variant, which involves the use of X and X^2, is offered by B34S, PcGive and SHAZAM; it is this variant, as mentioned (and demonstrated in Fig. 5.1), that can be identified as the same as a Breusch-Pagan test. PcGive, SHAZAM, Stata, and TSP each offer the second variant, constructing it using X, X^2, and the (unique subset of) cross-product terms, here limited simply to the product of lagged FGE and lagged CGE, but in the case of PcGive the test is available only when the number of observations is sufficiently large, in particular when $T >> K(K+1)$. Furthermore, not only does PcGive require there to be a large number of observations, relative to the number of regressors, but it also explicitly presents this variant as based upon an auxiliary regression, using the squared residuals as the regressand. In addition, as well as reporting χ^2 and F-values, this program shows also individual coefficient t-statistics, in order "to help with model respecification" (41, p. 162). There are also certain associated qualifications and caveats: versions of PcGive prior to version 10 presented this test as being a functional form misspecification test, but this is no longer done, in response to criticism by Godfrey & Orme (1994). The manual for SHAZAM, version 10 (Whistler, White, Wong, & Bates, 2004), notes that the "general" White's test is"not defined" when dummy variables are included among the regressors, but it does not indicate if the test is automatically suppressed in this circumstance (albeit rather difficult to program) or if the user is expected to recognize the condition and issue SHAZAM's NOWHITE command. The important issue here is that, in the presence of dummy variables, the squares and cross-products can be collinear with other variables. For this reason, TSP adjusts the degrees of freedom when dummy variables are included.

The Engle ARCH test is the next shown in Fig. 5.1, displaying test values that are calculated using the residuals of the original regression, involving progressively from one to up to a total of four lag values of these residuals. In each case, this test too can be conceptually visualized as involving an auxiliary regression in which these values are the regressors, although as indicated earlier, in practice the test statistic will not necessarily be generated from such a regression by any given software package. Furthermore, not every package displays a test statistic corresponding to each of the four different lag configurations. Some packages set a default and allow no change. Others permit the user to select the order of the lag.

Evidently, there are a number of variations displayed as test statistics that go by the names Breusch-Pagan, Arch Process and White's. Similarly, in each particular case, the Goldfeld-Quandt test can also involve any of a number of substantive constructive issues (Renfro & Stokes, 2009). For example, depending upon the total number of observations, it is rather easy for developers of different packages to adopt slightly or even substantially different conventions for the creation of the two data sub-sets used to form the test. Even given a common choice of c, the number of sorted observations omitted can vary, inasmuch as the subsets of observations included can each be the same in number or can differ. Furthermore, the documentation of most packages does not state exactly what occurs if and when T-c is an odd number. In addition, it is not usual for a package to identify clearly the number of omitted values, although it is generally possible for the knowledgeable user to deduce this. Only certain packages, one of them being B34S (Stokes, 1997), permit the user to specify separately the number of observations to be included in each of the two sub-samples. Test statistic values are shown corresponding to a range of values of c, and it may be possible to infer from these that the power of the test is reduced both for "too small" and "too large" values of c.

There is yet another issue that needs to be addressed immediately, and that is the particular evaluative place of heteroscedasticity tests in the pantheon of statistical tests, or more precisely the question of the design of software packages in the possible presence of heteroscedasticity. Indeed, one of the reasons to have considered these tests first is the particular role that estimates of the sample variance and standard error play in the formulation of a variety of test statistics, as will be discussed. However, in certain specific cases there is also an issue of how best to design software packages so as to be naturally heteroscedastically robust. Consider the OLS parameter covariance matrix, as commonly presented in the literature:

$$\mathrm{Var}(\mathbf{b}) = \mathrm{s}^2(\mathbf{X}'\mathbf{X})^{-1}$$

where b denotes the estimated value of $\boldsymbol{\beta}$ and s^2 the sample error variance estimate:

$$\mathrm{s}^2 = \frac{\hat{\mathrm{u}}'\hat{\mathrm{u}}}{T - K}$$

This way of representing the matrix also appears to have been commonly implemented computationally by software developers since the 1960s – because of the ease of it. However, as Harvey, for instance, points out (39, p. 65), when the disturbances are heteroscedastic, but serially independent, a consistent estimator of the parameter covariance matrix is:

$$\mathrm{Var}(\mathbf{b}) = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\hat{\mathbf{U}}\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}$$

which of course can be computed directly, using the sample residuals and the sample observations on the regressor variables. If it happens, in addition, that:

$$\mathbf{U} = \sigma^2\mathbf{I}$$

then, *in principle*, computing this matrix one way rather than the other amounts to the same thing, albeit involving a sample, rather than population result.

It should, however, be noted specifically that the term "in principle", as used here, has a specific computational aspect, having to do with the issue of the particular computations performed and rounding error. Whereas $(\mathbf{X}'\mathbf{X})^{-1}$ is a direct by-product of the computation of the OLS estimates of $\boldsymbol{\beta}$, the computation of $\mathbf{X}'\hat{\mathbf{U}}\mathbf{X}$ involves an extra set of (rounding error prone) computations, as do also the successive matrix multiplications $(\mathbf{X}'\mathbf{X})^{-1}$ times $\mathbf{X}'\hat{\mathbf{U}}\mathbf{X}$ times $(\mathbf{X}'\mathbf{X})^{-1}$, even if there is in fact a way to simplify this computation, and even if today there is no longer the issue that existed, say, in 1965, of the computational burden the calculation imposes. It is also true – deriving from the sample properties of the residuals and the regressor observations – that it cannot be expected that the alternative ways of computing Var(**b**) will in practice yield the same main diagonal values of this matrix, which of course are used to compute t-statistics and for the standard errors used in the displayed regression results. Therefore, it is not necessarily true that, in every case, the long-standing methods of computation should be changed, but this example does serve to illustrate one more instance in which computational considerations are important to the applied research practices – and have heretofore been largely ignored. It is obviously true that, for example, it would be a small matter, in terms of the design of an econometric software package, to change the method of computation of Var(**b**), depending upon some "decision value", so that one method of computation is used in one instance and another otherwise, with of course provision made for the user to be notified and then have the capability to override making this calculation.

Disturbance Properties

Generally, the implemented tests of disturbance properties, displayed in Fig. 5.2, are predicated on a homoscedastic variance. These tests range quite widely, certain of them being tests of normality, others being tests, even general tests, against both autoregressive or moving average disturbance processes. The latter, the autoregressive tests, can be viewed as extensions of the basic Durbin-Watson, which, as Greene points out (33, p. 540), although powerful for AR(1) processes, has among its limitations insensitivity to processes of other orders.

In contrast, the Breusch-Godfrey (B-G) test is of course less restrictive, testing against both general autoregressive and moving average processes. The LM form of the B-G statistic can be specified as:

$$\mathrm{LM} = \hat{\mathbf{u}}'\mathbf{U}[\mathbf{U}'\mathbf{U} - \mathbf{U}'\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{U}]^{-1}\mathbf{U}'\hat{\mathbf{u}}/(\hat{\mathbf{u}}'\hat{\mathbf{u}}/\mathrm{T})$$

where:

X = the matrix of sample data on all the regressor variables of the original regression
$\hat{\mathbf{u}}$ = the vector of residuals from that regression
T = the number of sample period observations on each variable

and

$$\mathbf{U} = \begin{bmatrix} 0 & 0 & \ldots 0 \\ \hat{u}_1 & 0 & \ldots 0 \\ \hat{u}_2 & \hat{u}_1 & \ldots 0 \\ \hat{u}_3 & \hat{u}_2 & \ldots 0 \\ \vdots & \vdots & \vdots \\ \hat{u}_{T-1} & \hat{u}_{T-2} & \hat{u}_{T-p} \end{bmatrix}$$

is the matrix of lagged residuals, where p can be arbitrarily set.

```
Jarque-Bera Original Form:
Jarque-Bera Test:                              Chi-Sqr( 2) =    1.9273

Jarque-Bera Standardized Variables:
Jarque-Bera Test:                              Chi-Sqr( 2) =    2.1667

Box-Pierce, Progressive Lags
Box-Pierce Test:                               Chi-Sqr( 1) =    4.2480
Box-Pierce Test:                               Chi-Sqr( 2) =    6.0280
Box-Pierce Test:                               Chi-Sqr( 3) =   13.1188
Box-Pierce Test:                               Chi-Sqr( 4) =   14.0712
Box-Pierce Test:                               Chi-Sqr( 5) =   15.6443
Box-Pierce Test:                               Chi-Sqr( 6) =   17.6923
Box-Pierce Test:                               Chi-Sqr( 7) =   17.7289
Box-Pierce Test:                               Chi-Sqr( 8) =   19.7110

Ljung-Box, Progressive Lags
Ljung-Box Test:                                Chi-Sqr( 1) =    4.9187
Ljung-Box Test:                                Chi-Sqr( 2) =    7.0946
Ljung-Box Test:                                Chi-Sqr( 3) =   16.2706
Ljung-Box Test:                                Chi-Sqr( 4) =   17.5801
Ljung-Box Test:                                Chi-Sqr( 5) =   19.8874
Ljung-Box Test:                                Chi-Sqr( 6) =   23.1056
Ljung-Box Test:                                Chi-Sqr( 7) =   23.1676
Ljung-Box Test:                                Chi-Sqr( 8) =   26.8015

Breusch-Godfrey With Missing Lagged Residual Values = 0
Breusch-Godfrey Test:    F( 1, 16)=    5.3331,  Chi-Sqr( 1) =    4.9998
Breusch-Godfrey Test:    F( 2, 15)=    9.2990,  Chi-Sqr( 2) =   11.0709
Breusch-Godfrey Test:    F( 3, 14)=    6.0692,  Chi-Sqr( 3) =   11.3064
Breusch-Godfrey Test:    F( 4, 13)=    4.4953,  Chi-Sqr( 4) =   11.6078

Breusch-Godfrey with Missing Lagged Residual Values Suppressed
Breusch-Godfrey Test:    F( 1, 15)=    5.0267,  Chi-Sqr( 1) =    4.7690
Breusch-Godfrey Test:    F( 2, 14)=    8.9163,  Chi-Sqr( 2) =   10.0836
Breusch-Godfrey Test:    F( 3, 13)=    6.0038,  Chi-Sqr( 3) =    9.8736
Breusch-Godfrey Test:    F( 4, 12)=    5.5421,  Chi-Sqr( 4) =   10.3808

Shapiro-Wilk Test:                              Chi-Sqr(1) =    0.9240
```

Fig. 5.2 Tests of disturbance properties

A slightly different, but sometimes useful perspective from which to view the above B-G test statistic proceeds from recalling that

$$\hat{\mathbf{u}}'\mathbf{U}[\mathbf{U}'\mathbf{U} - \mathbf{U}'\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{U}]^{-1}\mathbf{U}'\hat{\mathbf{u}}/\hat{\mathbf{u}}'\hat{\mathbf{u}}$$

is in principle arithmetically identical to the computed R^2 from the regression of $\hat{\mathbf{u}}$ on the lagged values of itself and the regressor, $\mathbf{X}$, from the original regression, thus on the expanded data matrix $[\mathbf{U}\ \mathbf{X}]$, so that the B-G test statistic can be equivalently expressed by TR^2. As mentioned earlier, there is also a close mathematical correspondence between this statistic and the one proposed in 1970 by Durbin (1970) for use in the context of regressions containing lagged values of the dependent variable, which can be expressed by

$$\hat{\mathbf{u}}'\mathbf{U}[\mathbf{U}'\mathbf{U} - \mathbf{U}'\mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{U}]^{-1}\mathbf{U}'\hat{\mathbf{u}}/\mathrm{p}/(\hat{\mathbf{u}}'\hat{\mathbf{u}}/[\mathrm{T} - (\mathrm{p} + \mathrm{r} + \mathrm{s})])$$

where r is the order of the lag on the dependent variable in the original regression, s is the number of other regressor variables in that regression and p is the order of the disturbance autoregressive scheme tested against. Breusch (1978) has shown the asymptotic equivalence of the B-G and Durbin test statistics.

As is indicated by Table 5.2, the Breusch-Godfrey test is commonly implemented in econometric software packages. However, this implementation can differ among packages. For instance, MODLER and RATS have historically implemented this test in the manner described in Greene's earlier quoted footnote, namely by omitting the first p (zero filled) lagged values of the residuals. In contrast, Stata offers the user the choice between omitting these values or using them. In the light of this practice, the question is, are these variants equally defensible? In response, it can be

Table 5.2 Residuals properties tests for time series regression

	Jarque-Bera	Box-Pierce	Ljung-Box	Breusch-Godfrey	Shapiro Wilk	Bartlett	Shapiro Francia	Doornik Hansen	Harvey LM
Independent software packages									
AREMOS				(X)					
AutoBox		X	X						
B34S	(X)	(X)	(X)	(X)					
Betahat	(X)		(X)						
EasyReg	X								
EViews	(X)		(X)	(X)					
FP									
Gretl	(X)		(X)	(X)	(X)			(X)	
LIMDEP	X	X	X	X					
MicroFit	(X)			(X)					
Modeleasy$^+$	(X)		(X)	(X)					
MODLER	X	X	X	X					
NLOGIT	X	X	X	X					
PcGive		(X)	(X)	(X)				(X)	
RATS	(X)	(X)	(X)	(X)					
REG-X	(X)	(X)	(X)						(X)
SHAZAM	(X)		(X)	(X)		(X)			
SORITEC									
STAMP									
Stata	(X)*		(X)	(X)*	(X)	(X)	(X)		
TROLL									
TSP	X		(X)	(X)	(X)				
Wysea									
Econometric programming language applications									
GaussX	(X)		(X)	(X)	(X)	(X)	(X)		
TSM 4	X	X	X	(X)#	X				

() – indicates that the statistic can be produced at the option of the user.
Breusch-Godfrey – A "*" here indicates that the user can choose to perform this test with or without zeros per the footnote on page 269 of Greene's 5th edition (Greene, 2003) and as mentioned in the text of this volume. The "#" indicates that the counterpart M-test is also produced.
Jarque-Bera – A "*" here indicates that modifications are made to the classic form of this test.

argued that when zeros are used these values are effectively arbitrarily chosen and risk biasing the statistic, whereas the truncation of the test sample, although it does reduce the efficiency of the test (compared to using known values of the presumably unknown lagged values), also imposes no other qualification per se. The Breusch-Godfrey test statistic can be formulated in a Lagrange Multiplier form or as an F-test, the latter of which may have somewhat better small sample properties.

The Box-Pierce test (Box & Pierce, 1970), which is sometimes referred to as the Q test, is, as Greene (2000) points out, an alternative test that is asymptotically equivalent to the Breusch-Godfrey when the null hypothesis is true, $p = 0$, and **X** contains no lagged values. The Box-Pierce is a Chi-square test, with p degrees of freedom, such that:

$$\mathrm{Q} = \mathrm{T}\sum_{t=1}^{P} r_i^2$$

And

$$\mathrm{r_j} = \frac{\sum\limits_{t=i+1}^{T} u_t\, u_{t-i}}{\sum\limits_{t=1}^{T} u_t^2}$$

The Ljung-Box statistic (Ljung, 1979) in turn is defined by:

$$Q' = \mathrm{T}(\mathrm{T}+2)\sum_{i=1}^{P} \frac{r_i^2}{T-i}$$

It is obviously a particular variation on the Box-Pierce, with potentially improved small sample properties. However, Johnston and DiNardo, as well as MacKinnon (1992), maintain that the use of these statistics may be, at best, ad hoc and at worst inappropriate whenever the original regression is not a pure autoregressive scheme, but instead contains other regressor variables. Like the Durbin-Watson, the Box-Pierce and Ljung-Box tests are strictly valid only when the regressors are non-stochastic. However, it is also worth mentioning, as pointed out by Davidson (20, p. 164), that the Box-Pierce test, applied to the squared residuals, has been interpreted as a portmanteau test for ARCH (Bollerslev, 1988; McLeod & Li, 1983). Furthermore, it can easily be shown (Davidson, 2000) that not only are these several tests asymptotically equivalent in the case of exogenous regressors, but that in the case $p = 1$ they are essentially equivalent to the Durbin-Watson h statistic.

One of the operational questions these test raise is, how should the length of the lag be set? One possibility is for the program to do this automatically, for instance by setting it equal to the observation frequency of the data used in the original regression. Alternatively, the individual user might be permitted to set it.

In practice, the choice made varies from program to program, although sometimes the observation frequency will be used as an initial default, with the user then able to change this setting as an option.

The Jarque-Bera test (Bera & Jarque, 1981; Jarque & Bera, 1980), in contrast, is constructively quite different from these others inasmuch as it directly uses certain distribution properties to test normality, employing the residuals of the original regression. When implementing it, there are at least two possible options. The original approach, adopted by MicroFit (Pesaran & Pesaran, 1997), MODLER, and TSP, for example, is to construct measures of skewness and kurtosis directly from the original regression residuals and then to use these to form:

$$\chi^2_N(2) = \mathrm{T}(\mu_3^2/6\mu_2^3) + \mathrm{T}((\mu_4/\mu_2^2 - 3)^2/24) + \mathrm{T}(3\mu_1^2/2\mu_2 - \mu_3\mu_1/\mu_2^2)$$

where:

$$\mu_\mathrm{j} = \sum_{t=1}^{T} \frac{e_t^j}{T}$$

Notice in particular that whenever the original regression contains a constant term then $\mu_1 = 0$.

Alternatively, it is possible to normalize the residuals from the original regression to generate the set ε_t, $\mathrm{t} = 1, 2, \ldots, \mathrm{T}$:

$$\varepsilon_\mathrm{t} = \frac{\hat{\mathrm{u}}_\mathrm{t} - \bar{\mathrm{u}}}{\mathrm{s}}$$

where $\bar{\mathrm{u}}$ is the sample mean of the residuals and s is the sample standard error. The ε_t are then used to construct the measures of skewness and kurtosis and hence to form a variant of the Jarque-Bera test. A particular variation, implemented by Stata, corrects the Jarque-Bera for sample size by applying the skewness and kurtosis test of D'Agostino, Balanger, and D'Agostino Jr. (1990) and, in addition a modified version of this test (Royston, 1991). In contrast, B34S and RATS use adjusted third and fourth moments.

A further alternative is the test found in PcGive (Hendry & Doornik, 1999), which is due to Doornik & Hansen (2008). This test is described in the program's user guide as similar in spirit to Jarque-Bera but functionally differs inasmuch as it incorporates a small sample correction. Table 5.2 shows that, in addition to the Doornik-Hansen, other tests of disturbance properties that are occasionally implemented by developers, are the Bartlett's periodogram-based test (Bartlett, 1955), as well as the Harvey LM (Cummins, 2006; Harvey, 1990), Shapiro-Francia (Shapiro & Francia, 1972), and Shapiro-Wilk (Shapiro & Wilk, 1965).

Specification Tests: Functional Form, Nonlinearity, and Simultaneity

It is possible, with justification, to view the entire set of Gauss-Markov assumptions as defining the "specification," and if this interpretation is strictly enforced, (mis)specification tests consist of any and all tests of any of these assumptions, jointly and severally. However, it is also possible to argue that the term "specification" can apply more locally to the hypotheses embodied in the basic statement of the linear model. That is, given the linear hypothesis (A1):

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{u}$$

its misspecification can occur in a variety of ways: among other things, regressor variables may be improperly included or excluded; the unknown parameters, $\boldsymbol{\beta}$, may not be constant; linear or nonlinear relationships may exist among the elements of $\boldsymbol{\beta}$; the variables $X_{it} \in \mathbf{X}$ may not be stationary; or these variables $X_{it} \in \mathbf{X}$ and $\mathbf{u}$ may not be distributionally independent.

The Ramsey *REgression Specification Error Test* (RESET) is based upon the assumption of normally distributed disturbances; it is therefore only asymptotically valid under the assumptions A1–A4 above. However, given this caveat, it falls into the category of a rather general misspecification test, and its properties compared to other misspecification tests have been considered by Thursby (1989) and subsequently Godfrey & Orme (1994). As demonstrated by Ramsey (1969), if the estimated specification takes the form:

$$\mathbf{y} = \mathbf{Z}\boldsymbol{\beta}_{k-1} + \mathbf{v}$$

where

$$\begin{aligned}\mathbf{X} &= [\mathbf{Z}\ \ \mathbf{x}_1]\\ \boldsymbol{\beta}' &= [\boldsymbol{\beta}'_{k-1}\ \beta_1]\\ \mathbf{v} &= \mathbf{x}_1\beta_1 + \mathbf{u}\end{aligned}$$

Then, under the assumptions A2–A4, except that:

$$(\text{A2})^{*}\mathbf{u} = \text{N}(0, \sigma^2\mathbf{I})$$

is used to replace A2 (which configuration will hereafter be referred to as A1–A4*), and given that the distribution of the $\mathbf{u}$ are independent of the elements of $\mathbf{X}$, which obviously include $\mathbf{Z}$, it can be shown that the $\mathbf{v}$ are distributed normally but with a non-zero mean $\mathbf{M}'\xi$:

$$\mathbf{v} = \text{N}(\mathbf{M}'\xi, \sigma^2\mathbf{I})$$

where

$$\mathbf{M} = \mathbf{I} - \mathbf{X}(\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'$$

Ramsey RESET Test:	F(1, 16)=	0.39572,	Chi-Sqr(1) =	0.4827
Ramsey RESET Test:	F(2, 15)=	0.18999,	Chi-Sqr(2) =	0.4941
Ramsey RESET Test:	F(3, 14)=	0.27797,	Chi-Sqr(3) =	1.1243
Ramsey RESET Test:	F(4, 13)=	0.56154,	Chi-Sqr(4) =	2.9466
Ramsey RESET Test:	F(5, 12)=	0.55713,	Chi-Sqr(5) =	3.7680

Fig. 5.3 Functional form and other misspecification tests

and ξ is "a vector that summarizes the effect of the specification error on the mean of the disturbance in the misspecified model." A similar result can be shown for other misspecifications, including types of misspecification of the functional form and when elements of **X** and **y** are jointly determined.

In the general case, under the assumptions A1–A4*, and focusing on the *alternative* hypothesis, Ramsey argues that the relevant residuals will be distributed normally, but with a mean vector of the form $\mathbf{A}'\xi$. This vector involves unknown elements, but can be expressed as a linear function of the moments of the least squares estimator of the conditional mean of **y**. That is, consider the augmented regression:

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \hat{\mathbf{y}}^2\alpha + \varepsilon$$

where ε is a vector of disturbances, β is a vector of parameters, α is a scalar parameter and $\hat{\mathbf{y}}$ is a $\mathrm{T} \times 1$ vector of fitted values from the original regression. The RESET test for specification error is then whether $\alpha = 0$. A generalization is:

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}\alpha + \varepsilon$$

where

$$\mathbf{Z} = [\hat{\mathbf{y}}^2\hat{\mathbf{y}}^3\hat{\mathbf{y}}^4\hat{\mathbf{y}}^5\hat{\mathbf{y}}^6]$$

and $\boldsymbol{\alpha}$ is a vector of constant parameters, in which case the condition to be tested is obviously $\boldsymbol{\alpha} = 0$.

The implemented tests are shown in Fig. 5.3, using the Theil-Grunfeld data and based progressively upon increasing powers of the predicted dependent variable, starting with $\hat{\mathbf{y}}^2$ and ending with the complete set **Z**. Table 5.3 shows the degree to which the RESET Test appears in the existing econometric software packages.

Structural Stability Tests

The Chow test (Chow, 1960; Fisher, 1970) is commonly presented in the form of a predictive test. Its logic can be viewed as follows: consider a particular sample of data $[\mathbf{y}_1\mathbf{X}_1]$ consisting of observations on the time intervals t = 1,2,..,T and non-overlapping observations on the same variables for another set of time intervals, $[\mathbf{y}_2\mathbf{X}_2]$, which can usually be understood to refer to a later time period, $t = T + 1, \ldots, T + N$. One of the obvious questions to ask is whether the parameters of the

Table 5.3 General misspecification tests

	Ramsey RESET
Independent software packages	
AREMOS	
B34S	(X)
Betahat	(X)
EasyReg	
EViews	(X)
FP	
Gretl	(X)
LIMDEP	
MicroFit	(X)
Modeleasy+	(X)
MODLER	X
NLOGIT	
PcGive	(X)
RATS	(X)
REG-X	(X)
SHAZAM	(X)
SORITEC	
STAMP	
Stata	(X)
TROLL	
TSP	X*
Wysea	
Econometric programming language applications	
GaussX	X
TSM 4	(X)

() – indicates that the statistic can be produced at the option of the user.
* – indicates that the order 2 test is computed by default, with higher order tests computed by request.

specification are the same for both time periods. A possible way to approach this question is to consider the use of the first set of observations, $T > K+1$, to estimate the specification:

$$\mathbf{y}_1 = \mathbf{X}_1\boldsymbol{\beta} + \mathbf{u}_1$$

and then to generate predictions:

$$\hat{\mathbf{y}}_2 = \mathbf{X}_2\mathbf{b}$$

where:

$$\mathbf{b} = (\mathbf{X}_1'\mathbf{X}_1)^{-1}\mathbf{X}_1'\mathrm{y}_1$$

Notice that under the hypothesis that $\boldsymbol{\beta}$ is constant for the combined sample [$\mathbf{y}$ $\mathbf{X}$], where $\mathbf{y} = [\mathbf{y}_1\mathbf{y}_2]$ and similarly $\mathbf{X} = [\mathbf{X}_1\mathbf{X}_2]$, the prediction error takes the form:

$$\mathbf{e} = \mathbf{y}_2 - \hat{\mathbf{y}}_2 = \mathbf{u}_2 + \mathbf{X}_2(\mathbf{b} - \boldsymbol{\beta})$$

since, in this case: $\mathbf{y}_2 = \mathbf{X}_2\boldsymbol{\beta} + \mathbf{u}_2$.

It can therefore easily be shown that the variance covariance matrix of **e** can be stated as:

$$\mathrm{E}(\mathbf{ee}') = \sigma^2[\mathbf{I}_{\mathrm{N}} + \mathbf{X}_2(\mathbf{X}_1'\mathbf{X}_1)^{-1}\mathbf{X}_2']$$

Which then leads to the test statistic:

$$\mathrm{C} = \frac{e'[I_N + X_2(X_1'X_1)^{-1}X_2']e/N}{RSS_1/(T-K)}$$

where $\mathrm{RSS}_1 = \hat{\mathbf{u}}_1'\hat{\mathbf{u}}_1$. An aspect of the test that should be recognized is that it rather crucially depends upon assumption A2, and in particular that $\mathrm{E}(\mathbf{uu}') = \sigma^2\mathbf{I}$, where definitionally $\mathbf{u} = [\mathbf{u}_1\mathbf{u}_2]$.

An alternative, which is another Chow test (actually in precedence the first Chow test) and has been implemented more or less equally often by econometric software developers, is simply to divide the original sample into m subsets, where m is a small number, usually 2 or 3, and the number of observations in each sample is at least K, the number of parameters to be estimated using it. The issue of whether or not the parameter $\boldsymbol{\beta}$ is constant for all T can then be viewed as being a restricted specification, compared with the hypothesis that the values of one or more elements of $\boldsymbol{\beta}$ vary from sub-sample to sub-sample. For example, let m = 2, and consider the unrestricted case that:

$$\mathbf{y}_1 = \mathbf{X}_1\boldsymbol{\beta}_1 + \mathbf{u}_1$$

and

$$\mathbf{y}_2 = \mathbf{X}_2\boldsymbol{\beta}_2 + \mathbf{u}_2$$

where it is possible that $\boldsymbol{\beta}_1 \neq \boldsymbol{\beta}_2$. This can occur in any of three ways: one or more slope parameters can differ between the time periods, one or more constant terms can differ, or both types of parameters can differ between periods. However, what is fundamentally involved is simply a restrictions test. The restricted case of no change in parameter values between the two sub-sample periods is defined by:

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{u}$$

where, as before:

$$\begin{aligned}\mathbf{y} &= [\mathbf{y}_1\mathbf{y}_2]\\ \mathbf{X} &= [\mathbf{X}_1\mathbf{X}_2]\\ \mathbf{u} &= [\mathbf{u}_1\mathbf{u}_2]\end{aligned}$$

and $\boldsymbol{\beta}_1 = \boldsymbol{\beta}_2 = \boldsymbol{\beta}$ is asserted.

As demonstrated by Johnston (1984, pp. 217–219), it is straightforward to set up three alternative specifications, model 1, model 2, and model 3, such that model 1 is the restricted model, model 2 allows the restriction of constant intercept parameters to be relaxed, and model 3 allows the restriction of constant slope parameters to be

```
Based on three subperiods:
   Intercept Parameters: F( 2, 15)=    4.5196
   Slope Parameters:     F( 4, 11)=    3.4655
   All Parameters:       F( 6, 11)=    4.8073
   Break Point(s):    194301,  194901
```

Fig. 5.4 Structural stability tests

relaxed. The three test statistics, numeric values for which are displayed in Fig. 5.4, then take the form:

$$\mathrm{F} = \frac{RSS_1 - RSS_2}{RSS_2/(T-K-1)} \sim \mathrm{F}(1, \mathrm{T}-\mathrm{K}-1)$$

$$\mathrm{F} = \frac{(RSS_2 - RSS_3)/(K-1)}{RSS_3/(T-2K)} \sim \mathrm{F}(\mathrm{K}-1, \mathrm{T}-2\mathrm{K})$$

$$\mathrm{F} = \frac{(RSS_1 - RSS_3)/K}{RSS_3/(T-2K)} \sim \mathrm{F}(\mathrm{K}, \mathrm{T}-2\mathrm{K})$$

where RSS_1 refers to the residual sum of squares from the restricted regression, $\boldsymbol{\beta}_1 = \boldsymbol{\beta}_2$, RSS_2 refers to the residual sum of squares from the regression that restricts the slope parameters but not the intercepts, and RSS_3 the residual sum of squares when the intercept parameters are restricted to be equal but not the slope parameters. The generalization to $m > 2$ is only slightly more complicated. Obviously, the sub-samples must each provide at least K+1 observations on each variable, which places an upside restriction on m. Otherwise, it is not clear a priori how the breakpoints should be chosen in order to form the sub-samples. In the absence of any a priori information, one option is to choose the sub-samples according to the rule of thumb T/m, with any remainder assigned to specific time periods.

Another possibility, which appears to be unique to EViews, is to perform a test, called a Factor Breakpoint Test, that is similar to the Chow breakpoint test, but instead allows the user to create the sub samples on the basis of the characteristics of a categorical variable, such as male/female. This test is conceived to be most obviously applicable for regressions using cross-section data, but as implemented it can also be used when the data are time series. In the latter case, it is provided on a "user beware" basis, inasmuch as in the presence of lagged variables there is the obvious problem of how to create meaningful sub samples.

A different approach, due to Hansen (1992), is based on a cumulative sum of the least squares residuals. The test makes use of two circumstances in particular. First, the property of the OLS estimator that:

$$\mathbf{X}'\hat{\mathbf{u}} = 0$$

and next that:

$$\sum_{t=1}^{T} (\hat{\mathrm{u}}_{\mathrm{t}}^2 - \hat{\mathbf{u}}'\hat{\mathbf{u}}/\mathrm{T}) = 0$$

Table 5.4 Structural stability tests

	Breakpoint Chow tests	Factor breakpoint test	Chow prediction test	Hansen's test	Recursive residuals	CUSUM	Anderson Darling	Cramer-von Mises	Lilliefors	Watson
Independent software packages										
AREMOS	(X)									
AutoBox	(X)									
B34S	(X)				(X)	(X)				
Betahat					(X)	(X)				
EasyReg										
EViews	(X)	(X)	(X)		(X)	(X)	(X)	(X)	(X)	(X)
FP										
gretl	(X)					(X)				
LIMDEP										
MicroFit	(X)		(X)							
Modeleasy+	(X)		(X)		(X)	(X)				
MODLER	X									
NLOGIT										
PcGive	(X)		(X)	(X)		(X)				
RATS			(X)	(X)	(X)	(X)				
REG-X			(X)		(X)	(X)				
SHAZAM	(X)		(X)	(X)	(X)	(X)				
SORITEC										
STAMP			(X)							
STATA			(X)							
TROLL										
TSP	(X)*				(X)	(X)				
Wysea										
Econometric programming language applications										
GaussX	(X)		(X)	(X)	(X)	(X)				
TSM 4	(X)		(X)	(X)						

() – indicates that the statistic can be produced at the option of the user.
* – indicates that the program implements the hetroscedastic robust Chow test (MAC2) proposed by Thursby (1992).

Therefore, let:

$$f_{it} = x_{it}\hat{u}_t \qquad i = 1, ..K$$
$$= \hat{u}_t^2 - \hat{\mathbf{u}}'\hat{\mathbf{u}}/T \; i = K + 1$$

and

$$s_{it} = \sum_{j=1}^{t} f_{ij}$$

Clearly

$$s_{iT} = \sum_{t=1}^{T} f_{it} = 0$$

or, following Green (33, pp. 294–295), let $\mathbf{f}_t$ be the ith observation in the above pair of sums and define the vector and matrices:

$$\mathbf{s}_t = \sum_{j=1}^{t} \mathbf{f}_j$$
$$F = \frac{1}{T}\sum_{t=1}^{T} \mathbf{f}_t\mathbf{f}_t'$$
$$\mathbf{S} = \frac{1}{T}\sum_{t=1}^{T} \mathbf{s}_t\mathbf{s}_t'$$

The Hansen statistic can then be expressed as:

$$H = \mathrm{tr}(\mathbf{F}^{-1}\mathbf{S})$$

Somewhat heuristic in its application, the Hansen H will be “small” under the null hypothesis and “large” otherwise. However, even apart from the fact of a non-standard distribution, a potential defect of the Hansen statistic is contained in the Johnston and DiNardo comment that, “The technical derivation of the test is beyond the level of [their textbook],” raising the issue that the user of a software package may thereby be presented with a “black box” statistic, rather than an easily comprehended concept. Certain statistics may be sufficiently useful and have properties that advocate for their use, notwithstanding complexity. Under these circumstances, each user arguably should feel the need to comprehend the various aspects of their use and textbooks should describe them. Is this one of these statistics?

A somewhat less commonly implemented test of structural stability is that originally formulated by Brown, Durbin, & Evans (1975). This approach, based upon the construction of recursive residuals, can be seen as linked to the Hansen test by the question of an estimated equation’s ability to provide sufficiently satisfactory predictions outside the sample period. Consider the values:

$$v_t = \frac{u_t}{\sqrt{1 + x_t'(X_{t-1}'X_{t-1})^{-1}x_t}}$$

such that the u_t are the residuals from a regression for $t = K + 1, \ldots, T$, and where X_{t-1} are the prior regressor observations and x_t those for the t^{th} period. Under the set of assumptions A1–A4*, including normally distributed disturbances, not only are the v_t pairwise uncorrelated, but in addition, the elements of the vector of these, **v**, have the property:

$$v_t = N(0, \sigma^2 \mathbf{I}_{T-K})$$

The cumulative sum (CUSUM) is given by:

$$CS_t = \sum_{i=K+1}^{t} v_i / s$$

where:

$$s^2 = \hat{\mathbf{u}}'\hat{\mathbf{u}}/(T - K)$$

Under the null hypothesis that the original regression parameters are constant $E(CS_t) = 0$, implying that when CS_t is plotted against t, it should tend to cleave to the zero mean value axis, whereas when the parameters are non-constant, it can be expected to diverge from this axis. Confidence bounds are specified in order to judge the significance of the degree of departure from the axis of the CS_t. In turn, the Cumulative Sum Squared (CUSUMSQ) is based on the statistic:

$$CSQ_t = \frac{\sum_{i=K+1}^{t} v_i^2}{\sum_{i=K=1}^{T} v_i^2}$$

where, under the null hypothesis:

$$E(CSQ) = \frac{t - K}{T - K}$$

and which obviously can be used to plot a vector of values from zero at t = K to the value one at t = T. Divergence boundaries for different sample sizes and significance levels can be found in Brown, Durbin, and Evans. It can be shown that:

$$(T - K)CSQ_t \sim \chi^2(t - K)$$

It is clear that in principle, it may be possible to infer parameter changes from the recursive residuals, but it is also true that the CUMSUM and CUMSUMSQ tests are not robust in the presence of heteroscedasticity, outliers in the sample residuals, or breaks in the regressor variable values – essentially any instabilities, including changes in the $\boldsymbol{\beta}$ parameter values (Andrews, 1993; Greene, 2003; Johnston & DiNardo, 1997). For a more detailed examination of the several issues involved, see, for instance, the numerical example and discussion in chapter sect. 4.4

of Johnston and DiNardo (pp. 121–126), which discusses the Chow, Hansen, and CUMSUM and CUMSUMSQ individually and severally. For an analysis that also examines the relationship between recursive residuals and Theil's BLUS residuals each as members of the Least Unbiased Scalar (LUS) class of variates, see Chap. 9 of Stokes (1997). During the past 10 years or so, there has been further work in this area, in the form of contributions by Andrews (1993), Andrews & Ploberger (1994), Hjort & Koning (2002), Kuan, & Hornik (1995), Zeileis & Hornik (2007), and most recently Zeileis (2006), that is not yet fully reflected either in the textbooks or in econometric software.

Omitted Variables, Linear Restrictions and Related Tests

To a degree, there is a historically sanctioned, almost natural way to classify the various possible misspecification tests, either because of their original purpose individually or because of an individual association with one of the assumptions (A1–A4) considered earlier. Because of this history, this classification scheme has generally been adopted here. However, as mentioned before, an attempt has also been made in the conduct of the present survey to do this in such a way so as not to precondition the results –in order to permit them to be directly determined by the properties of the existing econometric software packages. These two principles can conflict, for although certain tests are unambiguous in classification, some are much less so, and the way in which econometricians categorize these at different times and in different contexts is itself interesting. For example, it was mentioned earlier that PcGive formerly presented a variant of White's test as a test for misspecification of functional form. To at least some degree, this original interpretation appears to have been based upon an earlier conjecture by Pagan & Hall (1983) that the conclusions obtained from White's test might be expected to be similar to those obtained from the use of Ramsey's RESET test (Godfrey & Orme, 1994).

Actually, in the final analysis, the resolution of this particular classification problem is not all that important in itself. In contrast, the fact that this type of interpretation problem can occur from time to time is illustrative, for it is true that omitted variables is one of the misspecifications that originally motivated the RESET test, yet this test has been considered earlier, notwithstanding that the title of the present section specifically refers to omitted variables, possibly implying that that test should instead be considered here. The choice made not to do this, in part, simply reflects the current practices of econometric software developers. Notice that the particular problem that the omission of *relevant* variables poses – for it is only such variables that matter in this context, at least in principle – is that this pathology affects both the properties of the included parameter estimates and the ability to make reliable inferences. Indeed, it is recognition of this by Godfrey & Orme (1994) that motivates their argument to the effect that White's test is "ineffective" in the presence of omitted variables.

There are at least two possible approaches to the problem of omitted variables. One is to consider it from an agnostic perspective: that of not knowing *if* there are relevant omitted variables; in this instance, Ramsey's RESET test is a prime candidate for consideration. Alternatively, it is possible to test specifically to see if certain variables that have not been included are in fact relevant. In the second instance, particular circumstances will condition the nature of the test. These can include not only the omission (or not) of one or more variables, but also the way in which these variables might be interpreted. For example, as Davidson points out (2000, pp. 189–190), it is possible to consider as a type of omitted variables problem the question of the simultaneity or exogeneity of one or more of the regressors. To see this, consider the regression specification:

$$\mathbf{y} = \mathbf{X\beta} + \mathbf{u}$$

and the alternative such that:

$$\begin{aligned} \mathbf{y} &= \mathbf{X\beta} + \mathbf{v} \\ \mathbf{v} &= \mathbf{V\gamma} + \mathrm{e} \end{aligned}$$

where $\mathbf{V}$ is a matrix of observations on variables that can in fact be interpreted as the disturbances of a set of OLS regressions that happen to be the first-stage regressions of Two-Stage Least Squares on a set of instruments. Of course, these disturbances are unobserved, but it is possible to perform the regression:

$$\mathbf{y} = \mathbf{X\beta} + \hat{\mathbf{V}}\boldsymbol{\gamma} + \mathbf{e}$$

where the observations:

$$\hat{\mathbf{V}} = \mathbf{X}_2 - \mathbf{X}_2$$

are directly obtained from the regression of variables originally included in $\mathbf{X}$ that are thought possibly to be jointly determined with the regressand, $\mathbf{y}$. If in fact these variables are not jointly determined with $\mathbf{y}$ the implication is that $\gamma = 0$. Under the null hypothesis, the variable-addition test statistic is:

$$\mathrm{T}(\hat{\mathbf{u}}'\hat{\mathbf{u}} - \hat{\mathbf{e}}'\hat{\mathbf{e}})/\hat{\mathbf{e}}'\hat{\mathbf{e}} \sim \chi^2(\mathrm{K}_1)$$

Where K_1 is the number of columns of $\mathbf{V}$. This statistic is evidently formulated using the sums of squared residuals from each of the regression specifications shown. Obviously, as presented, this test is not of immediate utility in the context in which instruments have not been previously chosen, but it fits within the context of the Durbin-Wu-Hausman exogeneity tests (Durbin, 1954; Hausman, 1978; Wu, 1973) and shows another aspect of these. In particular, Davidson demonstrates that it is a generalization of the Hausman test (Hausman, 1978). Note also that, like the Hausman test, the Davidson variant just described is specifically a test of the consistency of OLS in the presence of possible jointly determined variables. At once it is also an omitted variables test such that the relevant (omitted) variables happen to be systemically related to the dependent variable, although it is also clear on further reflection that it is the precise definition of the variable $\mathbf{V}$ that is critical to the interpretation made.

There are two reasons to have adopted this somewhat crab-like approach to the question of omitted variables. The first is to demonstrate once again that the contextual nature of a test is very often a matter of the particular variables that are included in its construction. In the example just considered, simply identifying **V** as a matrix of possibly relevant omitted variables conceptually changes the test from an exogeneity test to an omitted variables test. The second reason is the issue of software design: a computational attraction is that, to deal with such cases, a particular software package can be designed so as to allow regressions of different types to be performed based upon some type of switch in the individual regression command. In the case just considered, the instruments may in practice have already been chosen for later use in Two- or Three-Stage Least Squares regressions, and are therefore available when an OLS regression is performed, making it quite easy to show the above test statistic in the context of each of the OLS regression displays. The necessary sums of squares from the "first stage" regressions can simply be saved and plugged in, as needed, to compute the test statistic. Instead, the standard set up for the Hausman LM test is often to use the auxiliary regression:

$$\mathbf{v} = \mathbf{X}\boldsymbol{\beta} + \hat{\mathbf{V}}\boldsymbol{\gamma} + \boldsymbol{\varepsilon}$$

from which the test statistic is computed as TR^2, having previously saved the residuals from the first-stage regressions.

Of course, in the more general omitted variables case, if one wishes to consider if a particular, additional variable should in fact be included in the regression, then it is always possible to formulate this test case as an auxiliary regression, performing first the regression exclusive of the possibly relevant variable, and then using the residuals of this regression, **v**, to form this auxiliary regression and compute TR^2. Alternatively, it is possible to follow the Davidson procedure described above, using the sum of the squared residuals from each of the two regressions. Notice also that in each of the cases discussed, it is possible to reformulate the test as a F-test, rather than a χ^2.

Turning attention now to tests of linear restrictions, a standard test is obtained by asserting the null hypothesis to be that there exist such restrictions on the elements of the OLS parameter vector, $\boldsymbol{\beta}$, such that:

$$\mathrm{R}\boldsymbol{\beta} = \mathbf{c}$$

where **R** is a r x K matrix of constants, with rank r, and **c** is a r x 1 vector of constants. An immediately available standard result is:

$$\frac{(R\beta - c)\,[R(X'X)^{-1}R']^{-1}(R\beta - c)}{rs^2} \sim \mathrm{F(r, n-k)}$$

where:

$$\mathrm{s}^2 = \hat{\mathbf{u}}'\hat{\mathbf{u}}/(\mathrm{T}-\mathrm{K})$$

or, equivalently:

$$(\hat{\mathbf{u}}_c'\hat{\mathbf{u}}_c - \hat{\mathbf{u}}_{nc}'\hat{\mathbf{u}}_{nc})/\hat{\mathbf{u}}_{nc}'\hat{\mathbf{u}}_{nc}\frac{T-K}{r} \sim \mathrm{F(r, n-k)}$$

where $\hat{\mathbf{u}}_c'\hat{\mathbf{u}}_c$ is the residual sum of squares from the restricted regression and $\hat{\mathbf{u}}_{nc}'\hat{\mathbf{u}}_{nc}$ from the unrestricted.

Testing non-linear restrictions, in contrast, is rather less simply described (Bhargava, 1987; Critchley, Marriott, & Salmon, 1996; Gregory & Veall, 1985, 1987). In the first place, non-linear forms are obviously potentially quite various, making it therefore difficult to even envisage a general test (Harvey, 1990; Spitzer, 1976). However, when particular types of non-linear restrictions are considered, such as $\boldsymbol{\beta}_4 = \boldsymbol{\beta}_2\boldsymbol{\beta}_3$, for example, there is, to begin, the problem that whereas the unrestricted model may be linear, the restricted model will not be. A problem that also occurs is that the restrictions themselves may be variously formulated, for example: $\boldsymbol{\beta}_4 - \boldsymbol{\beta}_2\boldsymbol{\beta}_3 = 0$ or $\boldsymbol{\beta}_3 = \boldsymbol{\beta}_4/\boldsymbol{\beta}_2 = 0$, if $\boldsymbol{\beta}_2 \neq 0$, which can obviously create inference problems. Because of such problems, seemingly, it is uncommon for econometric software packages to include tests of non-linear restrictions, as is indicated by Table 5.5, especially those that are generally applicable, as opposed to one that might be used in a specific, given context.

Table 5.5 Omitted variables, linear restrictions and other related tests

	Omitted variables	Linear restrictions	Other coefficient restriction	Wald zero restrictions	ComFac
Independent software packages					
AREMOS					
B34S	(X)	(X)		(X)	
Betahat		(X)			
EasyReg		(X)		(X)	
EViews	(X)	(X)	(X)	(X)	
FP					
Gretl	(X)	(X)	(X)	(X)	
LIMDEP					
MicroFit	(X)	(X)			
Modeleasy+	(X)	(X)			(X)
MODLER	(X)	(X)			
NLOGIT					
PcGive	(X)	(X)			(X)
RATS	(X)	(X)		(X)	
REG-X					
SHAZAM		(X)			
SORITEC					
STATA	(X)	(X)	(X)	(X)	
TROLL					
TSP			(X)		(X)
Wysea	(X)	(X)	(X)	(X)	
Econometric programming language applications					
GaussX		(X)		(X)	
TSM 4	(X*)	(X)	(X)	(X)	(X*)

() – indicates that the statistic can be produced at the option of the user.
Omitted Variables – the "*" indicates the use of an F test.
ComFac – the "*" indicates the computation as a LM test in a fitted AR model.

Chapter 6
Cointegration and Alternative Specifications

The particular topics considered in this chapter can be viewed as being at least in part determined by this study's stated focus on misspecification tests in the context of Ordinary Least Squares. However, the chapter's subject matter can also be seen to reflect the way in which econometric theory has developed during the past 30 years. In important respects, there has been a broadening in the range of econometric applications, epitomized by such things as the development of microeconometrics so called, the greater use of survey and panel data in economic research – often in the form of microdata – and other such expansions in the application of econometric techniques, which have not always been limited solely to economics and economists. But notwithstanding this "rippling out," involving a coincident growing interest in such things as "non-standard" dependent variables, cross-section techniques, and "spatial econometrics" – which has certainly affected the development of econometric software and would be evident from any more general survey that specifically attempted to probe these and related aspects – the dominant themes in the development of econometric theory, or at least the most obvious areas of that development, have continued to be associated with the properties of time series, what can be called specification search, and the characterization of the disturbance term and its possible distribution, jointly and severally.

The tests and statistics considered to this point characteristically (if implicitly) assume knowledge of the data generating process (DGP) or, in other words, if much more casually, the set of candidate variables in the specification. Considering the test statistics presented in Chaps. 4 and 5, the questions that have been addressed are those of functional form, properties of the disturbances, and whether or not there appear to be relevant variables omitted. Even in the case of tests for omitted variables, the question has not yet been raised whether such variables, if they exist, represent alternatives to those already included? However, considering these matters more generally and expansively, it is obviously possible to ask instead, or in addition, whether a fundamental specification selection mistake might have been made, or at least whether or not there is an alternative specification that might be more appropriate. But once such questions are posed, another also arises, namely, why were such possible better alternatives not considered at the beginning? Obviously, the deep questions of functional form and variable selection are logically prior questions, involving not the "fine tuning" of an hypothesis about economic behavior,

C.G. Renfro, *The Practice of Econometric Theory*, Advanced Studies in Theoretical and Applied Econometrics 44, DOI 10.1007/978-3-540-75571-5_6,

but instead – at a more fundamental level – what the gross form of that hypothesis should be? The logic of beginning to consider these types of tests only at this relatively late stage is simply a matter of historical precedence: The questions that give rise to these tests have been asked, in the particular way they are now, only during the past 25 or so years.

The early days of this story can be dated to the early 1980s, or perhaps the later 1970s, although the particular dating does depend upon the location of the person doing it. In early 1984, Christopher Gilbert began to lecture to undergraduates at the University of Oxford on the topic "Professor Hendry's Econometric Methodology," the lecture later published in the *Oxford Bulletin of Economics and Statistics* (Gilbert, 1986). In this context, Gilbert characterized alternative econometric methodologies as British and American and classified the first as represented most noticeably by the work of David Hendry. The second he identified by the ostensibly quite arbitrary name AER, under the rubric "the AER Procedure;" AER was said to be short for the "Average Economic Regression." The strawman set up by Gilbert under this label was the linear-in-parameters single equation specification:

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \mathbf{u}$$

interpreted to be derived from theory and "*known* to be correct" (p. 284). The econometric problem was defined by this procedure to be the estimation of the (constant) parameters contained in the vector β, involving perhaps a choice of estimator, but most fundamentally an exploration of the potential *pathologies*, which might include "serial correlation, multicollinearity, heteroscedasticity, simultaneity, and so forth" (p. 284), essentially involving the types of tests (but not the variety) considered in Chap. 5. Typically, says Gilbert, "the problems [as conceived by this paradigm] manifest themselves in applied work in terms of *low* Durbin–Watson statistics, *wrong* signs, *insignificant* coefficients and so forth. The term "wrong" is telling – we know what the right sign is; the estimates give us the wrong sign; and the econometrician's response to these pathological manifestations is to respecify his equation in some way – to add or subtract variables, change the definition of variables and so forth – until eventually he gets an equation which has all correct signs, statistically significant coefficients, a Durbin–Watson statistic of around 2, a relatively high R^2, and so forth" (p. 284).

The Hendry methodology – what has previously been referred to as the "LSE methodology" and can also be called the General-to-Specific methodology (GSM) – Gilbert portrays as grounded in the concept of the *data generating process* (DGP), essentially "the joint probability of all the sample data (i.e. on both endogenous and exogenous variables)" (p. 285), which clearly has some (unstated) relationship to the (marginal) distributions posited as individual specifications by Cowles Commission analysts, although perhaps more grounded in the sample itself, rather than an economic phenomena that could be either observed or not observed within some error tolerance. Gilbert indicates that "econometric modeling consists of judicious simplification of this DGP," through a process that he characterizes as involving marginalization, conditioning, simplification and estimation. "Provided that the marginalization and conditioning are valid, this allows the econometrician to replace the very general representation [of the joint probability of the sample data] by the much more specific" (p. 286) conditional distribution. However, Gilbert asserts "We do not know in advance whether the simplifications we have made [in this transition] are valid. Furthermore, the concept of validity is not straightforward. Economies are complicated organizations, and this must be reflected in the complexity of the general representation [of the original] DGP... This implies that we know for certain that the simplified representation of the [conditional distribution] cannot be strictly valid. The question therefore becomes one of *adequacy* rather than *validity*. Hendry proposes that we look for a *tentatively adequate conditional data characterization*, or what might be called a model which is *congruent* with all the evidence" (p. 286).

Professor Hendry is still very much alive and quite capable of speaking for himself (Davidson, Hendry, Srba, & Yeo, 1978; Hendry, 1986, 1993, 2000; Hendry & Doornik, 1999; Mizon, 1984), so that the reason to have provided this particular (actually partial) summary of the Hendry-GSM-LSE methodology is neither to establish another strawman, nor to try to counterpose more correctly two alternative methodologies. Instead, it is merely to bring out explicitly that the discussion of the misspecification tests in Chaps. 4 and 5 does not adequately consider the way in which these tests are applied in practice. In fact, one of the reasons to have previously considered in Chap. 2 some of the existing evidence on the use of software is to make clear, other than in the form of hearsay, just how little is generally known about the application of econometric software by economists. It is possible to present the views of proponents of particular research methodologies and it is possible to determine the facilities that are available to economists (even historically), but it is rather more difficult to determine the effect on actual applied economic research. In addition, the just contrasted AER and GSM methodologies are of course but two of those that are seemingly being applied. It is possible to identify a Bayesian school among econometricians, proponents of which include Leamer and Zellner (Berry, Chaloner, Geweke, & Zellner, 1996; Jeffreys & Zellner, 1989; Leamer, 1978, 1994; Zellner, 1997). The use of VAR and other time series techniques, which can be seen as an extension of the original Box–Jenkins approach, represents another methodological gambit (Barton, 1941; Foschi & Kontoghiorghes, 2003; Rudebusch, 1998; Sargent & Sims, 1977; Sims, 1980), although it is not clear which pawn has been sacrificed. The Real Business Cycle approach and the calibrated general equilibrium model represent two further ways to characterize economic processes (de Mello, 1988; Kydland & Prescott, 1991, 1996; Lofgren, Harris, Robinson, Thomas, & El-Said, 2002; McKibben, 1998; Robinson & Roland-Holst, 1988; Rutherford, 1999). In addition, although declared dead somewhat exaggeratedly (Diebold, 1998), macroeconometric modeling still continues to be practiced, even if seldom defended in the modern economics literature. The amount of synthesis that has taken place is questionable, but the diversity is evident.

Unit Root Tests

A central question that remains is the meaningfulness of econometric representations, which is often considered in terms the distinction between correlation and causality. Of course, the idea of a "spurious regression" is, in itself, rather long-standing. Karl Pearson, beginning in the nineteenth century, and George Yule, in the 1920s, both considered the issue of correlation without causation (Aldrich, 1995; Yule, 1926). Furthermore, as an idea, it was not absent from the thoughts of econometricians in the 1930s through the 1970s, even before the seminal contribution by Granger and Newbold (1974). For much of the time, it was simply subsumed, known to be there, but seldom overtly considered. Today, in contrast, this idea is closely associated with cointegration.

The phenomenon of cointegration can be viewed as having its economic roots in the idea of the "great ratios" (Klein & Kosobud, 1961), the tendency for certain macroeconomic variables to be observed to have characteristic or normal values when formulated as ratios, especially those that might be considered to have the attributes of propensities, in the original spirit of Keynes' (1936) marginal propensity to consume and similar constructs. Granger (1991) has elaborated on this aspect, noting that (p. 65):

> At the least sophisticated level of economic theory lies the belief that certain pairs of economic variables should not diverge from each other by too great an extent, at least in the long-run. Thus, such variables may drift apart in the short-run or according to seasonal factors, but if they continue to be too far apart in the long-run, then economic forces, such as a market mechanism or government intervention, will begin to bring them together again. . .However, in each case the correctness of the beliefs about long-term relatedness is an empirical question. The idea underlying cointegration allows specification of models that capture part of such beliefs, at least for a particular type of variable that is frequently found in macroeconomics. Since a concept such as the long-run is a dynamic one, the natural area for these ideas is that of time-series theory and analysis.

However, another path that leads from this generalized economic logic to cointegration is via the so-called Error-Correction Mechanism (ECM) that has its econometric origins in the work of Phillips (1957) and Sargan (1964) and an underlying logic very similar to that expressed by Granger, but its actual motivation can be viewed as permitting the marriage of the short term and the long term.

In addition, there is yet another thread. As indicated earlier, it was recognized by statisticians such as Pearson as early as the nineteen century that time series economic variables can coincidentally and adventitiously exhibit common secular behavior, leading to "nonsense" or "spurious" regressions as false characterizations of relationships (Aldrich, 1995; Yule, 1926). During the first days of the development of econometrics, econometricians were aware of this problem and realized that macroeconomic variables might be particularly affected. As they began to develop macroeconometric models in the 1950s and 1960s, they quickly came to appreciate that model specifications stated in terms of differences, rates of change, and similar transformations would not involve the same degree of "nonsense" risk as do level formulations – and this result has been formalized more recently by

Stigler and Sherwin (1985; Johnston and DiNardo, 1997). However, model builders also soon discovered that this approach tends to have the unfortunate effect of modeling short-term behavior to the exclusion of the long-term, among other things leading to problems when solving these models dynamically. It is apparent that, in the case of linear and certain other specifications, the parameters can be interpreted as first derivatives, whereas they take on at least some of the attributes of second derivatives once the relationships are expressed in terms of rates-of-change or differences. This distinction both has particular computational consequences and involves significantly different degrees of representational complexity. Insight into the nature of this complexity can be gained by recognizing the degree to which economic theory commonly ignores dynamic effects; for example, the theory of the perfectly competitive firm is directionally unambiguous in its prediction of the behavioral effects of super-normal profits, but is entirely silent concerning the speed with which any changes might occur as a function of the particular amount of those profits.

The inference to be drawn is not that the approach associated with the ECM necessarily provides a complete answer, but instead simply that it provides a way to model long-term constraints and tendencies in order to explain short term disequilibrium behavior. A specific formalization of this approach began with the work of Davidson et al. (1978; Hendry, 1993), who modeled short term processes as involving proportional feedback corrections towards equilibrium, the happy result then being a co-modeling of short-term and long-term behavior. Granger and Engle subsequently have demonstrated the intimate relationship between cointegrated series and the ECM representation (Engle & Granger, 1991; Granger, 1981). However, as Pesaran has recently observed (Pesaran, 1997, p. 178), the disproportionate focus on the purely statistical properties of economic time series during the past more than 20 years has rather obscured for many economists the associated economic content and the potentially related theoretical insights that might come from "a more satisfactory integration of the cointegration analysis with the various structural economic modeling approaches that are extant in the literature."

As Table 6.1 demonstrates, the existing econometric software packages collectively implement a reasonably large number of unit root test statistics, and in addition exhibit a fairly high degree of commonality in the selections made. However, this table also illustrates by its number of columns the variety found in existing econometric software packages. Furthermore, it is additionally evident from this table that certain of these are more commonly implemented than others. At the same time, the table hides, in plain sight, that when viewing it there is a distinction that needs to be made between the unit root and the more general, specification tests. This table has not been organized so as to distinguish sharply between these types of tests, but instead it focuses attention on which are most commonly implemented. Nevertheless, the distribution of the selections displayed carries with it implications about the coverage the surveyed software packages provide. Notice also that all these tests are offered as optional tests, rather than as automatic defaults.

The Unit Root tests include the Dickey–Fuller, the Augmented Dickey–Fuller, the Phillips–Perron, and the Elliot–Rothenberg–Stock tests. Each of these can be

Table 6.1 Unit root tests

	Dickey Fuller	Augmented Dickey Fuller	Phillips Perron	Ng Perron	Kwiatkowski Phillips Schmidt Shin (KPSS)	Elliot Rothenberg Stock	Bierens	Bretung's	Weighted Symmetric
Independent software packages									
AREMOS									
AutoBox									
B34S	(X)	(X)	(X)			(X)			
Betahat	(X)	(X)	(X)						
EasyReg	(X)	(X)	(X)		(X)		(X)	(X)	
EViews	(X)	(X)	(X)	(X)	(X)	(X)			
FP									
gretl	(X)	(X)			(X)	(X)			
LIMDEP		(X)	(X)						
MicroFit	(X)	(X)	(X)						
Modeleasy+									
MODLER	(X)	(X)	(X)		(X)	(X)			
NLOGIT		(X)	(X)						
PcGive	(X)	(X)							
RATS	(X)	(X)	(X)		(X)	(X)			
REG-X	(X)	(X)							
SHAZAM	(X)	(X)	(X)						
SORITEC									
STATA		(X)	(X)			(X)			
TROLL									
TSP	(X)	(X)	(X)						(X)
Wysea		(X)							
Econometric programming language applications									
GaussX	(X)	(X)			(X)				
TSM 4	(X)	(X)	(X)		(X)				

() – implies that the statistic can be produced at the option of the user used to probe the properties of a univariate process, but they can also be viewed as progressively involving a more generalized characterization of the disturbance process. To fix ideas initially, consider first a time series variable x_t, and the stochastic process:

$$x_t = \lambda x_{t-1} + u_t$$

where u_t is IID(0, σ^2) and λ is a scalar constant. The specific properties of this process obviously depend on the value taken by λ; in particular whether $|\lambda|$ is less, greater, or equal to 1. In the case of equality, a unit root is implied, $\Delta x_t = u_t$, in which event the distribution of x_t is that of a random walk.

Consider next the reparameterization:

$$\Delta x_t = (\lambda - 1)x_{t-1} + u_t$$

which, with a little imagination, can be seen to be a first order autoregressive regression specification. Performing this regression in a simple, straightforward manner will produce both an OLS estimate of the autoregressive parameter and what initially might be regarded as its associated t-statistic, t_λ, which – with certain reservations – can be used to test for the condition $\lambda - 1 = 0$. This specification is, however, obviously rather restrictive. In practice, the Dickey–Fuller tests (Dickey & Fuller, 1979; Fuller, 1976) might be based upon any one of three regressions:

$$\Delta x_t = (\lambda - 1)x_{t-1} + u_t$$

or

$$\Delta x_t = \alpha + (\lambda - 1)x_{t-1} + u_t$$

or

$$\Delta x_t = \alpha + \beta t + (\lambda - 1)x_{t-1} + u_t$$

where $t = 2, 3, \ldots, T$, but these tests will each involve the use of the particular statistics, $\lambda - 1$ and t_λ – or rather τ as it is commonly termed, since it is not distributed as Student's t (Davidson & MacKinnon, 1993; Dickey, Bell, & Fuller, 1986). To distinguish between these specifications, the value of τ can be, respectively, denoted as $\tau_{\mathbf{nc}}$, $\tau_{\mathbf{c}}$, and $\tau_{\mathbf{ct}}$. These cases evidently can be viewed as increasingly more general and, in the case of the third, more specifically as providing a means to test between the alternatives:

$$x_t = \delta_0 + \delta_1 t + u_t$$

and

$$x_t = \delta_1 + x_{t-1} + u_t$$

With a little effort, these two equations can be shown (Davidson & MacKinnon, 1993, pp. 700–702) to be special cases closely related to the third equation above, where λ originates as the autoregressive parameter of a disturbance error process $v_t = \lambda v_{t-1} + u_t$. Under this interpretation, because α and β are actually terms that definitionally include λ, values of λ impose restrictions on the values of the other parameters; in particular $\lambda = 1$ implies $\beta = 0$ (p. 701).

At the same time, it is important to recognize that there are two possible test statistics. The first is computationally the same as the ordinary t-statistic, but as indicated earlier it is called τ in recognition that its distribution is *not* Student's t. The second type of Dickey–Fuller test is based upon the statistic:

$$\mathbf{z} = T(\hat{\lambda} - 1)$$

where the regression equation to which a particular statistic pertains can be determined by the particular way that $\mathbf{z}$ is denoted – as $\mathbf{z}_{nc}$, $\mathbf{z}_{c}$ or $\mathbf{z}_{ct}$, just as τ can be denoted as $\tau_{\mathbf{nc}}$, $\tau_{\mathbf{c}}$, and $\tau_{\mathbf{ct}}$. Clearly, the statement that a particular software package performs a Dickey–Fuller test is not completely informative.

In fact, not only is there an issue of the particular statistic, but there is as well a question of the particular specification of the test regression: for example, RATS, B34S, and LIMDEP use level values of the dependent variable – that is x_t – whereas other packages, such as MODLER and PcGive, use Δx_t. However, the principal distinction involved here is neither the values taken by the parameter estimates nor those by the test statistics, which are the same in either case. Instead, it is the interpretation of such ancillary statistics as the mean of the dependent variable. If, in particular, the dependent variable is stated in difference form then the sample value of its mean will provide some indication (albeit heuristic) whether the variable obeys a random walk – or else involves either trend stationarity or a drift. Obviously, $E(\Delta x_t) = 0$ in the case of a random walk.

Actually, the Dickey–Fuller tests are not sufficiently robust. A particular problem they pose is that they are not valid in the presence of serial correlation. Therefore, as an alternative, consider the more general specification:

$$\Delta x_t = (\lambda - 1)x_{t-1} + \sum_{i=1}^{k} \beta_i \Delta x_{t-1} + u_t$$

where it is assumed, for the particular choice of k that might be made, that u_t retains its stated white noise properties – because of the inclusion of the terms:

$$\sum_{i=1}^{k} \beta_i \Delta x_{t-i}.$$

The relevant test statistics, in this case the Augmented Dickey–Fuller, are the appropriately modified versions of the τ-statistics considered above. Originally proposed for use when the error terms are autoregressive of a known order (Dickey, 1979), they have since been shown to be less restrictively asymptotically valid (Phillips & Perron, 1988; Said & Dickey, 1984). However, as discussed by Davidson (2000), the properties of these statistics depend upon the ability to choose an appropriate value for k, the number of lagged values of Δx_t: "if the process generating Δx_t has moving average as well as autoregressive components, the true value k is actually infinite..." (p. 350).

```
Variable:  IGE
Unit Root Tests: Summary Results
  Both Drift and Trend
  Base Sample Period:1936-1954          Frequency: Annual
  Number of Observations =  19     Dependent Mean =     8.2368
```

	(λ −1)	Tau	T(λ −1)	AIC	SBC
DF	-0.6005	-2.5531	-11.4101	6.6815	6.8306
ADF(1)	-1.0428	-4.4887	-59.3110	6.2906	6.4885
ADF(2)	-1.5922	-4.7783	65.8894	6.0666	6.3116
ADF(3)	-1.9466	-3.2674	27.4977	6.0557	6.3454
ADF(4)	-2.4549	-2.6548	15.5951	6.2306	6.5610
ADF(5)	-3.8143	-3.2295	7.3977	6.0219	6.3871
ADF(6)	-5.7367	-4.0796	5.4007	5.2718	5.6629

```
 Unit Root Tests: Summary Results
  No Trend, Drift Only
  Base Sample Period:1936-1954          Frequency: Annual
  Number of Observations =  19     Dependent Mean =     8.2368
```

	(λ −1)	Tau	T(λ −1)	AIC	SBC
DF	-0.1714	-1.1252	-3.2568	6.8499	6.9493
ADF(1)	-0.2730	-1.5118	-7.0756	6.9344	7.0828
ADF(2)	-0.1294	-0.5769	-2.0702	7.0237	7.2197
ADF(3)	-0.1023	-0.4939	-1.0146	6.6366	6.8780
ADF(4)	-0.0877	-0.3690	-0.5389	6.7143	6.9975
ADF(5)	-0.0135	-0.0447	-0.0627	6.8997	7.2192
ADF(6)	0.1060	0.3338	0.2829	6.7984	7.1460

```
 Unit Root Tests: Summary Results
  Random Walk: No Drift, No Trend
  Base Sample Period:1936-1954          Frequency: Annual
  Number of Observations =  19     Dependent Mean =     8.2368
```

	(λ −1)	Tau	T(λ −1)	AIC	SBC
DF	0.0411	0.6343	0.7818	6.8737	6.9234
ADF(1)	0.0200	0.2711	0.4379	7.0106	7.1096
ADF(2)	0.0588	0.7823	0.8067	6.9654	7.1124
ADF(3)	0.1252	1.8839	1.0248	6.6265	6.8196
ADF(4)	0.1571	2.0423	0.8794	6.7043	6.9403
ADF(5)	0.1611	1.6998	0.7680	6.8087	7.0826
ADF(6)	0.2448	2.3259	0.6628	6.6874	6.9916

Fig. 6.1 Dickey-Fuller and augmented Dickey-Fuller unit root tests. *DF* Dickey-Fuller, *ADF(k)* Augmented Dickey-Fuller, based on k lag terms, *AIC* Akaike Information Criterion, *SBC* – Schwarz Criterion

Figure 6.1 displays, in summary form, both Dickey–Fuller and Augmented Dickey–Fuller statistics. These have been generated using as input observations on the Grunfeld variable IGE. This particular variable has been chosen simply to provide an easily replicable demonstration. Notice that the figure shows the statistics in the form of three distinct sets, the first corresponding to the case that both a constant term and a trend term are included, the so-called Drift and Trend case. The second set omits the trend term. The third omits as well the constant term. The Dickey–Fuller statistics are shown in the first row of each set, followed by the Augmented Dickey–Fuller in successive rows, with the number of lagged Δx_{t-i} terms progressively increasing from a single term to a total of six. In the case of each set, the column headed $T(\lambda - 1)$ should be understood to include this particular Dickey–Fuller statistic column element in the first row, but for the remaining rows the values represent instead the estimated Augmented Dickey–Fuller statistic

$$\frac{T(\lambda - 1)}{1 - \sum_{i=1}^{k} \gamma_i}$$

where k of course represents the number of lagged Δx_{t-i} terms, the number of lags indicated by the corresponding label ADF(k), $k = 1, 2, \ldots, 6$. The Akaike Information (AIC) and Schwarz (SBC) criteria are provided as supplementary test statistics that might be employed to aid in the determination of the appropriate number of lags. In addition, the appendix to this chapter displays the corresponding full set of underlying regressions so as to provide the ability to properly evaluate the display in Fig. 6.1.

The Augmented Dickey–Fuller statistics, although they accommodate serial correlation, have the limitation that this constitutes only certain characterizations. More accommodative alternatives include the Phillips–Perron unit root test statistics, which are intended to be valid in the presence of serial correlation of unknown form. These nonparametric statistics are best considered in relation to the ordinary Dickey–Fuller statistics. They are alternatives to the Augmented Dickey–Fuller, but do not involve an explicit specification of the properties of the error process, u_t, although they arguably serve to characterize aspects of these properties sufficiently well to provide a basis to consistently test for a unit root.

To begin, it is useful to consider the particular generalization (Davidson & MacKinnon, 1993):

$$\Delta x_t = \alpha + \beta t + \gamma t^2 + (\lambda - 1)x_{t-1} + u_t \equiv \mathbf{Z}\delta + \mathbf{u}$$

where $\mathbf{Z}$ is a $T \times m$ matrix containing columns of observations corresponding to the constant term α (a column of 1 s) and the variables t, t^2, and x_{t-1}, and $\mathbf{u}$ in turn is a $T \times 1$ vector. Considering this equation, let:

$$S_T^2 = \frac{1}{T}\sum_{t=1}^{T} \hat{u}_t^2$$

$$S_u^2 = \frac{1}{T}\sum_{t=1}^{T} \hat{u}_t^2 + \frac{2}{T}\sum_{j=1}^{L} w_{jL} \sum_{t=j+1}^{T} \hat{u}_t \hat{u}_{t-j}$$

where L is a number of lags parameter and $w_{jL} = 1 - j/(L+1)$. In order to compute the Phillips–Perron $\mathbf{z}^*$ test statistic, Greene (2003, p. 754), Stokes (1997, p. 332), and other software developers, following Hamilton (1994), choose to use the particular formula:

$$\mathbf{z}^* = \mathbf{z} - \frac{(T\sigma_\tau / S_{T-K})^2 \left(S_u^2 - S_T^2\right)}{2}$$

where

$$\mathrm{S}^2_{\mathrm{T-K}} = \frac{1}{T-K}\sum_{t=1}^{T}\hat{\mathrm{u}}^2_{\mathrm{t}}$$

$$\sigma^2 = \mathrm{S}^2_{\mathrm{T-K}}(\mathbf{Z}'\mathbf{Z}_{\mathrm{k,k}})^{-1}$$

$(\mathbf{Z}'\mathbf{Z}_{\mathrm{k,k}})^{-1}$ denotes the principal diagonal element of $(\mathbf{Z}'\mathbf{Z})^{-1}$ that directly corresponds to the location of the parameter $\lambda - 1$ on $\mathrm{x}_{\mathrm{t-1}}$ and $\mathbf{z}$ is the Dickey–Fuller statistic $\mathrm{T}(\lambda - 1)$. This expression of $\mathbf{z}^*$, among other things, demonstrates the relation of the Phillips–Perron statistic to this Dickey–Fuller statistic. The τ^* test statistic in turn can be computed using the formula:

$$\tau^* = \frac{\mathrm{S_T}\tau}{\mathrm{S_u}} - \frac{(T\sigma_\tau/S_{T-K})(S_u^2 - S_T^2)}{2S_u}$$

In each case, if serial correlation is absent, then the terms after the minus sign should each ideally evaluate as zero, because $\mathrm{S_T} = \mathrm{S_u}$, equating τ^* and τ as well as $\mathbf{z}^*$ and $\mathbf{z}$.

Notice also that the $\mathbf{z}^*$ statistic formula can be simplified to:

$$\mathbf{z}^* = \mathbf{z} - \mathrm{T}^2\omega(\mathrm{Z}'\mathrm{Z}_{\mathrm{k,k}})^{-1}$$

where

$$\omega = \sum_{i=1}^{L} w_{jL}\sum_{t=1}^{T}\frac{1}{T}\hat{\mathrm{u}}_{\mathrm{t}}\hat{\mathrm{u}}_{\mathrm{t-i}}$$

and that for τ^* to:

$$\tau^* = \frac{\mathrm{S_T}\tau}{\mathrm{S_u}} - \frac{T\omega\sqrt{(Z'Z_{\mathrm{k,k}})^{-1}}}{S_u}$$

The Phillips–Perron statistics obviously represent a modification of the Dickey–Fuller. As such, they have not, however, been free of “issues,” both as regards their computational aspects (Haldrup & Jansson, 2006; Davidson, 2009) and their application. Considering first the computational aspects, the formulae for the Phillips–Perron statistics have been stated in various different ways in the literature. As a consequence, the possibility of faulty computation might, in particular cases, arise from confusion due to the alternative statements of these that appear in econometric textbooks and elsewhere. For instance, in contrast to the Hamilton formulae given earlier, for the case that:

$$\mathrm{y_t} = \alpha\mathrm{y_{t-1}} + \mathrm{u_t}$$

Haldrup and Jansson (2006, pp. 256–257) have recently presented the formulae for these statistics as:

$$\mathbf{z}^* = \mathrm{T}(\lambda - 1) - \frac{(S_{Tl}^2 - S_T^2)}{2T^{-2}\sum_{t=2}^{T} y_{t-1}^2}$$

and

$$\tau^* = \frac{S_T{}^{\tau}}{S_{Tl}} - \frac{S_{Tl}^2 - S_T^2}{2\sqrt{S_{Tl}^2 T^{-2} \sum_{t=2}^{T} y_{t-1}^2}}$$

See also Davidson (2000, p. 350) and Davidson and MacKinnon (1993, pp. 712–713).

Because of the apparent strong differences, it is worth taking a moment to consider the relationship between these alternatives. This random walk case was the one originally considered by Phillips (Phillips, 1987), who proposed the statistics that are shown just above – essentially in the form presented by Haldrup and Jansson. As necessary, these can be restated in slightly simplified form as:

$$Z(\tau) = (S_T/S_{Tl})\tau - \frac{\omega}{\sqrt{s_{Tl}^2 T^{-2} \sum y_{t-1}^2}}$$

$$Z(\alpha) = T(a-1) - \frac{\omega}{T^{-2} \sum y_{t-1}^2}$$

where:

$$S_T^2 = T^{-1} \sum u_t^2$$

$$\omega = T^{-1} \sum_{j=1}^{L} \left(1 - \frac{j}{L+1}\right) \sum_{t=j+1}^{T} u_t u_{t-j}$$

$$s_{Tl}^2 = T^{-1} \sum_{t=1}^{T} u_t^2 + 2\omega$$

However, Hamilton has demonstrated (1994, pp. 506–515) that:

$$T^2 r_a^2 \,/\, s^2 \equiv \frac{1}{T^{-2} \sum y_{t-1}^2}$$

where r_a^2 is the estimated variance of the OLS estimator of α and s^2 is the estimated variance of the regression, which implies:

$$Z(\tau) = (s/s_{Tl})\tau - \frac{T(r_a/s)\omega}{s_{Tl}}$$

and

$$Z(\alpha) = T(a-1) - \frac{T^2 r_a^2 \omega}{s^2}$$

Consider next the "Drift" case of a Constant, but No Trend:

$$y_t = \mu + \alpha y_{t-1} + u_t$$

Phillips & Perron (1988) originally proposed the test statistic:

$$
\begin{aligned}
Z(\tau) &= (S_T/S_{Tl})\tau - S_{Tl}(\lambda/s_{Tl}^2)/\sqrt{T^{-2}\sum(y_t - ybar)^2} \\
&= (S/S_{Tl})\tau - (\lambda/S_{Tl})/\sqrt{T^{-2}\sum(y_t - ybar)^2} \\
&= (S_T/S_{Tl})\tau - (\omega/S_{Tl})/\sqrt{T^{-2}\sum(y_t - ybar)^2} \\
&= (S_T/S_{Tl})\tau - \frac{\omega}{\sqrt{S_{Tl}^2 T^{-2}\sum(y_t - ybar)^2}}
\end{aligned}
$$

and in a similar way derived:

$$
Z(\alpha) = T(a-1) - \frac{\omega}{T^{-2}\sum(y_t - ybar)^2}
$$

where:

$$
\begin{aligned}
S_T^2 &= T^{-1}\sum u_t^2 \\
\omega &= T^{-1}\sum_{j=1}^{L}(1 - \frac{j}{L+1})\sum_{t=j+1}^{T} u_t u_{t-j} \\
s_{Tl}^2 &= T^{-1}\sum_{t=1}^{T} u_t^2 + 2\omega \\
\text{ybar} &= T^{-1}\sum y_t
\end{aligned}
$$

But notice in this case, as Hamilton has demonstrated (1994, pp. 511–512), that:

$$
T^2 r_a^2/s^2 \equiv \frac{1}{T^{-2}\sum(y_t - ybar)^2}
$$

where r_a^2 is the estimated variance of the OLS estimator of α and s^2 is the estimated variance of the regression, which implies:

$$
Z(\tau) = (S_T/S_{Tl})\tau - \frac{Tr_a/s\omega}{S_{Tl}}
$$

and

$$
Z(\alpha) = T(a-1) - \frac{T^2 r_a^2 \omega}{s^2}
$$

Finally, consider the case that:

$$y_t = \mu + \beta t + \alpha y_{t-1} + u_t$$

In the event, Phillips and Perron (1988) also introduced the slight complication that they chose to define the trend term as t – ½T (which involves simply a respecification of the observations on what is here called t). They proposed for this case the statistics:

$$\begin{aligned} Z(\tau) &= (S_T/S_{Tl})\tau - s_{Tl}(\lambda/S_{Tl})/\sqrt{M} \\ &= (S_T/S_{Tl})\tau - (\lambda/S_{Tl})/\sqrt{M} \\ &= (S_T/S_{Tl})\tau - \omega/(S_{Tl})\sqrt{M} \\ &= (S_T/S_{Tl})\tau - \frac{\omega}{S_{Tl}\sqrt{M}} \end{aligned}$$

and

$$Z(\alpha) = T(a-1) - \frac{\omega}{M}$$

where:

$$\begin{aligned} S_T^2 &= T^{-1}\sum u_t^2 \\ \omega &= T^{-1}\sum_{j=1}^{L}\left(1-\frac{j}{L+1}\right)\sum_{t=j+1}^{T} u_t u_{t-j} \\ S_{Tl}^2 &= T^{-1}\sum_{t=1}^{T} u_t^2 + 2\omega \\ M &= (1-T^{-2})T^{-2}\sum y_t^2 - 12(T^{-5/2}\sum t y_t)^2 \\ &\quad +12(1+T^{-1})\,(T^{-5/2}\sum t y_t(T-3/2\sum y_t) \\ &\quad -(4+6T^{-1}+2T^{-2})(T-3/2\sum y_t)^2 \end{aligned}$$

But as Hamilton has demonstrated:

$$T^2 r_a^2/s^2 \equiv \sqrt{a^2+b^2}\frac{1}{\sqrt{M}}$$

which implies:

$$Z(\tau) = (S_T/S_{Tl})\tau - \frac{T(r_a/s)\omega}{S_{Tl}}$$

and

$$Z(\alpha) = T(a-1) - \frac{T^2 r_a^2 \omega}{s^2}$$

The remaining differences are purely notational, since $S_u \equiv S_{Tl}$.

Of course, the Hamilton formulae displayed here only appear to stay constructively the same from case to case, a result of their statement in terms of particular symbols. Although in each instance the symbols r_a^2 and s^2 retain their conceptual meaning – as the estimated variance of the OLS estimator of α and the estimated variance of the regression, respectively – they differ in precise constructive form and value in each case. But, the computational attraction of these formulae is neither how they look superficially nor the underlying computations the symbols hide. What is pertinent computationally is that these formulae take account of the fact that the appropriate values of r_a^2 and s^2 are ordinarily generated automatically during the process of performing each regression, as is also the value of the relevant principal diagonal element of $(\mathbf{Z}'\mathbf{Z})^{-1}$. The values are therefore at hand to be "plugged into" the Hamilton formulae for the Phillips–Perron statistics in order to generate the appropriate test values for these.

However, this calculation process, although seemingly straightforward, is not numerically trivial. The possible difficulties are demonstrated by the fact that a number of differences were originally discovered in the values generated by different econometric software packages. Upon investigation, certain of these disparities were traced to particular interpretations made by individual developers, but once common formulae were implemented it was then discovered that the remaining problems were numerical – that the test values generated can be quite sensitive to the numerical precision of the individual calculations made. In particular, calculating the term:

$$T^{-1}\sum_{j=1}^{L}\left(1-\frac{j}{L+1}\right)\sum_{t=j+1}^{T}u_t u_{t-j}$$

involves, among other things, first the need to insure that $j/(L+1)$ is *not* calculated as a integer divide – which would result in this ratio being evaluated uniformly as 0. In addition, the component terms of this expression may in particular cases evaluate as rather small, in some ways potentially similar values, making the precision of each of the subordinate calculations an important practical consideration. Notice also that the covariance terms may differ in sign. When testing the code, it is therefore especially important to examine multiple examples, rather than simply to infer from a single successful worked example that the calculations are appropriately accurate. As a case in point, the worked examples supplied by Hamilton, using treasury bill interest rates (Hamilton, 1994, pp. 506–515), were uniformly easy for each tested software package to match. In contrast, other cases, including the example that will be displayed in Fig. 6.2, sometimes proved to be a significant hurdle.

Assuming that the formulae have been algorithmically implemented sufficiently precisely, the matter of the use and application of the Phillips–Perron statistics then becomes the primary concern. Both size and power properties have been found to be problems, as has also any occurrence of structural breaks in the input data series, not to mention what has been termed the "elusiveness" of the concept of the "non-integrated" process (Davidson, 2009). Recent relevant survey articles include that by Haldrup and Jansson (2006). Perron (2006) has considered the problem of changes in structural trends. Previously, Schwert (1989), among others, found – following

```
Variable: Ln(GDP)

 Unit Root Tests: Summary Results
  Both Drift and Trend                    PP Lag Truncation No = 4
  Base Sample Period:195003-200004        Frequency: Quarterly
  Number of Observations = 202     Dependent Mean =      0.0085

              (λ-1)       Tau         Zstat        AIC         SBC
  DF         -0.0415    -2.3212     -8.3842     -9.2404     -9.1913
  PP                    -2.7589    -13.0670    (L = 4)
  ADF(1)      0.0434    -2.5885    -13.2269     -9.3910     -9.3253
  ADF(2)      0.0469    -2.7442    -15.5770     -9.3829     -9.3004
  ADF(3)      0.0463    -2.6484    -14.8711     -9.3705     -9.2712
  ADF(4)      0.0426    -2.3894    -12.5141     -9.3613     -9.2451
  ADF(5)      0.0372    -2.0623     -9.5022     -9.3614     -9.2281
  ADF(6)      0.0408    -2.2319    -11.5952     -9.3549     -9.2044

 Unit Root Tests: Summary Results
  No Trend, Drift Only                    PP Lag Truncation No = 4
  Base Sample Period:195003-200004        Frequency: Quarterly
  Number of Observations = 202     Dependent Mean =      0.0085

              (λ-1)       Tau         Zstat        AIC         SBC
  DF         -0.0017    -1.1941     -0.3484     -9.2256     -9.1928
  PP                    -0.9998     -0.3765
  ADF(1)     -0.0008    -0.5666     -0.2284     -9.3685     -9.3192
  ADF(2)     -0.0007    -0.5420     -0.2315     -9.3559     -9.2900
  ADF(3)     -0.0008    -0.6102     -0.2470     -9.3459     -9.2631
  ADF(4)     -0.0008    -0.5504     -0.2025     -9.3428     -9.2432
  ADF(5)     -0.0007    -0.5009     -0.1610     -9.3500     -9.2334
  ADF(6)     -0.0008    -0.5408     -0.1866     -9.3396     -9.2058

 Unit Root Tests: Summary Results
  Random Walk: No Drift, No Trend         PP Lag Truncation No = 4
  Base Sample Period:195003-200004        Frequency: Quarterly
  Number of Observations = 202     Dependent Mean =      0.0085

              (λ-1)       Tau         Zstat        AIC         SBC
  DF          0.0010    12.1174      0.2056     -9.2176     -9.2012
  PP                     9.2991      0.2055
  ADF(1)      0.0007     6.4790      0.1994     -9.3728     -9.3399
  ADF(2)      0.0006     5.5213      0.1978     -9.3608     -9.3113
  ADF(3)      0.0007     5.4252      0.1975     -9.3498     -9.2836
  ADF(4)      0.0007     5.5409      0.1957     -9.3469     -9.2639
  ADF(5)      0.0008     5.8659      0.1938     -9.3539     -9.2539
  ADF(6)      0.0008     5.0220      0.1932     -9.3435     -9.2264
```

Fig. 6.2 Unit root tests: Dickey-Fuller and Phillips-Perron statistics. *DF* Dickey-Fuller, *PP* Phillips-Perron, *ADF(k)* Augmented Dickey-Fuller, based on k lag terms, *AIC* Akaike Information Criterion, *SBC* Schwarz Criterion

from a general discovery that non-parametric test statistics of the Phillips–Perron type can be under-sized – that the Augmented Dickey–Fuller test statistics characteristically have critical values that are considerably more robust in the context of moving average terms and time trends. In other contexts, the finite sample properties of the Phillips–Perron statistics have been found to be problematic.

Another aspect of these unit root tests that needs to be recognized – as has been considered by Davidson and MacKinnon (1993) and previously by Ghysels and Perron (1993) – is that the statistics are biased against rejecting the null when the observations on x_t have been seasonally adjusted using either a linear filter or the methods commonly employed by government statistical agencies. More generally, as just indicated, the data generating process for x_t needs to be stable over the entire sample period, t = 1,2, . . . ,T. Specifically, Perron (1989) has shown that exogenous shocks to series values can result in a significant reduction in the power of unit root tests. Furthermore, it is important to recognize that such shocks could, for example, take the form not only of a shock to the modeled economic process, but also simply a redefinition of the measured series during the sample period, as might for instance occur with long runs of employment data, reflecting that industry categories are reclassified from time to time. Shocks to the modeled process itself can include both "big" shocks, such as the great depression of 1930 and world wars, but also such things as periods of change in labor productivity (Perron, 2006, p. 323–324).

In the context of a particular economic specification, as will be discussed, these unit root tests can be directly related to cointegration to the degree that it is possible to use them to test for a cointegrating vector, α, such that:

$$\alpha' \mathbf{x}_t = u_t$$

which can be seen to involve an OLS regression, possibly as the first step of the Engle–Granger two-step estimator, which might be done in order to parameterize a long run equilibrium relationship among the $\mathbf{x}_t$. Furthermore, which of the elements of $\mathbf{x}_t$ is chosen as the dependent variable may not matter asymptotically, but in practice, especially when the sample is small, alternative choices possibly should be considered. Obviously, such considerations may need to be taken into account as a matter of software design.

On the computational side, a matter that additionally needs consideration, and is particularly important in the present context, is that the Phillips–Perron statistics are computationally somewhat arbitrary, particularly to the degree that software developers choose to "improve" these test statistics while at the same time retaining this given name, possibly leading users to make unwarranted inferences. Davidson & MacKinnon (1993), among others, have pointed out that neither the number of specified lags, L, nor the w_{jL} weights themselves are necessarily uniquely defined. In concept, the Phillips–Perron statistics are intended to be valid in the presence of serial correlation of unknown form, but there is a tendency among theoretical econometricians not to view the specific weights scheme originally selected by Phillips and Perron as necessarily defining the essence of these statistics. As a consequence of this indeterminacy, a uniquely correct specification of the weights arguably does not exist. The opinion is sometimes expressed that "Of course, there is no single 'PP statistic', just a large family of statistics that (we theorize!) share a common pivotal limiting distribution under H0. There is accordingly a slight sense of get-any-answer-you-like, as so often with semi-parametric methods… The choice of kernel and bandwidth is [or should be] user-selectable. You can also adjust for trends, etc."

Indeed, on this telling, it might seem that Phillips originally and later Phillips and Perron made their particular choice of weights (albeit from a restricted choice set) simply to insure that the estimator of S^2_{Tl} will be positive, as well as consistent. Therefore, as just indicated, these weights can be interpreted to represent an arbitrary, if qualified choice, defined by properties – rather than a necessary component of the specific formulae originally proposed. It is consequently apparently permissible for software developers, as well as econometricians, to justify their own, independent choice of weighting scheme, compatible with the criterion that an estimator of S^2_{Tl} should be chosen so as to be a consistent estimator, possibly among other properties. Clearly, this consistency qualification imposes an important restriction on the set of possible choices, but notice that, given the uncertainty associated with the definition and use of each set of weights w_{jL}, all other things equal it is in practice possible (is even to be expected) for different software packages to display different values of these statistics even if the underlying economic dataset used is the same. Does it therefore follow that simply knowing that a test statistic so displayed belongs to the "Phillips–Perron" class (especially if presented together with a "p" value) provides sufficient information about the inferences to draw? Is this a context in which all that needs to be known is immediately evident from two or even three numbers on a computer screen to which a class label is attached?

Of course, from the point of view of the theoretical econometrician, these two questions are likely to be considered misguided. From this position, it might be argued that, among other things, they betray a dated perspective on what properly constitutes a unit root test, by implicitly equating "test = statistic". In contrast, the Durbin–Watson statistic and the "F-statistic of the regression," as classic tests that are directly defined by the statistics themselves, admit the possibility of a common agreement as to their "correct" values. As such, these can be held to the standard that each econometric software package should generate (within an appropriate tolerance) the same values, ceteris paribus. However, the context of the modern unit root test is that of the asymptotic test, which arguably exists in the form of multiple test statistics, each subject to decisions made about lags, kernels, bandwidth, and other properties. In addition, there are certain subtleties, such as the treatment of initial conditions, which in a theoretical context will tend to be ignored inasmuch as they are asymptotically irrelevant, yet which in a finite sample will affect the test statistic values that are generated. This argument leads to skepticism concerning the degree of uniformity it is reasonable to expect among the test values reported by different software packages. If pursued further, it would refocus attention onto such software design questions as, how much control does the user have over the statistics a particular program produces, as well as the quality of the program's documentation and how well it serves to provide the user insight concerning his or her control choices? It might even lead to the construed suggestion that the software developer, or some other person directly concerned with the application of the test, possibly the software user, should be the only one to define the specific characteristics of the initial conditions, relieving the theoretician of the burden of ever considering these.

It may indeed be true that to survey the existing econometric software packages with the stated aim of discovering which specific test statistics they produce to some degree puts the cart before the horse. There is undoubtedly a need for additional studies that focus more specifically upon categories of tests and that approach the topic in this more top down fashion, taking account of the range of testing facilities offered and what provision is made by each package to provide the user with a satisfactory testing environment. At the same time, it is also true that a child must crawl first. A counter question that might be asked is whether the economics profession as a whole is yet ready to embrace one or more econometric software packages that impose the significant knowledge requirement that a high degree of user control implies, even if such packages supply sufficient associated documentation to provide insight concerning a user's possible choices? There are of course a number of econometric programming environments available today, ranging from computer programming languages to econometric programming languages that might be seen to include Gauss, Mathematica, Ox, S-Plus, and others. Collectively, these offer extensive user control, but they also simultaneously impose the requirement that users bring to the task the knowledge of a theoretical econometrician plus the motivation to bridge the gap between theory and practice. What appears to be missing are programs that inhabit the middle ground, that on the one hand free the user from the requirement to build from scratch, yet at the same time offer both a high degree of user control over the specific tests performed, a wide range of appropriate testing facilities, and comprehensive, accessible documentation.

It is important to view the various issues from the proper perspective and to address the question of unit root tests from a modern vantage point. However, once the focus becomes a particular, named test statistic, particularly one the constructive characteristics of which are able to be determined, the question of where that statistic fits into the strategy of unit root tests is actually quite an abstract question, one that may presuppose a more nearly ideal world of the future. For years, theoretical econometricians have been in the process of discovering what might someday be a clearly distinguishable, relatively comprehensively defined suite of unit root tests, each subject to decisions made about lags, kernels, bandwidth, and other properties (including, when applied, initial conditions) but so far this discovery process has generally proceeded one test statistic at a time. Even among theoreticians the development of this testing strategy has in many respects been bottom up, certainly in the way that candidate test statistics have been introduced in the literature. A consequence is that econometric software developers, eager to stay abreast of theoretical progress, have progressively incorporated particular constructions into their programs, and thus made them available one by one to their users. Therefore, today, it is quite justifiable to ask the question whether multiple, constructively different test statistics should be associated with the same name designation without very careful and explicit qualification, especially in the context of the use of econometric software that to many users may have a distinct "black box" quality. In this context it is imperative to consider the practical argument that if a test statistic is coded into software meant to be used generally for applied economic research its constructive properties should be specifically defined, both in order that its algorithmic implementation can be validated – and so that users of the software package will have a fighting chance to understand the limits to the inferences they might draw using this particular test statistic.

Quite possibly the time has come for all interested parties to begin to think of tests in the way just described, particularly in the case of asymptotic tests, rather than to present a variety of seemingly separated test statistics in the context of a table, like Table 6.1 above. But, at the moment, the pertinent issue is simply the validation of the algorithmic implementation of a given formula or formulae. Later, when related alternatives are considered, it is always possible to distinguish nominally between constructively different test statistics, if only in terms of classified categories such as "Phillips-Perron A", "Phillips-Perron B", and so on. But, from the perspective of the econometric software developer, whether the user might choose to use version B rather than A is not nearly as important *in the present context* as the consideration that, in the case of each constructively established version, a definite test statistic value should be produced given a particular set of input observations and initial conditions. Of course, this argument sidesteps the normative question whether it is irresponsible to include any test statistic in a software package intended for general use that does not meet certain criteria that determine its suitability for such use?

In the case of the Phillips-Perron test statistics, τ^* and $\mathbf{z}^*$, the particular original choice of weights $[w_{jL} = 1\text{–}j/(L + 1)]$, which was adopted from Newey and West (1987), can be construed to imply that L should generally be on the order of $T^{1/4}$, which establishes a guideline, but not an absolutely firm choice. Obviously L will need to be an integer, so that for example if $T = 201$, then $T^{1/4} = 3.77$ – in this case, does one choose 3 or 4? In addition, Newey and West (p. 705) point out that under some circumstances a value less than $T^{1/2}$ "will suffice for consistency" of the estimator S^2_{Tl}, which under those circumstances would appear to imply a choice of $L = 14$. Of course, at the moment, what is most important initially is not the details associated with the particular choice of L, but rather the design issue that the requirement to make this choice poses for econometric software developers and perhaps even for the applied economist who uses the software. The software developer can choose and enforce a criterion – such as the nearest integer to $T^{1/4}$ – or simply allow each user to make his or her own choice. However the latter option obviously opens the door to the possibility that a user might make an inappropriate choice that may result in an inappropriate computation of the Phillips–Perron statistics. But this possibility is best left for later consideration.

An additional, obviously important aspect of the computation of the Phillips–Perron statistics is the need to use a sufficient number of observations, certainly more than the 20 observations available on each of the variables in the Grunfeld dataset. Therefore, in order to provide a contextually defensible benchmark example, Fig. 6.2 displays a set of summary unit root statistics, including Phillips–Perron statistics, that have been generated using quarterly data on the natural log of GDP for the period from 1950 to 2004. These particular observations have previously been employed by Greene (2003, p. 646), following earlier use by Cecchetti and Rich (2001). The untransformed set of GDP observations can themselves be found, in hard copy form, in the Appendix to the present volume. They can also be found, in machine-readable form, in the context of an Excel spreadsheet file, at http://pages.stern.nyu.edu/~wgreene/Text/econometricanalysis.htm. This file is identified there as containing the data displayed in Appendix Table F5.1 of the Greene textbook.

In a similar way to the earlier treatment in Fig. 6.1, Fig. 6.2 displays the statistics as three distinct sets, the first corresponding to the case in which a constant term and a trend term are both included, the so-called Drift and Trend case. The second set omits the trend term. The third set also omits the constant term. The Dickey–Fuller statistics are shown as the first row of each set, followed in the next row by the Phillips–Perron statistics and then in successive rows the Augmented Dickey–Fuller

(ADF) statistics, with the number of lagged Δx_{t-i} terms progressively increasing from a single term to a total of six. The headings Tau and Zstat refer to the Dickey–Fuller or Phillips–Perron test statistics, as relevant. The Akaike Information (AIC) and Schwarz (SBC) statistics are provided as an aid to determine the appropriate number of the ADF lags. In addition, so as to provide the ability to properly interpret the display in Fig. 5.3, the appendix to this chapter displays the corresponding full set of underlying regressions.

The Phillips–Perron statistics displayed in Fig. 6.2 are computed employing the Hamilton formulae discussed earlier, with the weights lag value, L, set equal to 4; that is, the number of observations, T, is 202, so that $T^{1/4} = 3.77$, which when rounded up implies $L = 4$. Phillips has called L the Lag Truncation Number (Phillips, 1987, p. 285), and it is so identified in the figure. Given that the aim at present is merely to provide a computational benchmark, the possibility that these GDP data have been seasonally adjusted is not here a matter of concern.

The values displayed in Fig. 6.2 are those generated by a single econometric software package, specifically MODLER. However, they have been replicated by other packages as well and it is important to understand the nature and characteristics of this particular benchmarking process as it transpired. In the preceding discussion in this chapter, the several statistics the calculated results of which have been considered – the Dickey–Fuller, the Augmented Dickey–Fuller, and the Phillips–Perron – were described as being generated using particular formulae. This phrasing is expositionally convenient, but when considering the specific values displayed in Figs. 6.1 and 6.2, it is necessary to recognize that these *formulae* are actually directives in the sense of the discussion in Chap. 2. In order to generate the values shown, algorithms first had to be created that express the essential aspects of these formulae in a constructive and numerically precise way. Of course, additional algorithms had to be created in order to compute other values that these algorithms require as "input data." Furthermore, once the calculations were made, the establishment of the benchmark values shown in Figs. 6.1 and 6.2 initially involved an intensive comparison of the results generated by four software packages, in particular B34S, MODLER, RATS and SAS.

These four programs have been created independently of each other by different people, or groups of people, and do not share any source code in common, although it is possible that for certain of these programs (but not all) the same, publicly available mathematical library routines may have been used. Moreover, there has not been any sharing of information concerning the specific algorithmic characteristics of this source code between the developers, which is also true of all the other numbers displayed in this volume. *What* has been done in the case of each program, and *why*, have each, to a degree, been the subject of conversation between developers, but such conversations have not included any discussion of the algorithmic strategies employed nor has there been any information transfer concerning the characteristics or specific form of the particular algorithms created and employed. In addition, at least three different computer programming languages have been used in this process, including Assembler, Fortran, and some variant or variants of the C language, involving, in the case of Fortran, the use of different compilers and linkers by different developers. Nevertheless, the results that have been generated by each package match at the level of numerical precision displayed in Figs. 6.1 and 6.2. Subsequently, comparisons with other independently developed software packages have also been made, with similar agreement. Of course, a priori, it might be expected that if the same observations are input and the calculations performed are performed so as to conform to precisely specified directive formulae using sufficient precision during the calculations that the values ultimately generated would be the same. It is nonetheless satisfying to achieve this sameness.

However, even when different developers working independently are able to generate results in common, there will inevitably still be differences in the output displayed by each package. For example, some developers might require the user to choose to generate particular results whereas others will display them automatically. For instance, it was mentioned earlier that the lag value, L, might either be determined by default, in a formulaic way, or can be left to the user's choice. Obviously, once having coded the necessary computational algorithms, it is possible for a developer to organize them so as to display automatically a set of Phillips–Perron statistics that correspond to a sequence of values of L, in much the same way that in Figs. 6.1 and 6.2, the Augmented Dickey–Fuller statistics are displayed automatically for more than one lag by some, perhaps even most, econometric software packages.

Consider, therefore, Fig. 6.3a, which displays the Phillips–Perron statistics that are associated with progressive values of L, where $L = 2, 3, \ldots, 7$. Seen simply as a set of benchmark calculations, the values shown in Fig. 6.3a are arguably useful towards the future development and validation of econometric software. However, from the perspective of the applied economist as a user, or even as a matter of fostering good applied econometric practice (or perhaps failing to), it is not entirely obvious what benefit this display provides in those cases that it is produced as a default. Some packages even go so far as to display statistics for the sequence $L = 0, 1, 2, \ldots$ As a display of the effect on the computed values of the Phillips–Perron statistics as L is allowed to vary over a range, such a display is possibly interesting. Phillips (1987, p. 286) originally considered the estimator:

$$\mathrm{S_u^2} = \frac{1}{T}\sum_{t=1}^{T}\mathrm{\hat{u}_t^2} + \frac{2}{T}\sum_{j=1}^{L}\sum_{t=j+1}^{T}\mathrm{\hat{u}_t\hat{u}_{t-j}}$$

in the context of his Theorem 4.2, in terms of the idea that it provides a consistent estimator (letting $L = o(T^{1/4})$, as $T \uparrow \infty$), apparently before instead adopting the Newey-West weighted variance estimator form, on the grounds that this thus-modified variance estimator is necessarily non-negative. However, just how the applied economist can productively use a display like Fig. 6.3a is not clear. In particular, might there not be some temptation for the novice to "cook" the results by choosing opportunistically? Recall the discussion in Chap. 2 of the use of econometric software by economists and the information about that use that is (more to the point, actually is not) commonly provided.

```
Variable: Ln(GDP)

      Unit Root Tests: Summary Results
        Both Drift and Trend
        Base Sample Period:195003-200004     Frequency: Quarterly
        Number of Observations = 202    Dependent Mean =      0.0085

                                   Tau        Zstat      L
       Phillips-Perron          -2.6634     -11.9961     2
       Phillips-Perron          -2.7337     -12.7811     3
       Phillips-Perron          -2.7589     -13.0670     4
       Phillips-Perron          -2.7475     -12.9370     5
       Phillips-Perron          -2.7318     -12.7594     6
       Phillips-Perron          -2.7125     -12.5426     7

       Unit Root Tests: Summary Results
        No Trend, Drift Only
        Base Sample Period:195003-200004     Frequency: Quarterly
        Dependent Mean =      0.0085

                                Tau         Zstat         L
       Phillips-Perron          -1.0268     -0.3708       2
       Phillips-Perron          -1.0051     -0.3753       3
       Phillips-Perron          -0.9998     -0.3765       4
       Phillips-Perron          -1.0071     -0.3749       5
       Phillips-Perron          -1.0161     -0.3730       6
       Phillips-Perron          -1.0268     -0.3709       7

       Unit Root Tests: Summary Results
        Random Walk: No Drift, No Trend
        Base Sample Period:195003-200004     Frequency: Quarterly
        Dependent Mean =      0.0085

                                   Tau        Zstat         L
        Phillips-Perron           9.7417      0.2055        2
        Phillips-Perron           9.4003      0.2055        3
        Phillips-Perron           9.2991      0.2055        4
        Phillips-Perron           9.3792      0.2055        5
        Phillips-Perron           9.4808      0.2055        6
        Phillips-Perron           9.6029      0.2055        7
```

Fig. 6.3 **(a)** Phillips-Perron statistics letting the number of lags vary

Considering once again the statistics displayed in Fig. 6.2, the two Fig. 6.3b, c, respectively, display the appropriate values of the autocovariance weight terms and values that are *miscalculated* due to the (purposeful) calculation of $1-\mathrm{j}/(\mathrm{L}+1)$ by making $\mathrm{j}/(\mathrm{L}+1)$ an integer divide, Since $j < (\mathrm{L}+1)$, by construction, $\mathrm{j}/(\mathrm{L}+1) = 0$ when calculated in this way. The effect on the value of the computed Phillips–Perron statistics is easily seen by comparing the values of these statistics displayed in these two figures. The rows labeled CVT(1) ... CVT(4) display in the column below the "Tau" statistic the product of each of the weights and the corresponding autocovariance term:

$$\mathrm{T}^{-1}\sum_{t=j+1}^{T}\hat{\mathrm{u}}_{\mathrm{t}}\hat{\mathrm{u}}_{\mathrm{t-j}}$$

```
Variable: Ln(GDP)

 Unit Root Tests: Summary Results
   Both Drift and Trend                 PP Lag Truncation No = 4
   Base Sample Period:195003-200004   Frequency: Quarterly
   Number of Observations = 202    Dependent Mean =      0.0085

                              Tau        Zstat
   Phillips-Perron         -2.7589    -13.0670

                                       1 - j/(L+1)
   CVT(1)    (x 10^4)        0.3188     0.8000       0.3188
   CVT(2)    (x 10^4)        0.1565     0.6000       0.1565
   CVT(3)    (x 10^4)        0.0195     0.4000       0.0195
   CVT(4)    (x 10^4)       -0.0679     0.2000      -0.0679
   2ω Value  (x 10^4)        0.6864


  Unit Root Tests: Summary Results
   No Trend, Drift Only                 PP Lag Truncation No = 4
   Base Sample Period:195003-200004   Frequency: Quarterly
   Number of Observations = 202    Dependent Mean =      0.0085

                              Tau        Zstat
   Phillips-Perron         -0.9998     -0.3765

                                       1 - j/(L+1)
   CVT(1)    (x 10^4)        0.3138     0.8000       0.3138
   CVT(2)    (x 10^4)        0.1450     0.6000       0.1450
   CVT(3)    (x 10^4)        0.0036     0.4000       0.0036
   CVT(4)    (x 10^4)       -0.0868     0.2000      -0.0868
   2ω Value  (x 10^4)        0.6442


  Unit Root Tests: Summary Results
   Random Walk: No Drift, No Trend      PP Lag Truncation No = 4
   Base Sample Period:195003-200004   Frequency: Quarterly
   Number of Observations = 202    Dependent Mean =      0.0085

                              Tau        Zstat
   Phillips-Perron          9.2991      0.2055

                                       1 - j/(L+1)
   CVT(1)    (x 10^4)        0.3253     0.8000       0.3253
   CVT(2)    (x 10^4)        0.1545     0.6000       0.1545
   CVT(3)    (x 10^4)        0.0129     0.4000       0.0129
   CVT(4)    (x 10^4)       -0.0791     0.2000      -0.0791
   2ω Value  (x 10^4)        0.6845
```

Fig. 6.3 (b) Values of Phillips-Perron test statistics weights

The item in the column to the right of this product is clearly the value of that weight. The rightmost column is the value of this autocovariance term. Figure 6.3b, c, when viewed comparatively, make the effect of the integer divide (and the Bartlett weights) quite obvious, in a way that viewing Fig. 6.3c on its own (showing the putative Phillips–Perron statistics) does not.

Of course, the intermediate values shown in these figures are not ordinarily displayed, inasmuch as their constructive purpose was to be used during the original program development and debugging process. It is perhaps even less common

```
Variable: Ln(GDP)

 Unit Root Tests: Summary Results
  Both Drift and Trend                PP Lag Truncation No = 4
  Base Sample Period:195003-200004    Frequency: Quarterly
  Number of Observations = 202    Dependent Mean =     0.0085

                           Tau      Zstat
  Phillips-Perron        -2.8580    -14.2102

  CVT(1)    (x 10^4)      0.3188      1.0000      0.3188
  CVT(2)    (x 10^4)      0.1565      1.0000      0.1565
  CVT(3)    (x 10^4)      0.0195      1.0000      0.0195
  CVT(4)    (x 10^4)     -0.0679      1.0000     -0.0679
  2ω Value  (x 10^4)      0.8540

 Unit Root Tests: Summary Results
  No Trend, Drift Only                PP Lag Truncation No = 4
  Base Sample Period:195003-200004    Frequency: Quarterly
  Number of Observations = 202    Dependent Mean =     0.0085


                              Tau        Zstat
  Phillips-Perron        -0.9802     -0.3811

  CVT(1)    (x 10^4)      0.3138      1.0000      0.3138
  CVT(3)    (x 10^4)      0.1450      1.0000      0.1450
  CVT(3)    (x 10^4)      0.0036      1.0000      0.0036
  CVT(4)    (x 10^4)     -0.0868      1.0000     -0.0868
  2ω Value  (x 10^4)      0.7511

 Unit Root Tests: Summary Results
  Random Walk: No Drift, No Trend    PP Lag Truncation No = 4
  Base Sample Period:195003-200004   Frequency: Quarterly
  Number of Observations = 202    Dependent Mean =     0.0085

                             Tau        Zstat

  Phillips-Perron          8.9246      0.2055

  CVT(1)    (x 10^4)      0.3253      1.0000      0.3253
  CVT(2)    (x 10^4)      0.1545      1.0000      0.1545
  CVT(3)    (x 10^4)      0.0129      1.0000      0.0129
  CVT(4)    (x 10^4)     -0.0791      1.0000     -0.0791
  2ω Value  (x 10^4)      0.8270
```

Fig. 6.3 **(c)** Phillips-Perror statistics with miscalculated weights

to present values knowingly that have been miscalculated; however, Fig. 6.3b, c viewed together stand as testimony to the possible benefit of occasionally doing this just to get a feel for the consequences. From these figures it is clear that the alternative test statistic values are not as different as might have been thought a priori. A similar degree of difference occurs when the statistics are instead alternatively calculated more and less precisely. It is not immediately obvious, when looking at sets of nominally identical test statistics that differ in value from one program to the next, whether these differences are necessarily the consequence of computational error or might instead reflect that one or more developers have chosen to produce idiosyncratic test statistics.

The values of the unit root statistics most commonly computed by the surveyed packages has been the focus of discussion so far, but it is evident from Table 6.1 that there are two additional test statistics that are each generated by at least six econometric software packages, namely those proposed by Kwiatkowski, Phillips, Schmidt, & Shin (1992) (KPSS) and by Elliot, Rothenberg, & Stock (1996) (ERS). The first of these provides an interesting contrast. As Davidson (2000, p. 355) remarked when beginning his textbook discussion of the KPSS test statistics, a "feature of the Dickey–Fuller and Phillips–Perron tests is that they make the unit root the null hypothesis. This is a natural choice when we think of nesting the unit root model in the family of autoregressive forms, but has the drawback that a lack of test power may encourage practitioners to 'find' unit roots when none exist." Davidson suggests that the test presented by Kwiatkowski, Phillips, Schmidt, and Shin might, in contrast, be seen as being motivated by the answer to the question "What is the particular characteristics of an I(0) process?" The answer could be a "process that, when integrated, yields a process with exactly one unit root." Kwiatkowski et al. (1992) imply as much both by the title of their paper – which indicates their consideration of stationarity as the null hypothesis – and by the use of a question as a subtitle, "How can we be sure that economic time series have a unit root?" Towards this end, they consider what Phillips and Xiao (1998) later characterized as a components model:

$$\mathrm{y_t = h_t + y_t^s + v_t}$$
$$\mathrm{y_t^s = y_{t-1}^s + u_t}$$

where $\mathrm{y_t}$ is implied to be decomposed into a deterministic trend, $\mathrm{h_t}$, a stochastic trend, $\mathrm{y_t^s}$, and a stationary residual, $\mathrm{v_t}$. In the general case $\mathrm{h_t} = \lambda' \mathrm{x_t}$, where $\mathrm{x_t}$ is a k-vector of deterministic trends. As KPSS indicate the "test is the LM test of the hypothesis that the random walk has zero variance" (p. 159). They also argue, as a motivation for this approach, that the "way in which classical hypothesis testing is carried out ensures that the null hypothesis is accepted unless there is strong evidence against it" (p. 160), having noted the prior prevalence of the finding, in many empirical studies (DeJong, Nankervis, Savin, & Whiteman, 1989; Nelson & Plosser, 1982), that "many or most aggregate economic time series contain a unit root." By effectively establishing as the null $\lambda = 0$ in contrast to the tests considered earlier, KPSS seek to provide a useful counter test.

The test algorithms that appear to have been implemented in the existing econometric software packages start from the more specific representation:

$$\mathrm{y_t} = \alpha + \beta \mathrm{t} + \lambda \sum_{i=1}^{t} x_i + \mathrm{u_t}$$

and the separate OLS estimation of both α and α and β, in each case setting $\lambda = 0$, which yields the residuals:

$$\hat{\mathrm{u}}_\mathrm{t} = \mathrm{y_t} - \alpha$$
$$\hat{\mathrm{u}}_\mathrm{t} = \mathrm{y_t} - \alpha - \beta \mathrm{t}$$

Then, given separately computed partial sums, each computed as:

$$U_t = \sum_{i=1}^{t} \hat{u}_i,\ t = 1, 2, \ldots, T$$

such that $U_T = 0$, the KPSS test statistic in these "NoTrend" and "Trend" cases can be expressed as:

$$\text{KPSS (L)} = \frac{\sum_{t=1}^{T} U_t^2}{T^2 S_u^2}$$

where

$$S_u^2 = \frac{1}{T}\sum_{t=1}^{T} \hat{u}_t^2 + \frac{2}{T}\sum_{j=1}^{L}\left(1 - \frac{j}{L+1}\right)\sum_{t=j+1}^{T} \hat{u}t\, \hat{u}_{t-j}$$

Figure 6.4a, b display the KPSS tests statistics computed for these cases for each lag index, as indicated in the parentheses. The test statistics shown in Fig. 6.4a have been computed using the Grunfeld data on investment expenditures by General Electric, IGE. These data are annual frequency observations. Among any other possible problems that might be associated with the use of these observations, they are too few in number to be considered for serious investigation in the present context and are simply provided as computational benchmarks. In contrast, the test statistics shown in 6.4b are based upon the same quarterly GDP series used earlier to compute the second set of Phillips–Perron test statistics shown in Fig. 6.2. As before, they are first transformed into natural logs. This data series was originally used by Cecchetti and Rich (2001) and has been chosen, in part, because of its easy availability to anyone interested. As mentioned earlier, the untransformed set of GDP observations can themselves be found, in hard copy form, in the Appendix to the present volume. They are also to be found, in machine-readable form, in the context of an Excel spreadsheet file, at http://pages.stern.nyu.edu/∼wgreene/Text/econometricanalysis.htm. This file is identified there as containing the data displayed in Appendix Table F5.1 of the Greene textbook. Incidentally, on page 755 of the 2008 edition of Greene's textbook (2008), the values shown for KPSS(10) in Fig. 6.4b are also shown.

The Elliott–Rothenberg–Stock test represents an attempt to improve on the Dickey–Fuller type test when an unknown mean or trend is present. It is described by these authors as the DF-GLS procedure, designed to accommodate more general statements of time series processes, including possibly ARMA. The computation procedure involves two steps (Elliott et al., 1996, p. 824–826, Stock & Watson, 2006, p. 650–653). The first is to estimate the intercept and trend of the standard Augmented Dickey-Fuller regression:

$$y_t = \alpha + \beta_t + \lambda y_{t-1} + \sum_{i=1}^{k} \gamma_i \Delta y_{t-i} + u_t$$

a) Variable: IGE

```
 Unit Root Tests: Summary Results
  Both Drift and Trend
  Base Sample Period:1936-1954         Frequency: Annual
  Number of Observations =  19    Dependent Mean =     8.2368

              (λ-1)       Tau       Zstat       KPSS          AIC         SBC
  DF         -0.6005    -2.5531    -11.4101                  6.6815      6.8306
  ADF(1)     -1.0428    -4.4887    -59.3110     0.0425       6.2906      6.4885
  ADF(2)     -1.5922    -4.7783     65.8894     0.0474       6.0666      6.3116
  ADF(3)     -1.9466    -3.2674     27.4977     0.0725       6.0557      6.3454
  ADF(4)     -2.4549    -2.6548     15.5951     0.1389       6.2306      6.5610
  ADF(5)     -3.8143    -3.2295      7.3977     0.2273       6.0219      6.3871

 Unit Root Tests: Summary Results
  No Trend, Drift Only
  Base Sample Period:1936-1954         Frequency: Annual
  Number of Observations =  19    Dependent Mean =     8.2368

              (λ-1)       Tau       Zstat       KPSS          AIC         SBC
  DF         -0.1714    -1.1252     -3.2568                  6.8499      6.9493
  ADF(1)     -0.2730    -1.5118     -7.0756     0.8389       6.9344      7.0828
  ADF(2)     -0.1294    -0.5769     -2.0702     0.6675       7.0237      7.2197
  ADF(3)     -0.1023    -0.4939     -1.0146     0.5873       6.6366      6.8780
  ADF(4)     -0.0877    -0.3690     -0.5389     0.5348       6.7143      6.9975
  ADF(5)     -0.0135    -0.0447     -0.0627     0.4882       6.8997      7.2192
```

b) Variable: Ln(GDP)

```
 Unit Root Tests: Summary Results
  Both Drift and Trend
  Base Sample Period:195002-200004     Frequency: Quarterly
  Number of Observations = 203    Dependent Mean =     0.0086

              (λ-1)       Tau       Zstat       KPSS          AIC         SBC
  DF         -0.0463    -2.6140     -9.3900                 -9.2329     -9.1839
  ADF(1)     -0.0483    -2.8917    -15.2627     1.4066      -9.3757     -9.3102
  ADF(2)     -0.0460    -2.7176    -15.1639     0.9591      -9.3871     -9.3050
  ADF(3)     -0.0453    -2.6169    -14.4251     0.7359      -9.3747     -9.2757
  ADF(4)     -0.0442    -2.5045    -13.3646     0.6025      -9.3638     -9.2479
  ADF(5)     -0.0396    -2.2059    -10.5549     0.5138      -9.3596     -9.2268
  ADF(6)     -0.0390    -2.1422    -10.6449     0.4505      -9.3549     -9.2049
  ADF(7)     -0.0402    -2.1800    -11.2003     0.4029      -9.3450     -9.1778
  ADF(8)     -0.0386    -2.0556    -10.1360     0.3658      -9.3308     -9.1461
  ADF(9)     -0.0435    -2.2937    -13.7473     0.3360      -9.3309     -9.1287
  ADF(10)    -0.0453    -2.3464    -15.4210     0.3116      -9.3172     -9.0974

 Unit Root Tests: Summary Results
  No Trend, Drift Only
  Base Sample Period:195002-200004     Frequency: Quarterly
  Number of Observations = 203    Dependent Mean =     0.0086

              (λ-1)       Tau       Zstat       KPSS          AIC         SBC
  DF         -0.0021    -1.4582     -0.4266                 -9.2119     -9.1793
  ADF(1)     -0.0012    -0.8505     -0.3573    10.2097      -9.3459     -9.2968
  ADF(2)     -0.0007    -0.5303     -0.2248     6.8501      -9.3610     -9.2953
  ADF(3)     -0.0008    -0.5956     -0.2391     5.1679      -9.3510     -9.2685
  ADF(4)     -0.0009    -0.6724     -0.2526     4.1579      -9.3428     -9.2435
  ADF(5)     -0.0009    -0.6584     -0.2191     3.4844      -9.3455     -9.2292
  ADF(6)     -0.0006    -0.4570     -0.1533     3.0031      -9.3416     -9.2083
  ADF(7)     -0.0008    -0.5724     -0.1918     2.6422      -9.3308     -9.1803
  ADF(8)     -0.0008    -0.5617     -0.1783     2.3614      -9.3191     -9.1513
  ADF(9)     -0.0008    -0.5308     -0.1882     2.1368      -9.3135     -9.1282
  ADF(10)    -0.0007    -0.4817     -0.1757     1.9531      -9.2982     -9.0953
```

Fig. 6.4 **(a)** KPSS test statistic computed using Grunfeld data on IGE. **(b)** KPSS test statistic computed using Ln(GDP) by computing three variables V_t, X_{1t}, and X_{2t}, where:

$$\begin{aligned}
V_t &= y_t - a^* y_{t-1},\ t = 2, 3, \ldots, T \\
V_1 &= y_1 \\
X_{1t} &= 1 - a^*,\ t = 2, 3, \ldots, T \\
X_{11} &= 1 \\
X_{2t} &= t - a^*(1 - t),\ t = 2, 3, \ldots, T \\
X_{21} &= 1
\end{aligned}$$

and, $a^* = 1–13.5/T$, then performing the regression:

$$V_t = \delta_1 X_{1t} + \delta_2 X_{2t} + \varepsilon_t$$

The estimated values of the δ_i, d_1 and d_2, are then used to compute a "detrended" version of y_t:

$$y_t^* = y_t - (d_1 + d_2 t)$$

for all $t = 1, 2, \ldots .T$.

The second step is to perform the intercept and time trend free regression:

$$\Delta y_t^* = \lambda y_{t-1}^* + \sum_{i=1}^{k} \gamma_i \Delta y_{t-i}^* + u_t$$

for a user (or software developer) selected number of lags k.

In the case of no time trend, three modifications are made. First a* is computed as $a^* = 1–7/T$, second X_{2t} is omitted from the calculations, and finally the series is computed as:

$$y_t^* = y_t - d_1$$

The initial GLS regression obviously complicates the ERS test, but arguably "improves its ability to discriminate between the null hypothesis of a unit autoregressive root and the alternative that y_t is stationary. This improvement can be substantial. For example, suppose that $[y_t]$ is stationary AR(1) with autoregressive coefficient $[\lambda] = 0.95$ and there are $T = 200$ observations and that the unit root tests are computed without a time trend... Then the probability that the ADF test correctly rejects the null hypothesis at the 5% significance level is approximately 31% compared to 75% for the DF-GLS test" (Stock & Watson, 2006, p. 652).

Figure 6.5a, b display the computations of the ERS DF-GLS test using, respectively, the Theil–Grunfeld data on IGE and the Cecchetti–Rich–Greene data on the natural log of GDP. As before, the numbers in the parentheses identify the lag associated with each of the test values.

a) Variable: IGE

Unit Root Tests: Summary Results
Elliott-Rothenberg-Stock Method
Both Drift and Trend
Base Sample Period:1936-1954 Frequency: Annual
Number of Observations = 19 Dependent Mean = 8.2368

	(λ-1)	Tau	Zstat	AIC	SBC
ADF(1)	-1.0149	-4.7353	-54.1711	6.0898	6.1887
ADF(2)	-1.4913	-4.9797	85.0583	5.9124	6.0595
ADF(3)	-1.5042	-2.9605	55.1896	5.9912	6.1844
ADF(4)	-1.6406	-2.1603	29.3682	6.2130	6.4490

Unit Root Tests: Summary Results
Elliott-Rothenberg-Stock Method
No Trend, Drift Only
Base Sample Period:1936-1954 Frequency: Annual
Number of Observations = 19 Dependent Mean = 1.1427

	(λ-1)	Tau	Zstat	AIC	SBC
ADF(1)	-0.2650	-1.4679	-7.5056	6.8889	6.9878
ADF(2)	-0.1193	-0.5301	-2.0982	6.9883	7.1353
ADF(3)	-0.0987	-0.4159	-1.2006	6.8713	7.0644
ADF(4)	-0.0097	-0.0343	-0.0947	7.0528	7.2888

b) Variable: LN(GDP)

Unit Root Tests: Summary Results
Elliott-Rothenberg-Stock Method
Both Drift and Trend
Base Sample Period:195003-200004 Frequency: Quarterly
Number of Observations = 202 Dependent Mean = 0.0085

	(λ-1)	Tau	Zstat	AIC	SBC
ADF(1)	-0.0265	-1.9894	-8.0599	-9.3960	-9.3631
ADF(2)	-0.0285	-2.1139	-9.3288	-9.3863	-9.3369
ADF(3)	-0.0271	-1.9801	-8.4711	-9.3734	-9.3072
ADF(4)	-0.0251	-1.8154	-7.1424	-9.3676	-9.2846
ADF(5)	-0.0225	-1.6146	-5.5678	-9.3712	-9.2712
ADF(6)	-0.0239	-1.7000	-6.4682	-9.3625	-9.2455
ADF(7)	-0.0234	-1.6422	-6.0654	-9.3486	-9.2143
ADF(8)	-0.0225	-1.5582	-5.6915	-9.3361	-9.1845
ADF(9)	-0.0252	-1.7355	-7.2443	-9.3322	-9.1631
ADF(10)	-0.0265	-1.8292	-7.2844	-9.3581	-9.1715

Unit Root Tests: Summary Results
Elliott-Rothenberg-Stock Method
No Trend, Drift Only
Base Sample Period:195003-200004 Frequency: Quarterly
Number of Observations = 202 Dependent Mean = -0.0001

	(λ-1)	Tau	Zstat	AIC	SBC
ADF(1)	0.0041	4.2732	1.5559	-9.2692	-9.2363
ADF(2)	0.0035	3.4761	1.5343	-9.2765	-9.2270
ADF(3)	0.0034	3.2455	1.5242	-9.2618	-9.1956
ADF(4)	0.0035	3.2336	1.5240	-9.2521	-9.1690
ADF(5)	0.0037	3.3355	1.5254	-9.2448	-9.1448
ADF(6)	0.0031	2.6919	1.5101	-9.2559	-9.1388
ADF(7)	0.0030	2.5502	1.5030	-9.2441	-9.1098
ADF(8)	0.0028	2.2800	1.4877	-9.2330	-9.0814
ADF(9)	0.0024	1.9600	1.4835	-9.2471	-9.0780
ADF(10)	0.0027	2.1943	1.4938	-9.2751	-9.0885

Fig. 6.5 **(a)** ERS test statistics computed using Grunfeld data on IGE. **(b)** ERS test statistics computed using Ln(GDP)

Evaluating Cointegration

As discussed earlier, the idea of cointegration is that variables that are regarded as being *cointegrated* more or less by definition must have a long-term equilibrium relationship in the sense of Granger's comments quoted earlier. Of course, in order to give economic content to this idea there is still the additional need to explain why two or more particular economic variables are likely to behave in a related way. However, in the simpliest case, a set of two or more economic variables, $\mathbf{y}_{\mathrm{i}}$, $\mathrm{i} = 1, 2, \ldots$ may be said to be cointegrated if there exists a cointegrating vector $\mathbf{a}$ such that:

$$\mathbf{Ya} = 0$$

where the columns of $\mathbf{Y}$ are the $\mathbf{y}_{\mathrm{i}}$. Notice that, when such a vector exists, it will not be unique, inasmuch as multiplication by any non-zero scalar leaves the posited relationship unaffected. Furthermore, this particular expression rules out the alternative possibility that the variables are stably related, in a systematic way, rather than static. Therefore, an obvious generalization is to specify that:

$$\mathbf{Ya} = \mathbf{Z}\gamma$$

where $\mathbf{Z}$ might contain one or more trend terms and a constant term, or even one or more exogenous variables, and where γ is a vector of parameters. That the relationships are not exact can in turn be expressed by:

$$\mathbf{Ya} = \mathbf{Z}\gamma + \mathbf{u}$$

where $\mathbf{u}$ is an unobserved vector of stochastic values. Then, simply normalizing the vector $\mathbf{a}$, so that the parameter on the first of its elements takes the value 1, allows us to restate this expression as:

$$\mathbf{y}_1 = \mathbf{Y}^*\mathbf{a}^* + \mathbf{Z}\gamma + \mathbf{u}$$

but exactly how many terms to include in this cointegrating regression has been subject to debate (Hansen, 1995).

Engle and Granger (1987) of course proposed this type of regression formulation so as to estimate a cointegrating vector, choosing the perform the regression using OLS. At first sight such an approach would appear to involve two problems in particular. First, that in the case that the variables are cointegrated, thus also jointly determined, the regression might be spurious. Second, in this instance the error term, $\mathbf{u}$, will not be distributed independently of at least some of the regressor variables. However, in this case, the very fact that the $\mathbf{y}_{\mathrm{i}}$ are cointegrated implies both that the OLS estimates are super-consistent and that the correlation between the error term and the regressor $\mathbf{y}_{\mathrm{i}}$ will be asymptotically negligible (Stock, 1987) (although there are circumstances that can create finite sample bias and inefficiency (Davidson & MacKinnon, 1993, p. 719)). The upshot is the ability not only to perform the regression justifiably, but to test for the cointegration of the vectors by performing the unit root regression:

$$\Delta u_t = (\lambda - 1)\, u_{t-1} + \varepsilon_t$$

on the residuals of this regression. Of course, as considered earlier, if serial correlation of the errors is likely, an improvement on this methodology is to include further lag terms and/or to use non-parametric tests that accommodate serial correlation. If the $\mathbf{y}_i$ are *not* cointegrated the u_t should have a unit root. For instance, see the Phillips and Hansen (1990) and the Saikkonen (1991) papers for extensions apparently both represented only in the case of one package, as indicated in Table 6.2.

This description of the Engle–Granger approach is no more than a bare outline. The aim is not to examine the various particular issues associated with testing for cointegrating vectors among a set of economic variables, so much as to demonstrate the connection between the unit root tests considered earlier and this testing. Table 6.2 is included in order to demonstrate the degree to which econometric software packages make the connection. The caveats are essentially the same in

Table 6.2 Cointegration procedures

	Johansen ML	Engle Granger	Phillips Hansen	Stock Watson Saikkonen
Independent software packages				
AREMOS				
AutoBox				
B34S	(X)	(X)		
Betahat	(X)	(X)		
EasyReg	(X)			
EViews	(X)			
FP				
gretl	(X)	(X)		
LIMDEP				
MicroFit				
Modeleasy+				
MODLER		(X)		
NLOGIT				
PcGive	(X)	(X)		
RATS	(X)	(X)		(X)
REG-X	(X)			
SHAZAM				
SORITEC				
STATA	(X)			
TROLL				
TSP	(X)	(X)		
Wysea	(X)			
Econometric programming language applications				
GaussX	(X)	(X)		
TSM 4	(X)	(X)	(X)	(X)

each case, although there are certain specific issues, such as the need to take into account the number of variables included in the cointegrating regression when considering the distributions of the test statistics.

A well-known alternative to the types of tests just considered is that associated with the name of Johansen (Hall, 1989; Hansen & Johansen, 1998; Johansen, 1995), which is based upon the use of maximum likelihood as an estimation method and vector autoregression (Johansen, 1988; Johansen & Juselius, 1990; Johansen, 1991; Johansen & Juselius, 1992). The argument for the Johansen ML approach is that residual-based cointegration tests of the type just briefly considered can be inefficient and, as the number of variables in the cointegrating equation increase, can produce contradictory results. Instead the Johansen approach and its extensions attempt the full system estimation of autoregressive models possibly consisting of cointegrated variables. As indicated in Table 6.2, a number of the existing econometric software packages implement this testing method. However, the simple single equation model that has provided the data for the present study obviously cannot be used in order to determine the degree to which these packages are consistent in their implementation of the approach.

Encompassing and Non-Nested Specifications Tests

Even the cointegration tests just considered represent tests of the properties of a particular candidate specification, rather than whether or not there is an alternative specification that might be more appropriate. In 1972, this problem would have been described as the problem of non-nested specifications (Dhrymes et al., 1972). Today, it is more common to speak of it as the "encompassing" problem although the term "non-nested" continues to be used (Hendry & Richards, 1989; Kiviet & Phillips, 1986; McAleer & Pesaran, 1986). In this context, it is generally addressed by beginning from an agnostic position: the actual data generating process is not known, but a particular specification has been put forward as a candidate. However, it may not be the only candidate. More formally, consider the two specifications:

$$\mathbf{y} = \mathbf{X}\beta_1 + \mathbf{u}_1$$
$$\mathbf{y} = \mathbf{Z}\beta_2 + \mathbf{u}_2$$

Where X and Z obviously represent alternative sets of regressors, β_1 and β_2 alternative parameter vectors and u_1 and u_2 alternative T x 1 disturbance vectors. The observation matrices, X and Z, respectively, are dimensioned T x K_1 and T x K_2 and β_1 and β_2, respectively, $K_1 \times 1$ and $K_2 \times 1$ vectors of constants.

One way to state the nature of the situation is to say that these two specifications are non-nested (or non-encompassing) if neither set of regressors can be expressed as an exact linear combination of the other. Alternatively, given the specification alternatives M1 and M2, Hendry and Doornik (1999, p. 217) pose the "encompassing question" as "whether or not M1 mimics the DGP by correctly predicting the results of the misspecified M2: that provides a basis for testing if M1 adequately represents the DGP. Failure to do so is a rejection of the adequacy of M1; success suggests that M2 is inferentially redundant." Greene expresses this idea in terms of whether "a maintained model can explain the features of its competitors" (Greene, 2003, p. 153), whereas Hoover and Perez (1999, p. 168), in the context of their formalization of the LSE method, suggest that "one model encompasses another if it conveys all the information conveyed by another model," where both models are restrictions of a (third) joint model that in unrestricted form subsumes both models.

It appears from Table 6.3 that only three tests are offered in common by at least two econometric software packages, namely the so-called J-Test, originally proposed by Davidson and MacKinnon (1981), the Cox test, and the "Encompassing Test." Consider the compound model:

$$\mathbf{y} = \mathbf{X}\beta_1 + \mathbf{Z}\beta_2 + \mathbf{u}$$

Table 6.3 Non-nested specification tests

	N-Test	NT-Test	W-Test	J-Test	JA-Test	Encompassing Test	Cox Test
Independent software packages							
AREMOS							
AutoBox							
B34S							
Betahat							
EasyReg							
EViews							
FP							
gretl							
LIMDEP				(X)			(X)
MicroFit	(X)	(X)	(X)	(X)	(X)	(X)	
Modeleasy+							
MODLER							
NLOGIT							
PcGive						(X)	(X)
RATS							
REG-X							
SHAZAM							
SORITEC							
STATA							
TROLL							
TSP							
Wysea							
Econometric programming language applications							
GaussX				(X)			
TSM 4							

() implies that the statistic can be produced at the option of the user, *ML* implies that the statistic is produced automatically, but only when technique performed is Maximum Likelihood and then its alternative formulations:

$$\begin{aligned}\mathbf{y} &= \mathbf{X}\beta_1^* + \mathbf{Z}\beta_2^* + \mathbf{C}\gamma + \mathbf{u} \\ &= \mathbf{X}\beta_1 + \mathbf{Z}\beta_2^* + \mathbf{u}\end{aligned}$$

where $\mathbf{X}\beta_1^*$ selects the variables unique to the first specification and $\mathbf{Z}\beta_2^*$ those unique to the second, whereas $\mathbf{C}$ consists of observations on the variables common to both specifications. In the context of the final formulation, a test of $\beta_2^* = \mathbf{0}$ constitutes a test that the first of the original specifications "encompasses" the second; as Greene independently points out (p. 154), Davidson and MacKinnon (1993, pp. 384–387) demonstrate that this can be achieved by a standard F-test.

In contrast, consider:

$$\mathbf{y} = (1 - \lambda)\, \mathbf{X}\beta_1 + \lambda \mathbf{Z}\beta_2^* + \mathbf{u}$$

and the fact that $\lambda = 0$, if properly formulated as a test, implies a test *against* the second specification hypothesis. The J-Test can be formulated by first regressing $\mathbf{y}$ on $\mathbf{Z}\beta_2^*$, yielding the OLS estimate $\mathbf{b}_2^*$, then setting up a second estimating equation in the form:

$$\mathbf{y} = \mathbf{X}\beta_1 + \hat{\mathbf{s}}\lambda + \mathbf{v}$$

where:

$$\hat{\mathbf{s}} = \mathbf{Z}\mathbf{b}_2^*$$

using a standard t-test to test for $\lambda = 0$.

These two tests actually provide a useful software design case study. Although it is obviously possible to be more elaborate in both motivating and describing them, they have been introduced here in the specific way they have in order to show that although few econometric software programs automatically offer them as automated, focused procedures, they would not be difficult to perform using almost any of the existing packages. Conducting the tests is hardly the problem; the problem is that of formulating realistic alternative specifications to then test in order to provide a useful benchmark. As in the case of many of the other tests that have been considered previously in this volume, the logic that dictates including these automatically in an econometric software package is the benefit of their more common application, without the user needing to experience the tedium of formulating them one by one – not to mention any possible issues associated with needing to know how to construct them properly. In contrast, the tests being considered here instead necessarily place the analytical burden on the user to formulate appropriate specifications to test, and require only that a program be designed so as to make accessible to that user either generated results, such as $\mathbf{s}$, or slightly generalized test facilities, such as the ability to quickly produce a joint F-Test for a particular parameter estimate subset.

Chapter Appendix

This appendix displays the underlying regression equations that correspond to the summary displays in Figs. 6.1 and 6.2. These results are provided in order to provide ample documentation of the background calculations.

Grunfeld Data

```
Variable: IGE

 Dickey-Fuller Unit Root Test
  Both Drift and Trend
  Frequency: Annual
  Sample Period:     1936-1954
  Number of Observations = 19

DEL(x)=B1*TREND+B2*x(-1)+B3

       Parameter Tau-Statistic  Std Error    Mean
B1)      4.24233     2.24392      1.89059   11.000
B2)     -0.60053    -2.55312      0.23522   97.695
B3)     20.24021     1.36519     14.82598    1.000

Variance=      690.611602  Dependent Mean=         8.237
Standard Error= 26.279490  Percent Error=        319.05%
R-Square=          0.2921  R-Bar-Square=          0.2036
Schwarz Criterion= 6.8306  Akaike Criterion=      6.6815
Max LLF=         -87.4342  F-Statistic:  F(2, 16)=3.3009

Augmented Dickey-Fuller Unit Root Test
  Both Drift and Trend
  Frequency: Annual
  Sample Period:    1937-1954
  Number of Observations = 18

DEL(x)=B1*TREND+B2*x(-1)+B3*DEL(x[-1])+B4

     Parameter  Tau-Statistic   Std Error      Mean
B1)    6.91180     3.97268       1.73983    11.500
B2)   -1.04283    -4.48869       0.23233   101.283
B3)    0.68352     3.28922       0.20780     8.133
B4)   28.61021     2.12676      13.45245     1.000

Variance=      444.747155  Dependent Mean=          8.033
Standard Error= 21.089029  Percent Error=         262.52%
R-Square=          0.6007  R-Bar-Square=           0.5152
```

```
Schwarz Criterion= 6.4885  Akaike Criterion=       6.2906
Max LLF=         -78.1566  F-Statistic:  F(3, 14)= 7.0216

Augmented Dickey-Fuller Unit Root Test
  Both Drift and Trend
  Frequency: Annual
  Sample Period:    1938-1954
  Number of Observations = 17

DEL(x)=B1*TREND+B2*x(-1)+B3*DEL(x[-1])+B4*DEL(x[-2])+B5

       Parameter  Tau-Statistic Std Error     Mean
B1)     10.75233     4.81148      2.23472   12.000
B2)     -1.59222    -4.77827      0.33322  104.594
B3)      0.88320     4.22401      0.20909    7.912
B4)      0.52760     2.14586      0.24587    7.306
B5)     33.27865     2.34441     14.19488    1.000

Variance=      339.222483 Dependent Mean=         6.612
Standard Error= 18.417993 Percent Error=        278.56%
R-Square=          0.7282 R-Bar-Square=         0.6376
Schwarz Criterion= 6.3116 Akaike Criterion=     6.0666
Max LLF=         -70.6879 F-Statistic:  F(4, 12)=8.0374

Augmented Dickey-Fuller Unit Root Test
  Both Drift and Trend
  Frequency: Annual
  Sample Period:    1939-1954
  Number of Observations = 16

DEL(x)=B1*TREND+B2*x(-1)+B3*DEL(x[-1])+B4*DEL(x[-2])+
       B5*DEL(x[-3])+B6

         Parameter Tau-Statistic Std Error      Mean
  B1)     12.37419    3.20258      3.86382    12.500
  B2)     -1.94661   -3.26738      0.59577   106.306
  B3)      1.18935    3.06321      0.38827     6.394
  B4)      0.68892    2.04488      0.33690     7.019
  B5)      0.25440    0.87208      0.29172     6.381
  B6)     47.25864    2.79231     16.92457     1.000

  Variance=      322.360245  Dependent Mean=         9.063
  Standard Error= 17.954393  Percent Error=        198.12%
  R-Square=          0.7584  R-Bar-Square=         0.6376
  Schwarz Criterion= 6.3454  Akaike Criterion=     6.0557
  Max LLF=         -65.1483  F-Statistic:  F(5, 10)=6.2782

Augmented Dickey-Fuller Unit Root Test
  Both Drift and Trend
```

```
  Frequency: Annual
  Sample Period:    1940-1954
  Number of Observations = 15

DEL(x)=B1*TREND+B2*x(-1)+B3*DEL(x[-1])+B4*DEL(x[-2])+B5*
       DEL(x[-3])+B6*DEL(x[-4])+B7

        Parameter  Tau-Statistic  Std Error     Mean
  B1)    15.50022     2.61289       5.93221   13.000
  B2)    -2.45493    -2.65480       0.92471  110.420
  B3)     1.60789     2.23117       0.72065    8.993
  B4)     1.04040     1.78514       0.58281    5.340
  B5)     0.47420     1.09600       0.43266    6.013
  B6)     0.23877     0.69168       0.34521    4.027
  B7)    55.17530     2.48254
 22.22536     1.000

  Variance=      374.608591  Dependent Mean=        9.433
  Standard Error= 19.354808  Percent Error=       205.17%
  R-Square=          0.7748  R-Bar-Square=         0.6060
  Schwarz Criterion= 6.5610  Akaike Criterion=     6.2306
  Max LLF=         -61.0136  F-Statistic:  F(6, 8)=4.5883

Augmented Dickey-Fuller Unit Root Test
  Both Drift and Trend
  Frequency: Annual
  Sample Period:    1941-1954
  Number of Observations = 14

DEL(x)=B1*TREND+B2*x(-1)+B3*DEL(x[-1])+B4*DEL(x[-2])+B5*
       DEL(x[-3])+B6*DEL(x[-4])+B7*DEL(x[-5])+B8

        Parameter  Tau-Statistic Std Error     Mean
  B1)    25.17896    3.26344       7.71548   13.500
  B2)    -3.81425   -3.22950       1.18107  114.871
  B3)     2.93237    2.87382       1.02038    9.386
  B4)     2.33909    2.62745       0.89025    8.050
  B5)     1.33351    2.02717       0.65782    4.143
  B6)     0.93406    1.89777       0.49219    3.464
  B7)     0.67941    1.87722       0.36192    4.657
  B8)    48.18458    2.01640      23.89633    1.000

  Variance=      306.842895  Dependent Mean=        8.229
  Standard Error= 17.516932  Percent Error=       212.88%
  R-Square=          0.8584  R-Bar-Square=         0.6933
  Schwarz Criterion= 6.3871  Akaike Criterion=     6.0219
  Max LLF=         -54.0184  F-Statistic:  F(7, 6)=5.1976
```

```
Augmented Dickey-Fuller Unit Root Test
  Both Drift and Trend
  Frequency: Annual
  Sample Period: 1942-1954
  Number of Observations = 13

DEL(x)=B1*TREND+B2*x(-1)+B3*DEL(x[-1])+B4*DEL(x[-2])+B5*
       DEL(x[-3])+B6*DEL(x[-4])+B7*DEL(x[-5])+B8*
       DEL(x[-6])+B9

         Parameter  Tau-Statistic Std Error     Mean
  B1)     38.76552      4.17991     9.27425   14.000
  B2)     -5.73669     -4.07964     1.40618  117.985
  B3)      4.40682      3.69316     1.19324    8.085
  B4)      3.92348      3.45191     1.13661    8.400
  B5)      2.83379      3.01641     0.93946    6.969
  B6)      1.66252      2.65037     0.62728    1.254
  B7)      1.34137      2.95923     0.45328    4.100
  B8)      0.64092      1.93877     0.33058    8.708
  B9)     38.51743      2.09085    18.42188    1.000

  Variance=      158.509907  Dependent Mean=        5.892
  Standard Error= 12.590072  Percent Error=       213.67%
  R-Square=          0.9472  R-Bar-Square=         0.8416
  Schwarz Criterion= 5.6629  Akaike Criterion=     5.2718
  Max LLF=         -43.7128  F-Statistic:  F(8, 4)=8.9723

Dickey-Fuller Unit Root Test
  No Trend, Drift Only
  Frequency: Annual
  Sample Period:    1936-1954
  Number of Observations = 19

DEL(x)=B1*x(-1)+B2

         Parameter Tau-Statistic Std Error     Mean
  B1)     -0.17141  -1.12521       0.15233   97.695
  B2)     24.98259   1.53046      16.32353    1.000

  Variance=      854.537988  Dependent Mean=         8.237
  Standard Error= 29.232482  Percent Error=        354.90%
  R-Square=          0.0693  R-Bar-Square=          0.0146
  Schwarz Criterion= 6.9493  Akaike Criterion=      6.8499
  Max LLF=         -90.0335  F-Statistic:  F(1, 17)=1.2661

Augmented Dickey-Fuller Unit Root Test
  No Trend, Drift Only
  Frequency: Annual
```

```
  Sample Period:    1937-1954
  Number of Observations = 18

DEL(x)=B1*x(-1)+B2*DEL(x[-1])+B3

         Parameter Tau-Statistic Std Error     Mean
  B1)     -0.27303  -1.51185       0.18059  101.283
  B2)      0.30543   1.17339       0.26030    8.133
  B3)     33.20237   1.75810      18.88534    1.000

  Variance=      883.035689  Dependent Mean=         8.033
  Standard Error= 29.715916  Percent Error=        369.91%
  R-Square=          0.1507  R-Bar-Square=         0.0374
  Schwarz Criterion= 7.0828  Akaike Criterion=     6.9344
  Max LLF=         -84.9503  F-Statistic:  F(2, 15)=1.3303

Augmented Dickey-Fuller Unit Root Test
  No Trend, Drift Only
  Frequency: Annual
  Sample Period:    1938-1954
  Number of Observations = 17

DEL(x)=B1*x(-1)+B2*DEL(x[-1])+B3*DEL(x[-2])+B4

         Parameter Tau-Statistic Std Error     Mean
  B1)     -0.12937   -0.57688      0.22426  104.594
  B2)      0.26422    0.97484      0.27104    7.912
  B3)     -0.32656   -1.16747      0.27972    7.306
  B4)     20.43851    0.89153     22.92509    1.000

  Variance=      917.214098  Dependent Mean=         6.612
  Standard Error= 30.285543  Percent Error=        458.06%
  R-Square=          0.2038  R-Bar-Square=         0.0201
  Schwarz Criterion= 7.2197  Akaike Criterion=     7.0237
  Max LLF=          79.8231  F-Statistic:  F(3, 13)=1.1094

Augmented Dickey-Fuller Unit Root Test
  No Trend, Drift Only
  Frequency: Annual
  Sample Period:    1939-1954
  Number of Observations = 16

DEL(x)=B1*x(-1)+B2*DEL(x[-1])+B3*DEL(x[-2])+B4*
       DEL(x[-3])+B5

         Parameter Tau-Statistic Std Error     Mean
  B1)     -0.10233  -0.49389       0.20720  106.306
  B2)      0.12517   0.45927       0.27255    6.394
  B3)     -0.24566  -1.07531       0.22845    7.019
```

```
B4)     -0.49337  -2.07907       0.23731    6.381
B5)     24.01354   1.15741      20.74772    1.000

Variance=      593.626882  Dependent Mean=         9.063
Standard Error= 24.364459  Percent Error=        268.85%
R-Square=          0.5106  R-Bar-Square=          0.3326
Schwarz Criterion= 6.8780  Akaike Criterion=      6.6366
Max LLF=         -70.7955  F-Statistic:  F(4, 11)=2.8692
```

```
Augmented Dickey-Fuller Unit Root Test
  No Trend, Drift Only
  Frequency: Annual
  Sample Period:    1940-1954
  Number of Observations = 15

DEL(x)=B1*x(-1)+B2*DEL(x[-1])+B3*DEL(x[-2])+B4*
       DEL(x[-3])+B5*DEL(x[-4])+B6

        Parameter Tau-Statistic Std Error      Mean
  B1)    -0.08775    -0.36899     0.23780   110.420
  B2)    -0.13563    -0.38823     0.34935     8.993
  B3)    -0.37102    -1.32109     0.28085     5.340
  B4)    -0.54044    -2.20686     0.24489     6.013
  B5)    -0.39508    -1.25327     0.31524     4.027
  B6)    27.16388     1.08703    24.98906     1.000

  Variance=      617.155422  Dependent Mean=       9.433
  Standard Error= 24.842613  Percent Error=      263.35%
  R-Square=          0.5827  R-Bar-Square=        0.3508
  Schwarz Criterion= 6.9975  Akaike Criterion=    6.7143
  Max LLF=         -65.6413  F-Statistic:  F(5, 9)=2.5133
```

```
Augmented Dickey-Fuller Unit Root Test
  No Trend, Drift Only
  Frequency: Annual
  Sample Period:    1941-1954
  Number of Observations = 14

DEL(x)=B1*x(-1)+B2*DEL(x[-1])+B3*DEL(x[-2])+B4*DEL(x[-3])+
       B5*DEL(x[-4])+B6*DEL(x[-5])+B7

         Parameter Tau-Statistic Std Error     Mean
  B1)     -0.01352    -0.04466     0.30271  114.871
  B2)     -0.27429    -0.64658     0.42421    9.386
  B3)     -0.44615    -1.14230     0.39057    8.050
  B4)     -0.69348    -2.07554     0.33412    4.143
  B5)     -0.45061    -1.17131     0.38470    3.464
  B6)     -0.15287    -0.38598     0.39605    4.657
  B7)     21.09340     0.61036    34.55910    1.000
```

```
  Variance=        729.848190  Dependent Mean=       8.229
  Standard Error= 27.015703  Percent Error=       328.32%
  R-Square=            0.6072  R-Bar-Square=        0.2704
  Schwarz Criterion= 7.2192  Akaike Criterion=    6.8997
  Max LLF=          -61.1630  F-Statistic:  F(6, 7)=1.8031

Augmented Dickey-Fuller Unit Root Test
  No Trend, Drift Only
  Frequency: Annual
  Sample Period:    1942-1954
  Number of Observations = 13

DEL(x)=B1*x(-1)+B2*DEL(x[-1])+B3*DEL(x[-2])+B4*DEL(x[-3])+
       B5*DEL(x[-4])+B6*DEL(x[-5])+B7*DEL(x[-6])+B8

        Parameter Tau-Statistic Std Error      Mean
  B1)     0.10600    0.33383      0.31751   117.985
  B2)    -0.48908   -1.03608      0.47204     8.085
  B3)    -0.75007   -1.77164      0.42338     8.400
  B4)    -1.00171   -2.39957      0.41746     6.969
  B5)    -0.80423   -1.82531      0.44060     1.254
  B6)    -0.36018   -0.87175      0.41317     4.100
  B7)    -0.46497   -1.13205      0.41073     8.708
  B8)    17.15602    0.46776     36.67685     1.000

  Variance=        680.693280  Dependent Mean=         5.892
  Standard Error= 26.090099  Percent Error=         442.78%
  R-Square=            0.7167  R-Bar-Square=          0.3200
  Schwarz Criterion= 7.1460  Akaike Criterion=      6.7984
  Max LLF=          -54.6356  F-Statistic:  F(7, 5)=1.8066


Dickey-Fuller Unit Root Test
 Random Walk: No Drift, No Trend
 Frequency: Annual
 Sample Period:    1936-1954
 Number of Observations = 19

DEL(x)=B1*x(-1)

         Parameter Tau-Statistic Std Error     Mean
  B1)     0.04115    0.63426       0.06488   97.695

  Variance=        918.263873  Dependent Mean=    8.237
  Standard Error= 30.302869  Percent Error=    367.89%
  R-Square=            0.0219  R-Bar-Square=     0.0219
  Schwarz Criterion= 6.9234  Akaike Criterion= 6.8737
```

```
Augmented Dickey-Fuller Unit Root Test
  Random Walk: No Drift, No Trend
  Frequency: Annual
  Sample Period:    1937-1954
  Number of Observations = 18

DEL(x)=B1*x(-1)+B2*DEL(x[-1])

         Parameter Tau-Statistic Std Error      Mean
  B1)      0.02003   0.27108       0.07389   101.283
  B2)      0.17677   0.66551       0.26562     8.133

  Variance=      998.433340  Dependent Mean=    8.033
  Standard Error= 31.597996  Percent Error=    393.34%
  R-Square=          0.0466  R-Bar-Square=    -0.0129
  Schwarz Criterion= 7.1096  Akaike Criterion= 7.0106

Augmented Dickey-Fuller Unit Root Test
  Random Walk: No Drift, No Trend
  Frequency: Annual
  Sample Period:    1938-1954
  Number of Observations = 17

DEL(x)=B1*x(-1)+B2*DEL(x[-1])+B3*DEL(x[-2])

        Parameter Tau-Statistic Std Error      Mean
  B1)     0.05882     0.78229     0.07519   104.594
  B2)     0.18319     0.72274     0.25347     7.912
  B3)    -0.42263    -1.64942     0.25623     7.306

Variance=      903.772560  Dependent Mean=    6.612
Standard Error= 30.062810  Percent Error=    454.69%
R-Square=          0.1951  R-Bar-Square=     0.0801
Schwarz Criterion= 7.1124  Akaike Criterion= 6.9654

Augmented Dickey-Fuller Unit Root Test
  Random Walk: No Drift, No Trend
  Frequency: Annual
  Sample Period:    1939-1954
  Number of Observations = 16

DEL(x)=B1*x(-1)+B2*DEL(x[-1])+B3*DEL(x[-2])+B4*DEL(x[-3])

        Parameter Tau-Statistic Std Error     Mean
  B1)     0.12517     1.88389     0.06644  106.306
  B2)    -0.02952    -0.12256     0.24086    6.394
  B3)    -0.35180    -1.65804     0.21218    7.019
  B4)    -0.57293    -2.48749     0.23032    6.381
```

```
  Variance=      610.426010  Dependent Mean=    9.063
  Standard Error= 24.706801  Percent Error=    272.63%
  R-Square=          0.5002  R-Bar-Square=     0.3753
  Schwarz Criterion= 6.8196  Akaike Criterion= 6.6265

Augmented Dickey-Fuller Unit Root Test
  Random Walk: No Drift, No Trend
  Frequency: Annual
  Sample Period:    1940-1954
  Number of Observations = 15

DEL(x)=B1*x(-1)+B2*DEL(x[-1])+B3*DEL(x[-2])+B4*DEL(x[-3])+
       B5*DEL(x[-4])

        Parameter Tau-Statistic Std Error      Mean
  B1)     0.15711     2.04229     0.07693   110.420
  B2)    -0.21439    -0.62171     0.34485     8.993
  B3)    -0.51504    -2.06117     0.24988     5.340
  B4)    -0.60369    -2.51503     0.24003     6.013
  B5)    -0.34661    -1.10072     0.31489     4.027

  Variance=      628.365188  Dependent Mean=    9.433
  Standard Error= 25.067213  Percent Error=    265.73%
  R-Square=          0.5709  R-Bar-Square=     0.3993
  Schwarz Criterion= 6.9403  Akaike Criterion= 6.7043

Augmented Dickey-Fuller Unit Root Test
  Random Walk: No Drift, No Trend
  Frequency: Annual
  Sample Period:    1941-1954
  Number of Observations = 14

DEL(x)=B1*x(-1)+B2*DEL(x[-1])+B3*DEL(x[-2])+B4*DEL(x[-3])+
       B5*DEL(x[-4])+B6*DEL(x[-5])

        Parameter Tau-Statistic Std Error      Mean
  B1)     0.16114     1.69980     0.09480   114.871
  B2)    -0.31391    -0.78002     0.40244     9.386
  B3)    -0.44994    -1.20018     0.37490     8.050
  B4)    -0.73374    -2.33349     0.31444     4.143
  B5)    -0.38109    -1.08033     0.35275     3.464
  B6)    -0.05878    -0.16784     0.35022     4.657

  Variance=      672.603996  Dependent Mean=    8.229
  Standard Error= 25.934610  Percent Error=    315.18%
  R-Square=          0.6144  R-Bar-Square=     0.3733
  Schwarz Criterion= 7.0826  Akaike Criterion= 6.8087
```

```
Augmented Dickey-Fuller Unit Root Test
  Random Walk: No Drift, No Trend
  Frequency: Annual
  Sample Period:    1942-1954
  Number of Observations = 13

DEL(x)=B1*x(-1)+B2*DEL(x[-1])+B3*DEL(x[-2])+B4*DEL(x[-3])+
       B5*DEL(x[-4])+B6*DEL(x[-5])+B7*DEL(x[-6])

        Parameter Tau-Statistic Std Error      Mean
  B1)     0.24482     2.32590     0.10526   117.985
  B2)    -0.55666    -1.32820     0.41911     8.085
  B3)    -0.76070    -1.92931     0.39428     8.400
  B4)    -1.00065    -2.57021     0.38933     6.969
  B5)    -0.77955    -1.91085     0.40796     1.254
  B6)    -0.28466    -0.80255     0.35470     4.100
  B7)    -0.41936    -1.12697     0.37211     8.708

  Variance= 592.067104 Dependent Mean= 5.892
  Standard Error= 24.332429 Percent Error= 412.95%
  R-Square= 0.7150 R-Bar-Square= 0.4299
  Schwarz Criterion= 6.9916 Akaike Criterion= 6.6874
```

Greene Data

```
Dickey-Fuller Unit Root Test
  Both Drift and Trend
  Frequency: Quarterly
  Sample Period: 195003-200004
  Number of Observations = 202

DEL(x)=B1*TREND+B2*x(-1)+B3

        Parameter Tau-Statistic  Std Error      Mean
  B1)     0.00033    2.23194       0.00015   102.500
  B2)    -0.04151   -2.32124       0.01788     8.313
  B3)     0.31982    2.39429       0.13358     1.000

  Variance=        0.000096  Dependent Mean=         0.009
  Standard Error=  0.009778  Percent Error=        114.55%
  R-Square=          0.0313  R-Bar-Square=          0.0216
  Schwarz Criterion=-9.1913  Akaike Criterion=     -9.2404
  Max LLF=         649.6551  F-Statistic:  F(2,199)=3.2179

Augmented Dickey-Fuller Unit Root Test
  Both Drift and Trend
  Frequency: Quarterly
```

```
  Sample Period: 195004-200004
  Number of Observations = 201

DEL(x)=B1*TREND+B2*x(-1)+B3*DEL(x[-1])+B4

        Parameter  Tau-Statistic Std Error      Mean
  B1)     0.00035     2.55077      0.00014   103.000
  B2)    -0.04340    -2.58846      0.01677     8.317
  B3)     0.34048     5.23719      0.06501     0.009
  B4)     0.33014     2.63645      0.12522     1.000

 Variance=        0.000082  Dependent Mean=          0.008
 Standard Error=  0.009047  Percent Error=         107.83%
 R-Square=          0.1411  R-Bar-Square=          0.1281
 Schwarz Criterion=-9.3253  Akaike Criterion=     -9.3910
 Max LLF=         662.5904  F-Statistic:  F(3,197)=10.7921

Augmented Dickey-Fuller Unit Root Test
  Both Drift and Trend
  Frequency: Quarterly
  Sample Period: 195101-200004
  Number of Observations = 200

DEL(x)=B1*TREND+B2*x(-1)+B3*DEL(x[-1])+B4*DEL(x[-2])+B5

        Parameter  Tau-Statistic Std Error      Mean
  B1)     0.00038     2.70920      0.00014   103.500
  B2)    -0.04686    -2.74425      0.01708     8.322
  B3)     0.32011     4.55324      0.07030     0.008
  B4)     0.07818     1.12463      0.06952     0.009
  B5)     0.35551     2.78882      0.12748     1.000

  Variance=        0.000082  Dependent Mean=         0.008
  Standard Error=  0.009061  Percent Error=        108.61%
  R-Square=          0.1431  R-Bar-Square=         0.1256
  Schwarz Criterion=-9.3004  Akaike Criterion=    -9.3829
  Max LLF=         659.4992  F-Statistic:  F(4,195)=8.1434
Augmented Dickey-Fuller Unit Root Test
  Both Drift and Trend
  Frequency: Quarterly
  Sample Period: 195102-200004
  Number of Observations = 199

DEL(x)=B1*TREND+B2*x(-1)+B3*DEL(x[-1])+B4*DEL(x[-2])+B5*
       DEL(x[-3])+B6

        Parameter  Tau-Statistic Std Error     Mean
  B1)     0.00038    2.60793       0.00014  104.000
  B2)    -0.04627   -2.64836       0.01747    8.326
```

```
  B3)     0.32180    4.55634       0.07063    0.008
  B4)     0.09708    1.30695       0.07428    0.008
  B5)    -0.03806   -0.54268       0.07014    0.009
  B6)     0.35135    2.69539       0.13035    1.000

  Variance=         0.000083  Dependent Mean=         0.008
  Standard Error=  0.009095  Percent Error=         109.19%
  R-Square=           0.1453  R-Bar-Square=          0.1231
  Schwarz Criterion=-9.2712  Akaike Criterion=     -9.3705
  Max LLF=          655.9915  F-Statistic:  F(5,193)=6.5600

Augmented Dickey-Fuller Unit Root Test
  Both Drift and Trend
  Frequency: Quarterly
  Sample Period: 195103-200004
  Number of Observations = 198

DEL(x)=B1*TREND+B2*x(-1)+B3*DEL(x[-1])+B4*DEL(x[-2])+B5*
       DEL(x[-3])+B6*DEL(x[-4])+B7

        Parameter  Tau-Statistic Std Error      Mean
  B1)     0.00035    2.35338       0.00015   104.500
  B2)    -0.04259   -2.38944       0.01782     8.330
  B3)     0.31746    4.47168       0.07099     0.008
  B4)     0.09975    1.33819       0.07454     0.008
  B5)    -0.03002   -0.40061       0.07493     0.008
  B6)    -0.06105   -0.86776       0.07036     0.009
  B7)     0.32425    2.43946       0.13292     1.000

  Variance=         0.000083  Dependent Mean=         0.008
  Standard Error=  0.009113  Percent Error=         109.99%
  R-Square=           0.1473  R-Bar-Square=          0.1205
  Schwarz Criterion=-9.2451  Akaike Criterion=     -9.3613
  Max LLF=          652.8220  F-Statistic:  F(6,191)=5.4986

Augmented Dickey-Fuller Unit Root Test
  Both Drift and Trend
  Frequency: Quarterly
  Sample Period: 195104-200004
  Number of Observations = 197

DEL(x)=B1*TREND+B2*x(-1)+B3*DEL(x[-1])+B4*DEL(x[-2])+B5*
       DEL(x[-3])+B6*DEL(x[-4])+B7*DEL(x[-5])+B8

        Parameter  Tau-Statistic Std Error      Mean
  B1)     0.00030    2.02947       0.00015   105.000
  B2)    -0.03722   -2.06231       0.01805     8.334
  B3)     0.30234    4.23739       0.07135     0.008
  B4)     0.09640    1.29446       0.07447     0.008
```

```
  B5)    -0.02540   -0.33946      0.07483     0.008
  B6)    -0.04815   -0.64314      0.07487     0.008
  B7)    -0.09684   -1.37081      0.07065     0.009
  B8)     0.28489    2.11784      0.13452     1.000

  Variance=       0.000083   Dependent Mean=        0.008
  Standard Error= 0.009090   Percent Error=       110.48%
  R-Square=          0.1545  R-Bar-Square=         0.1232
  Schwarz Criterion=-9.2281  Akaike Criterion=    -9.3614
  Max LLF=         650.5700  F-Statistic:  F(7,189)=4.9328

Augmented Dickey-Fuller Unit Root Test
  Both Drift and Trend
  Frequency: Quarterly
  Sample Period: 195201-200004
  Number of Observations = 196

DEL(x)=B1*TREND+B2*x(-1)+B3*DEL(x[-1])+B4*DEL(x[-2])
       +B5*DEL(x[-3])+B6*DEL(x[-4])+B7*DEL(x[-5])+B8*
       DEL(x[-6])+B9

         Parameter Tau-Statistic Std Error     Mean
  B1)      0.00033    2.19646     0.00015   105.500
  B2)     -0.04078   -2.23188     0.01827     8.339
  B3)      0.31769    4.38946     0.07237     0.008
  B4)      0.10410    1.39252     0.07476     0.008
  B5)     -0.02466   -0.32886     0.07499     0.008
  B6)     -0.05084   -0.67776     0.07502     0.008
  B7)     -0.10139   -1.34614     0.07532     0.008
  B8)      0.06584    0.92664     0.07105     0.008
  B9)      0.31085    2.28392     0.13610     1.000

  Variance=        0.000083  Dependent Mean=        0.008
  Standard Error=  0.009097  Percent Error=       110.13%
  R-Square=          0.1604  R-Bar-Square=         0.1244
  Schwarz Criterion=-9.2044  Akaike Criterion=    -9.3549
  Max LLF=         647.6671  F-Statistic:  F(8,187)=4.4643

Dickey-Fuller Unit Root Test
  No Trend, Drift Only
  Frequency: Quarterly
  Sample Period: 195003-200004
  Number of Observations = 202

DEL(x)=B1*x(-1)+B2

         Parameter Tau-Statistic  Std Error    Mean
  B1)     -0.00172    -1.19412      0.00144   8.313
  B2)      0.02287     1.90195      0.01203   1.000
```

```
  Variance=         0.000098  Dependent Mean=         0.009
  Standard Error=   0.009875  Percent Error=        115.69%
  R-Square=           0.0071  R-Bar-Square=          0.0021
  Schwarz Criterion=-9.1928   Akaike Criterion=     -9.2256
  Max LLF=          647.1579  F-Statistic:  F(1,200)=1.4259

Augmented Dickey-Fuller Unit Root Test
  No Trend, Drift Only
  Frequency: Quarterly
  Sample Period: 195004-200004
  Number of Observations = 201

DEL(x)=B1*x(-1)+B2*DEL(x[-1])+B3

        Parameter Tau-Statistic  Std Error    Mean
  B1)    -0.00077    -0.56662      0.00136   8.317
  B2)     0.32406     4.94107      0.06559   0.009
  B3)     0.01201     1.05858      0.01134   1.000

 Variance=         0.000084  Dependent Mean=          0.008
 Standard Error=   0.009172  Percent Error=         109.32%
 R-Square=           0.1128  R-Bar-Square=           0.1038
 Schwarz Criterion=-9.3192   Akaike Criterion=      -9.3685
 Max LLF=          659.3247  F-Statistic:  F(2,198)=12.5849

Augmented Dickey-Fuller Unit Root Test
  No Trend, Drift Only
  Frequency: Quarterly
  Sample Period: 195101-200004
  Number of Observations = 200

DEL(x)=B1*x(-1)+B2*DEL(x[-1])+B3*DEL(x[-2])+B4

         Parameter Tau-Statistic Std Error    Mean
  B1)     -0.00074    -0.54199      0.00137   8.322
  B2)      0.30860     4.32813      0.07130   0.008
  B3)      0.04931     0.70651      0.06980   0.009
  B4)      0.01151     1.00193      0.01149   1.000

Variance=         0.000085  Dependent Mean=          0.008
Standard Error=   0.009206  Percent Error=         110.35%
R-Square=           0.1109  R-Bar-Square=           0.0973
Schwarz Criterion=-9.2900   Akaike Criterion=      -9.3559
Max LLF=          655.8043  F-Statistic:  F(3,196)= 8.1477
```

```
Augmented Dickey-Fuller Unit Root Test
  No Trend, Drift Only
  Frequency: Quarterly
  Sample Period: 195102-200004
  Number of Observations = 199

DEL(x)=B1*x(-1)+B2*DEL(x[-1])+B3*DEL(x[-2])+B4*
       DEL(x[-3])+B5

         Parameter Tau-Statistic Std Error   Mean
  B1)    -0.00085    -0.61015      0.00139  8.326
  B2)     0.31216     4.36116      0.07158  0.008
  B3)     0.07497     1.00097      0.07489  0.008
  B4)    -0.06854    -0.97652      0.07018  0.009
  B5)     0.01272     1.09366      0.01163  1.000

  Variance=         0.000085  Dependent Mean=         0.008
  Standard Error=  0.009230  Percent Error=         110.81%
  R-Square=           0.1151  R-Bar-Square=          0.0969
  Schwarz Criterion=-9.2631  Akaike Criterion=     -9.3459
  Max LLF=          652.5455  F-Statistic:  F(4,194)=6.3110

Augmented Dickey-Fuller Unit Root Test
  No Trend, Drift Only
  Frequency: Quarterly
  Sample Period: 195103-200004
  Number of Observations = 198

DEL(x)=B1*x(-1)+B2*DEL(x[-1])+B3*DEL(x[-2])+B4*DEL(x[-3])+
       B5*DEL(x[-4])+B6

         Parameter Tau-Statistic Std Error    Mean
  B1)     -0.00077    -0.55040     0.00140   8.330
  B2)      0.30703     4.28283     0.07169   0.008
  B3)      0.08127     1.08363     0.07500   0.008
  B4)     -0.05298    -0.70486     0.07516   0.008
  B5)     -0.08684    -1.23508     0.07031   0.009
  B6)      0.01263     1.07727     0.01173   1.000

  Variance=         0.000085  Dependent Mean=         0.008
  Standard Error=  0.009220  Percent Error=         111.28%
  R-Square=           0.1226  R-Bar-Square=          0.0997
  Schwarz Criterion=-9.2432  Akaike Criterion=     -9.3428
  Max LLF=          649.9922  F-Statistic:  F(5,192)=5.3638

Augmented Dickey-Fuller Unit Root Test
  No Trend, Drift Only
  Frequency: Quarterly
```

```
  Sample Period: 195104-200004
  Number of Observations = 197

DEL(x)=B1*x(-1)+B2*DEL(x[-1])+B3*DEL(x[-2])+B4*DEL(x[-3])+
       B5*DEL(x[-4])+B6*DEL(x[-5])+B7

         Parameter Tau-Statistic Std Error    Mean
  B1)     -0.00070    -0.50086     0.00140   8.334
  B2)      0.29046     4.05160     0.07169   0.008
  B3)      0.07994     1.07113     0.07463   0.008
  B4)     -0.04384    -0.58545     0.07489   0.008
  B5)     -0.06659    -0.88875     0.07492   0.008
  B6)     -0.11770    -1.67022     0.07047   0.009
  B7)      0.01292     1.09677     0.01178   1.000

  Variance=        0.000084  Dependent Mean=         0.008
  Standard Error=  0.009164  Percent Error=        111.39%
  R-Square=          0.1360  R-Bar-Square=          0.1088
  Schwarz Criterion=-9.2334  Akaike Criterion=     -9.3500
  Max LLF=         648.4465  F-Statistic:  F(6,190)=4.9866

Augmented Dickey-Fuller Unit Root Test
  No Trend, Drift Only
  Frequency: Quarterly
  Sample Period: 195201-200004
  Number of Observations = 196

DEL(x)=B1*x(-1)+B2*DEL(x[-1])+B3*DEL(x[-2])+B4*DEL(x[-3])
       +B5*DEL(x[-4])+B6*DEL(x[-5])+B7*DEL(x[-6])+B8

         Parameter Tau-Statistic Std Error    Mean
  B1)     -0.00077    -0.54077     0.00142   8.339
  B2)      0.30108     4.14104     0.07271   0.008
  B3)      0.08454     1.12746     0.07498   0.008
  B4)     -0.04490    -0.59719     0.07518   0.008
  B5)     -0.06968    -0.92565     0.07528   0.008
  B6)     -0.12056    -1.59531     0.07557   0.008
  B7)      0.04575     0.64277     0.07117   0.008
  B8)      0.01303     1.09203     0.01193   1.000

  Variance=        0.000084  Dependent Mean=         0.008
  Standard Error=  0.009189  Percent Error=        111.25%
  R-Square=          0.1387  R-Bar-Square=          0.1066
  Schwarz Criterion=-9.2058  Akaike Criterion=     -9.3396
  Max LLF=         645.1709  F-Statistic:  F(7,188)=4.3249

Dickey-Fuller Unit Root Test
  Random Walk: No Drift, No Trend
  Frequency: Quarterly
```

```
  Sample Period: 195003-200004
  Number of Observations = 202

DEL(x)=B1*x(-1)

         Parameter Tau-Statistic Std Error   Mean
  B1)     0.00102    12.11741     0.00008   8.313

Variance=        0.000099  Dependent Mean=     0.009
Standard Error=  0.009939  Percent Error=     116.44%
R-Square=          0.4221  R-Bar-Square=      0.4221
Schwarz Criterion=-9.2012  Akaike Criterion= -9.2176

Augmented Dickey-Fuller Unit Root Test
  Random Walk: No Drift, No Trend
  Frequency: Quarterly
  Sample Period: 195004-200004
  Number of Observations = 201

DEL(x)=B1*x(-1)+B2*DEL(x[-1])

         Parameter Tau-Statistic Std Error    Mean
  B1)      0.00066    6.47898      0.00010   8.317
  B2)      0.33170    5.08691      0.06521   0.009

  Variance=        0.000084  Dependent Mean=     0.008
  Standard Error=  0.009174  Percent Error=     109.35%
  R-Square=          0.4912  R-Bar-Square=      0.4887
  Schwarz Criterion=-9.3399  Akaike Criterion= -9.3728

Augmented Dickey-Fuller Unit Root Test
  Random Walk: No Drift, No Trend
  Frequency: Quarterly
  Sample Period: 195101-200004
  Number of Observations = 200

DEL(x)=B1*x(-1)+B2*DEL(x[-1])+B3*DEL(x[-2])

         Parameter Tau-Statistic Std Error    Mean
  B1)      0.00063     5.52134     0.00011   8.322
  B2)      0.31247     4.38889     0.07120   0.008
  B3)      0.05458     0.78423     0.06960   0.009

Variance=         0.000085  Dependent Mean=     0.008
Standard Error=   0.009206  Percent Error=     110.35%
R-Square=           0.4879  R-Bar-Square=      0.4827
Schwarz Criterion= -9.3113  Akaike Criterion= -9.3608
```

```
Augmented Dickey-Fuller Unit Root Test
  Random Walk: No Drift, No Trend
  Frequency: Quarterly
  Sample Period: 195102-200004
  Number of Observations = 199

DEL(x)=B1*x(-1)+B2*DEL(x[-1])+B3*DEL(x[-2])+B4*DEL(x[-3])

         Parameter Tau-Statistic Std Error    Mean
  B1)      0.00066     5.42521     0.00012   8.326
  B2)      0.31608     4.41923     0.07152   0.008
  B3)      0.07703     1.02829     0.07491   0.008
  B4)     -0.06257    -0.89373     0.07001   0.009

  Variance=        0.000085  Dependent Mean=     0.008
  Standard Error=  0.009234  Percent Error=     110.87%
  R-Square=          0.4881  R-Bar-Square=      0.4802
  Schwarz Criterion=-9.2836  Akaike Criterion= -9.3498

Augmented Dickey-Fuller Unit Root Test
  Random Walk: No Drift, No Trend
  Frequency: Quarterly
  Sample Period: 195103-200004
  Number of Observations = 198

 DEL(x)=B1*x(-1)+B2*DEL(x[-1])+B3*DEL(x[-2])+B4*DEL(x[-3])+
        B5*DEL(x[-4])

         Parameter Tau-Statistic Std Error    Mean
  B1)      0.00073     5.54093     0.00013   8.330
  B2)      0.31140     4.34908     0.07160   0.008
  B3)      0.08282     1.10409     0.07501   0.008
  B4)     -0.05084    -0.67632     0.07517   0.008
  B5)     -0.08086    -1.15305     0.07012   0.009

Variance=        0.000085  Dependent Mean=     0.008
Standard Error=  0.009224  Percent Error=     111.32%
R-Square=          0.4900  R-Bar-Square=      0.4794
Schwarz Criterion=-9.2639  Akaike Criterion= -9.3469

Augmented Dickey-Fuller Unit Root Test
  Random Walk: No Drift, No Trend
  Frequency: Quarterly
  Sample Period: 195104-200004
  Number of Observations = 197
```

```
DEL(x)=B1*x(-1)+B2*DEL(x[-1])+B3*DEL(x[-2])+B4*DEL(x[-3])+
       B5*DEL(x[-4])+B6*DEL(x[-5])

         Parameter Tau-Statistic  Std Error    Mean
  B1)      0.00083    5.86591       0.00014   8.334
  B2)      0.29491    4.11805       0.07161   0.008
  B3)      0.08208    1.09959       0.07465   0.008
  B4)     -0.04221   -0.56345       0.07491   0.008
  B5)     -0.06470   -0.86327       0.07494   0.008
  B6)     -0.11030   -1.57162       0.07018   0.009

  Variance=         0.000084  Dependent Mean=     0.008
  Standard Error=  0.009169  Percent Error=     111.45%
  R-Square=           0.4951  R-Bar-Square=      0.4819
  Schwarz Criterion=-9.2539  Akaike Criterion= -9.3539

Augmented Dickey-Fuller Unit Root Test
  Random Walk: No Drift, No Trend
  Frequency: Quarterly
  Sample Period: 195201-200004
  Number of Observations = 196

DEL(x)=B1*x(-1)+B2*DEL(x[-1])+B3*DEL(x[-2])+B4*DEL(x[-3])+
       B5*DEL(x[-4])+B6*DEL(x[-5])+B7*DEL(x[-6])

         Parameter Tau-Statistic Std Error    Mean
  B1)      0.00077    5.02196      0.00015   8.339
  B2)      0.30573    4.21000      0.07262   0.008
  B3)      0.08673    1.15663      0.07499   0.008
  B4)     -0.04272   -0.56820      0.07519   0.008
  B5)     -0.06836   -0.90772      0.07531   0.008
  B6)     -0.11740   -1.55387      0.07555   0.008
  B7)      0.05379    0.75940      0.07083   0.008

  Variance=          0.000085  Dependent Mean=     0.008
  Standard Error=   0.009193  Percent Error=     111.30%
  R-Square=            0.4977  R-Bar-Square=      0.4817
  Schwarz Criterion= -9.2264  Akaike Criterion= -9.3435
```

Chapter 7
Several Historical Considerations

Early in Chap. 4, in order to introduce the set of values later presented as "Core" or "Basic" statistics, several examples of early regression displays were shown, starting with one from 1962 for the program RGR. The purpose was, in part, to demonstrate the historical antecedents for considering such a set of statistics as a group, for as that discussion illustrates, starting from the earliest days of econometric software, what later occurred historically was for evaluative statistics to be added progressively to the standard regression display from time to time. Actually, to say "from time to time" may convey too much of an impression of steady change. Although there was some variation from program to program during the period from 1962 to 1985, for most of this almost quarter of a century time period there was both a high degree of uniformity in the statistics displayed and relatively little change, notwithstanding that each econometric software developer was in principle able to chose independently which statistics to display and in what format. In contrast, from 1985 to the present day, not quite 25 years, has been a period of more change and less uniformity, especially the past 10 years, reflecting both the propagation of contesting econometric ideas and the "market dynamics" of the "econometric software industry." Such differences possibly mirror the change in econometric practice and this thought stands behind the content of this chapter.

However, these most obvious aspects do not tell the whole story. The provisions made for data management and access, including both software facilities and usable data sets, also affect both the quantity and quality of empirical research. It is possible to argue that these facilities are today the best they have ever been, reflecting the development of the Internet and the other capabilities of the modern computer. But, as demonstrated earlier, they are far from perfect as an enabling factor. In addition, there are more general questions that might be asked concerning the way in which research results are customarily presented. For instance, considering the modern economics literature, it is troubling that it is both possible and seemingly generally acceptable to publish books subtitled *Theory and Empirical Evidence* (Gong & Semmler, 2006) that neither precisely identify the empirical data used nor the details of the computational methodologies applied, often using the word "algorithm" as a synonym for an ideal mathematical formula, rather than the implemented code that computationally expresses that formula. It is not clear whether this type of approach reflects an increasing degree of unfamiliarity with economic measurement concepts

C.G. Renfro, *The Practice of Econometric Theory*, Advanced Studies in Theoretical and Applied Econometrics 44, DOI 10.1007/978-3-540-75571-5_7,

on the part of economists generally or instead an unwillingness to take the trouble to work with actual data and then describe that work in a way that would facilitate its replication by others. But whatever the reason, it has become all too common for analysts to focus nominally upon techniques, rather than to consider the correspondence between theory and the real world. The consequence is an increasing lack of communication: once data have been gathered, fed into a computer program that is only generically identified, and the findings then described in too cursory a fashion, the report on this research effectively becomes imaginary to the reader – irrespective of its actual underlying content.

This characterization is not intended to suggest the insincerity of reported results, but rather to point out that well designed and easily applied computational facilities can, on the one hand, have the effect of making the research process not only less tedious but also actively enjoyable, for both the researcher and anyone who later considers the findings. Consequently, there is both a point and a purpose in investigating how to design and develop software that fosters both effective research and its assessment. On the other hand, the way in which research is reported, and the degree to which others have the immediate ability to evaluate the details, if they so wish, has an effect on the extent to which they are then able to comprehend it fully. Inadvertent errors can of course be made during the research process, which is an aspect of the need to evaluate results, but even in the absence of error it is generally difficult for anyone to grasp all the implications second hand. There is simply too much information that inevitably "falls through the crack." At one time, the technology of the information transmission process, or rather its relative rusticity, possibly justified this inefficiency. Today, it is unnecessary, except to the degree that twenty first century economists choose to cling to twentieth century and earlier print-based, even hard copy determined technologies – and the implications also raise the question whether empirical research that continues to be reported in late twentieth century fashion will ultimately be judged to have sufficient information content to be worth the paper it is printed on.

This chapter considers certain historical aspects of the way in which parameter estimates have been displayed, so as to supplement the findings reported in the preceding several chapters. However, what became clear as this chapter was put together, and needs to be recognized as it is read, is the degree to which possibly critical historical information has been lost. In the case of large-scale econometric models especially and other complex entities, it is obviously extremely difficult in hindsight to assess their actual characteristics in the absence of an ability to employ the original data used and examine in detail the complete set of results obtained. However, although not so immediately obvious, recapturing the past is also difficult even for small models and specific equation specifications. The inference to be drawn is that, today, now that each economist either possesses a computer or else has ready access to one, simply reading an abbreviated report about a research project and its findings represents yesterday's methodology, an outdated convention that has been allowed to persist into the present. It can be argued that the traditional journal article or book, presented as a report on applied research performed, still has a place in scholarly communication, but now possibly should be presented as an abstract to an associated CD-Rom or other machine-readable media containing documentation, data, and software that potentially enables the interested reader to examine the findings at any desired level of detail. Of course, at some stage it may be necessary to go further and to consider how to make any software supplied in this way easy for the reader to use.

What is displayed, and how, can have a significant impact on the ability to properly interpret research results. On the face of it, this idea is plausible, even if the exact amount of credence to give it is open to question. But, so far, the economist's interaction with research facilities seems to be overly passive: what appears to be true is that most users of econometric software will utilize the particular facilities available without attempting to improve on them. An additional, possibly closely associated aspect is the degree of effectiveness of this utilization. It appears – or is certainly conventionally believed – that software users rarely begin to learn to use a new (or updated) program by first reading carefully any associated manuals (Cuff, 1980). There is no reason to expect economists to behave differently simply because the "manuals" might include information about such things as the properties of the particular misspecification tests performed. Consequently, it is likely that the typical analyst, confronted with a display of statistics, will from the beginning take it at face value and attempt to make any necessary inferences simply on the basis of the information it most obviously provides. Today, this display will usually identify the presumed distribution of each misspecification test statistic; it might also include in each case the "p-values" for the analyst to use as evaluative criteria. In addition, plots and other graphical displays might also be provided. In some cases, using hypertext and similar referencing technologies, it might even be possible for the analyst to summon help or "drill down" in various ways to look at the data more closely or otherwise inspect the underpinnings; however, these more exotic facilities, even if technically feasible, constitute future possibilities, rather than current capabilities. But now is the time to begin to imagine them.

Changes in Regression Displays 1969–2007

It has been demonstrated in previous chapters that some of the individual statistics that are shown in the parameter estimate displays that populate the existing econometric software packages have idiosyncratic characteristics that can cause them to differ arbitrarily from one package to the next. When the user is sufficiently knowledgeable and experienced, these differences might not be particularly significant, although an implication of this finding could still be that econometric software developers should in the future pay more attention to explaining to users exactly how the statistics displayed are formulated. More generally, it is also evident from the results presented in Chap. 4 through 6 that there is some need to consider the range of tests provided, although what may also be needed, as suggested there and in Chap. 3, is the cooperative participation of econometric theorists.

At the moment, however, it is useful to examine in somewhat greater detail, but nevertheless in passing, the way in which the regression display has changed during the past almost 40 years. One aspect of this change is the default display of evaluative statistics. Another is the impact of changing computer technologies. Two technological developments in particular stand out. First, recall that during this time there has been a fundamental change in the nature of the human interface. In 1969, it was common for numbers placed in fixed fields on punched cards (or sometimes paper tape) to control the operation of programs. Programs responded by causing the computer to print large sheets of paper, often in considerable quantity, sometimes exhibiting quite rudimentary formatting and lacking supporting information. Very occasionally, both input and results were displayed on CRT screens, but these monochrome screens were often limited to a width of perhaps 40–80 characters and a height of as few as 20–25 rows. Today, instructions are transmitted by clicking on icons or menu items, often using a pointing device in the form of a mouse. Alternatively, commands may be issued in the form of text that is then parsed and interpreted by the program. The results are conventionally displayed in the form of tables and comparatively high quality graphics, sometimes using multiple tiled or overlapping "windows" on screens that, in width, can display as many text characters per row as the large sheets of paper in 1969 and, in height, 50, 60, or sometimes even more rows, depending upon the size of the typeface displayed.

Second, the amount of computer memory available to a single microcomputer user has increased from no more than 64 KB to a gigabyte or more and simultaneously the execution speed has increased from a few MHz to several GHz. The modern computer is now capable of generating in a few minutes more output than an unaided person might be able to evaluate effectively in a lifetime. A single palm-sized external hard drive is capable of containing all the economic data on all the economies of the world that have ever been produced and published by the standard sources. Exactly how to organize and manage these resources is a subject that needs thought. This subject will be ignored here, but its spectral presence has been alluded to in order to indicate that there are important matters that lurk in the background.

The change in regression displays during the past almost 40 years will be considered using as examples three of the existing econometric software packages, MODLER, PcGive, and TSP. Although these packages are increasingly convergent in a number of respects, they are quite different in their origins and initial intent. TSP is the oldest of these, dating from 1965 to 1966, and appears to have been created originally in order to avoid the use of a desktop calculating machine, the software's original author, Robert Hall, having chanced upon a minimally used IBM 1620 in the basement of the Sloan Building at MIT (Hall, 2003). MODLER and PcGive were each initially conceived in or about 1968, the first at the Brookings Institution, to permit a single person to estimate effectively a new version of the Brookings Quarterly Econometric Model and other models of similar size, and the second at the London School of Economics (originally as AUTOREG), to "estimate econometric models with autoregressive errors" (Hendry, 1970). How much these original characteristics are evident from a few screenfulls of text and numbers remains to be determined, but there is perhaps more information here than is instantly apparent and which the textual commentary may help to reveal.

MODLER Regression Output 1969–1970

As discussed in Chap. 2, the creation and use of large-scale econometric models was one of the early spurs to the development of econometric software, not only to estimate parameters but also to manage databases and perform other model-related process management tasks. Of all of the early models, the one that posed the greatest challenge in this broader sense was the Brookings Quarterly Econometric Model of the United States. Lawrence Klein has traced its origins to speculative discussions at early meetings of the Social Science Research Council's Committee on Economic Stability in 1959–1960 (**?**, **?**, p. iv–vii; Klein, 1975, p. 13). Robert A. Gordon has instead argued for an earlier date, from the time of a 1957 proposal to set up a committee on business cycle research (Gordon, 1975, p. 31), noting that this model's original name was the SSRC model, then the SSRC-Brookings model, and only finally the Brookings model.

Whichever the specific choice of inaugural date, this model's estimation, construction, and solution was for that time computationally formidable, from the beginning until its swansong in 1975 – perhaps far greater than originally imagined and certainly greater than is evident from the contemporaneous reports on the model's development that are still available today. However, this statement needs careful amplification. The initial challenge was essentially organizational and managerial. At first, this model was estimated collectively by many scholars at various different institutions, with its component parts, or sectors, estimated independently (**?**, **?**, 1969; Fromm & Klein, 1975). The ostensible motivating concept was that by utilizing the knowledge of people who had particular sectoral expertise it would be possible "to increase our knowledge of the structure of the American economy" (Duesenberry & Klein, 1965, p. 3), but there was always the question of creating the whole. As explained in the preface to the second volume, published in the middle of 1969 (Duesenberry et al., 1969, pp. v–vi), but written in the early autumn of 1968,

> When the original large-scale system was first planned and constructed, there was no assurance (or knowledge from previous research) that the separate parts would fit together in a consistent whole. There was also no experience to suggest that the tremendous quantity of information required for such a system could be handled efficiently. Both efforts were successful: a condensed 200-equation version of the model has been solved as a simultaneous system and subjected to policy and stochastic shocks. This has required a slow and painstaking process of compiling and constructing data series from diverse sources in a consistent, compatible way (several thousand data series underlie the estimation and application of the model). It also required the design of efficient computer programs for data storage and transformation, simultaneous equation estimation, and model solution. . . .we have been forced to build a statistical computer system. The data bank was the initial step in this direction. . .[and the] program [to manage it] is virtually complete and operational. It provides for storing, verifying, and retrieving data series from computer files. (Currently, these files are kept on magnetic tape. Thus, the retrieval process is sequential and, therefore, time consuming; placing the files on disc – which is planned for the near future – will afford nearly instant random access to individual requested series). . .The transformation package has been linked (after modifying improvements) to a number of currently available estimation routines – Harvard University's SLURP, the University of Pennsylvania's ECON, and Robert E. Hall's TSP. It is now possible to withdraw data from the bank, [make transformations], and estimate functions directly using ordinary least squares, two-stage

least squares, two-stage least squares with principal components of independent variables, or limited-information maximum likelihood.

However, this description can be misleading, if interpreted too literally from a twenty first century perspective. There is almost always a contemporaneous incentive to present possible outcomes in a favorable light; for example, it has only recently been publicly revealed that, immediately after lift-off, the launch director of the Apollo 8 around-the-moon flight in December 1968 regarded the odds of a successful mission at only 50–50. In the case of the Brookings model, the details of its history also cannot be understood without a good appreciation of the degree to which it may have caused a difficult change in the scale of applied economic research for those economists individually involved, not all of whom had previously developed a good working relationship with a local computer. There is now minimal evidence of the way in which each of those responsible for model sectors gathered and managed the data used, or of how the computations were made. Some may have used computers, as implied, but at least a few undoubtedly used desktop calculating machines such as the Marchand, the Friden, or the Monroe. In any case, the results were transmitted in hard copy form, a distillation of which exist today in the form of the first two model volumes, the first of which was published in 1965 and the second in 1969, as just mentioned (Duesenberry et al., 1965, 1969).

The initial strategy may have utilized available knowledge and minimized the estimation problem, but as indicated it also required, as a distinct model building step, the later formation of a unified solvable model that compatibly integrated all these individual results. This work was in varying degrees performed by economists at several places, including the universities mentioned, but from September 1963 the model project staff function was officially located at the Brookings Institution in Washington, DC, in order to provide support for the ongoing development of the model. By 1968, as just stated, an integrated data base had been created there by James Craig and others, together with the necessary software to manage it, involving "programming and computational assistance by John Ahlstrom, Vijaya Duggal, Mark Eisner, Peter Hughes, and Ross Preston" (14, p. xi).

To create from this proffered diversity a consistent, unified model was a substantial undertaking. By the middle of 1968, software had been created for the IBM 7040 at Brookings that permitted the creation and maintenance of a substantial time series database in the form of "two large, formatted tape files (data banks) for the storage, retrieval, and use of time series. One bank, for quarterly data, [contained] over 2,400 series plus documentation. The other bank, to which minimum resources [were] devoted, [was] for monthly data. It [contained] almost 700 series" (14, p. vi). There was also software that, with difficulty, would permit parameter estimation. At Brookings in October 1968, source code was available, in particular, for the ECON program, Zellner's 3SLS (the University of Chicago version, with modifications by Hudson Thornber) (Zellner, Stroud, & Chau, 1963b), and several statistical regression packages, including BMD and DAM 67, as well as the Yale and BMD03M factor analysis programs, which could be used to produce principal components. A model solution program was also available – but at the earliest in 1969 – in the form of BROOKSIM, the Brookings model simulation program, a still-existing source listing for which, dated June 1969, identifies its authors as "K. Mori, T. F. Cooley, M.D. McCarthy, L. Klein" and includes the coded equations for a version of the model. At that time, for all these programs, the data employed were read from cards.

It is nonetheless true that in February 1969, it was "possible to withdraw data from the bank, [make transformations], and estimate functions directly using ordinary least squares, two-stage least squares, two-stage least squares with principal components of independent variables, or limited-information maximum likelihood," as the preface indicates, but at Brookings this was not done using any of the programs named. The original intention, as the preface quotation indicates, was to modify ECON minimally in order to read the Brookings data bank tape. However, of necessity, this became a more than minimal rewriting. In addition, at about this time, the Brookings trustees decided to change computers, from an IBM 7040 machine to a new Digital Equipment Corporation (DEC) PDP-10, a networked, time-sharing computer, for which the relevant applications software had yet to be written. Even before the installation of the PDP-10, it was necessary to create a new program to serve as the basis for planned research that included the estimation of the mentioned "condensed" model entirely by members of the project staff.

1969: MODLER 1.0

This program, today known as MODLER, was originally created specifically to retrieve data series directly from the (formatted tape file) Brookings data bank and then to make transformations and perform other data management operations. Ideally, it was necessary to perform these tasks efficiently with a minimum of human effort. The IBM 7040 was a batch processing, second-generation computer, providing only nine-track reel tapes for permanent storage and thus had rather rustic data management capabilities. MODLER 1.0 incorporated some code from ECON, and certainly some ideas, but also a matrix inversion algorithm based upon the then well-known MATINV subroutine, written by Burton S. Garbow at the Argonne National Laboratory. Its coding took into account Longley's findings (Longley, 1967), concerning the pitfalls of single precision computation, and its interface – as much as could be prospectively imagined – the future need to permit users to issue interactive commands in a free format semi-natural language form, which required commands to be parsed and interpreted.

Figure 7.1 displays regression output from MODLER 1.0 consisting of an equation that "explains" real Consumer Durable Expenditures, excluding automobiles, divided by population (CDEA58/NR), where the deflator base year was 1958. The run "printout" is dated 21 March 1969, which implies that the computer used was the IBM 7040. The actual printout exhibits more spacing between numbers and labels than shown here. In addition, it includes a crude graph of the predicted versus actual values of the dependent variables and a table beside, showing these values for the sample period, plus the residuals, and the percentage errors these residuals represent. Also shown were the raw sums of squares and cross products and the

```
DEPENDENT VARIABLE CDEA58/NR   25   MEAN  0.161742     STANDARD DEVIATION 0.022730

NUMBER OF OBSERVATIONS= 48    NUMBER OF VARIABLES = 5

   VARIABLE                               STANDARD                                    STANDARD
NAME          NUMBER    COEFFICIENT         ERROR           T-TEST       MEAN      DEVIATION

YD58/NR        1(26)   0.23749223D 00   0.24753315D-01   0.959D 01   1.923482   0.144146

PCDEA/PC       2(33) -0.73827485D-01   0.76880331D-01  -0.960D 00   0.983623   0.044841

KCDEA58/NRLG   3(39) -0.31731764D 00   0.13249028D 00  -0.240D 01   0.477821   0.046994

DMY04          4(23) -0.22685046D-02   0.22807376D-02  -0.995D 00   0.916667   0.279310

CONSTANT       5( 0) -0.68750521D-01   0.10151750D 00  -0.677D 00   1.000000 -0.000000

EXPLAINED SUM OF SQUARES= 0.127950D 01

VARIANCE = 0.11532160D-04

STAND ERR = 0.3395903E-02

R-SQUARED=   0.9777   R=   0.9888  (ADJUSTED FOR DEGREES OF FREEDOM)
R-SQUARED=   0.9796   R=   0.9897  (UNADJUSTED FOR DEGRESS OF FREEDOM)
F-TEST( 4, 43) = 515.67
DURBIN-WATSON D STATISTIC=  0.793D 00
```

Fig. 7.1 MODLER 1.0 regression display

parameter variance-covariance matrix and the scaling factor used in the inversion process. Optionally, the inverted $\mathbf{X}'\mathbf{X}$ matrix could be displayed. Other estimation methods, including two-stage least squares, could be employed if the user wished, which provided more copious, step-by-step displays.

Series were retrieved from the Brookings data bank by record number, assigned labels and new locator numbers, then transformed as necessary in order to generate the observation values actually used in the regression. The need to state regressions in terms of transformed variables, as well as to create dummy and other types of "synthetic" variables, as shown by Fig. 7.1, created the requirement for a temporary storage facility, what is today called a workspace or a "Memory File." However, the IBM 7040's available memory was tiny, especially by today's standards, so that "scratch files" were used to hold the Memory File series observations. The two Brookings programs that MODLER replaced consisted of a separate data transformation program, operated using numeric codes, and a regression program that read the tape created by the data transformation program. This tape contained all the dependent and regressor variable observations both transformed and untransformed. MODLER bypassed these two programs and read the data bank observations directly.

However, what is more significant is what else it did. The earlier ECON program permitted transformations to be made internally, but required all the transformations for each separate run to be coded into a dedicated Fortran subroutine, referencing each series as a column of a data matrix. One of the consequences was that errors could easily be made. As important, these errors might not be apparent until much later, at which time the affected estimated equations would have been coded into a model solution program. In the context of the estimation of a multi-hundred-equation model, because of the scale of this undertaking, it was obviously useful to be able to see from the regression display which transformations had been performed on which variables. ECON permitted the user to label variables, but not in the same revealing way. In addition, MODLER processed transformation commands effectively as macro instructions. However, MODLER 1.0 was only a first step in this direction, for it did not incorporate code to perform all possible transformations, only those most commonly performed; others needed to be made in advance. This version of MODLER was always expected to have a short life for, at its creation, the IBM 7040 at Brookings was of course already scheduled for replacement, even if it was not then known by what.

1970: MODLER 2.0

On or about 1 January 1970, the IBM 7040 at Brookings was replaced by the Digital Equipment Corporation PDP-10. The significance of this new machine was not just that it was designed explicitly for time-sharing, but also that it was hard-wired (local area) networked. Terminals with keyboards and CRT screens were located in office suites throughout Brookings and certain nearby buildings. No longer was it necessary for either a programmer or a user to walk to the computer center in order to submit a job. Furthermore, because of the machine's time sharing nature, so long as the number of users represented less than machine capacity, using the PDP 10 almost gave the impression of its being a single user machine, so much so that using it in 1970 and using the IBM PC/XT (the first PC to contain a hard drive) in 1982, were actually surprisingly similar experiences. The PDP 10 offered a removable storage facility called a "DECtape," which supported direct access to individual file records, not so very different than the file access conventions of the IBM PC/XT. The most obvious difference was that to use a DECtape, it was first necessary to send a message to the PDP 10's operator, who would then physically mount it. But once mounted, it could be accessed all day. The programs themselves could also be made available simultaneously to all users of the machine, so that these were easily multiply used. As is stated in the Wikipedia entry for DECtape, these media "were 3/4 in wide and formatted into blocks of data that could be read or written individually. One tape stored 184 K 12-bit PDP-8 words. Block size was 128 words. From a programming point of view, DECtape behaved like a very slow [hard] disk drive." At the time, however, it did not seem particularly slow. Because of Paul Allen's continuing affection for this machine, aspects of it can still be experienced at www.pdpplanet.com.

MODLER 2.0 incorporated some of the algorithmic code of MODLER 1.0, but was fundamentally a different program in the way it operated. Very much like an interactive IBM PC program 10–15 years later, running under DOS, once in execution MODLER 2.0 responded directly to each successive command issued by a user. Either Data Banks common to multiple users or a Memory File (or data bank) privately accessed by a single user could be used. An economist could work all morning, go to lunch, come back and pick up again where he or she had left off. As a consequence, the context placed a premium on the ability to parse and interpret what progressively became more or less "natural language" commands (albeit rather formalized), in order to manage a data base, make transformations, and execute a regression command, among other operations. This language development occurred gradually as the program began to be used. Actually, MODLER 2.0 rapidly became widely used at Brookings, by more than just the members of the Model Project, since it was the first general econometric program to become available for this machine, all other software having "expired" with the removal of the IBM 7040. Within 2 months of the PDP 10's installation, economists and others throughout the Brookings buildings were using it rather intensively and active feedback from these users encouraged its further development and improvement. Figure 7.2 displays the default onscreen regression output of MODLER 2.0 as of 18 March 1970.

In order for Fig. 7.2 to display exactly comparable values to those presented in Fig. 7.1 this display is actually simulated – essentially to emphasize the contrast between the displays using the same values, inasmuch as matching sets of results using different program versions do not survive from that time. Notice that the content of Fig. 7.2 is more selective than that of Fig. 7.1, which reflects that the size of the user's terminal screen, at only 24 rows by 80 columns, more strictly limited the amount of information that could be displayed at any one time, even compared to a single printout page, which was 60 rows by 132 columns. However, because of interactive processing, the offset was that the user then potentially had the immediate ability to request further information. The graph of sample period actual and predicted values of the dependent variable is likewise not shown, but in this context it was never produced automatically; MODLER 2.0 presented this graph, as well as a range of other supplementary output, only upon demand. Of course, an implication of this more selective display was possibly a greater econometric knowledge requirement, since the user now took on the responsibility to display and evaluate these extra results.

In contrast, the regression displays previously shown as Figs. 4.4 and 4.5 in Chap. 4 are modern MODLER displays, to which Fig. 7.2 can also generally be

```
REGRESSION OF CDEA58/NR    (MEAN=  0.161742)
ON:

                               COEFFICIENT       T-TEST       MEAN

YD58/NR          1(26)      0.23749223        9.5913       1.9235
PCDEA/PC         2(33)     -0.07382749       -0.9601       0.9836
KCDEA58/NRLG     3(39)     -0.31731764       -2.4396       0.4778
DMY04            4(23)     -0.00226850       -0.9951       0.9167
CONSTANT         5(48)     -0.06875052       -0.6768       1.000C

VARIANCE =       0.00001153
STAND ERROR =    0.00339590
R-SQUARED=       0.9796
R-SQUARED=       0.9777   (ADJUSTED FOR DEGREES OF FREEDOM)

F-TEST( 4, 43) = 515.67
DURBIN-WATSON D STATISTIC=  0.793
```

Fig. 7.2 MODLER 2.0 regression display compared. An obvious difference is the use of an explicit equation representation in Figs. 4.4 and 4.5 to communicate the nature of the regression performed, as well as additional supporting statistics. In the early 1970s, as the possible transformations grew progressively more complex, it soon ceased to be possible to display at least some of these in the space to the left of the parameter estimates. In addition, the estimated equations might also, upon selection, become model equations, so that it seemed logical to exhibit the candidate equation as part of the regression display, in a way that included each of the embedded variable transformations. MODLER generally shares this way of representing a regression result with TROLL – compare Figs. 4.2 and 4.4 – and has since the early 1970s. However, in MODLER's case, this equation display is a program-generated result, reflecting the program's long-standing capability to create "soft-coded" estimated equations that, on command, can then be directly inserted in models. In 1970, MODLER was only just on the verge of becoming an integrated econometric modeling language (Renfro, 1996, 2004a, c), so that this refinement occurred later.

TSP Regression Output 1978–2007

The regression displays in Figs. 7.1 and 7.2 can be presented here only because of the chance survival of program listings and some old printouts, rather than because anyone thought to keep them just in case they might be interesting nearly 40 years later. In contrast, historical information about TSP is the result of the survival of user guides. These user guides have interesting historical aspects. As mentioned before, TSP was created at MIT by Robert Hall, beginning in 1965–1966, with the later advice and assistance of others including Ernst Berndt, Charles Bischoff, James Brundy, J. Phillip Cooper, Ray Fair, Robert J. Gordon, Bronwyn Hall, Dale W. Jorgenson, Thomas J. Moore, and Richard Sutch.

By 1969, when TSP was cited in the preface to the 1969 Brookings volume, it was a comparatively widely used program and already relatively well known among (computer using) economists of that day. It was being developed in parallel with TROLL. In addition, Robert Hall to some degree utilized TSP in his and others' work on the development of the first large-scale data bank and analysis programs for Data Resources Inc, including EPL and EPS in the late 1960s and early to mid 1970s. Development of the original TSP program was taken over by Bronwyn Hall in 1970, and in 1971 it became a distributed mainframe product, then a commercial product in 1977, with the founding of TSP International. A PC version was released in the mid 1980s. Meanwhile, a number of earlier public domain versions of TSP had directly influenced the development of yet other programs, including SORITEC and MicroTSP, the latter a result of a joint venture, originally so that this program could run on the Apple II. MicroTSP subsequently evolved into EViews (Berndt, 1991; Renfro, 2004a, c), Clint Cummins became the principal developer of TSP during the mid 1980s and continues in that role today.

In 1978, Bronwyn and Robert Hall began to use then newly available document production software to prepare the User Guide for TSP 3.3. Originally, TSP user guides were simply typed manuscripts, until about version 2.5, and then produced as line printer output in utilitarian fashion; these guides have since almost all been thrown away. Of course, today, the use of "word processing" software would hardly constitute a notable event, but in 1978 it was much less common. The 1978 User Guide was printed on heavy paper, using a dot matrix printer, as was usual until the laser printer appeared. The first laser printer was created by Gary Starkweather at Xerox in 1971 and was a modified Xerox copier. The first commercially produced laser printer was the IBM 3800. It originally became available in 1976 and was approximately the size of a Sports Utility Vehicle (SUV). At about the same time, Xerox produced the Xerox 9700, also of a similar size. The first mass market laser printer, intended to be used with a personal computer, was the HP Laserjet, first released in 1984, which sold for approximately $3000 and weighed 71 lbs. It was comparable in size to the Xerox Star, released in 1981 and which had then sold for $17,000. However, the significance of the 1978 TSP guide is not only how it was produced, but also that it displayed output from an illustrative model, intended to demonstrate the use of several techniques, including OLSQ, INST, LSQ, ANALYZ, FIML, and SIML. The OLSQ output in particular was subsequently updated in the later editions of this User's Guide and this serial updating provides the means to compare the sample output over the years beginning in 1978.

1978: TSP 3.3

The regression display shown in Fig. 7.3 has been condensed slightly to fit the present page. As shown, the diagnostics are left adjusted, not aligned on the = sign and are displayed in upper case, inasmuch as most computer output at that time would have been produced in this form. This display obviously shows both the immediate regression results and the parameter variance-covariance matrix. Incidentally, this format and the information displayed are essentially the same as that for the previous TSP version, 2.8, so that this display generally represents a mid to late 1970s view of the TSP regression output. Notice that in comparison with the MODLER displays in Figs. 7.1 and 7.2, TSP differs by the inclusion of the log likelihood, the sum of the residuals (the value reflecting the absence of a constant term), and the sum of squared residuals. These displays were independently created, so that the similarities were determined by the conventions of the time, notwithstanding the time difference in production dates.

As discussed previously in Chap. 3 and 4, these "core" statistics were customarily the set available to any economist who used econometric – or, for that matter, statistical – software during the 1970s. A more comprehensive evaluation of the "regression display" of the 1970s, taking into account those produced by all the programs then publicly available, might reveal a number of additional small differences from program to program of the type just described, but not significant variation from what has been shown here.

```
EQUATION   1
************

ORDINARY LEAST SQUARES


DEPENDENT VARIABLE:   CONS

SUM OF SQUARED RESIDUALS =          3958.15
STANDARD ERROR OF THE REGRESSION =          12.5828
MEAN OF DEPENDENT VARIABLE =        519.032
STANDARD DEVIATION =         147.459
R-SQUARED =  0.9930
ADJUSTED R-SQUARED =  0.9927
F-STATISTIC(   1.,     25.) =          3545.80
LOG OF LIKELIHOOD FUNCTION =        -105.645
NUMBER OF OBSERVATIONS =      27.
SUM OF RESIDUALS =          .366211E-02
DURBIN-WATSON STATISTIC (ADJ. FOR  0. GAPS) =  0.4617


RIGHT-HAND           ESTIMATED            STANDARD            T-
VARIABLE             COEFFICIENT            ERROR           STATISTIC
C                    -17.8024             9.33493           -1.90707
GNP                    .633919            .106458E-01       59.5467


ESTIMATE OF VARIANCE-COVARIANCE MATRIX OF ESTIMATED COEFFICIENTS

                          C                 GNP
                 ..............................
C                     87.1409          -.959755E-01
GNP                -.959755E-01         .113332E-03
                          1                  2
```

Fig. 7.3 TSP 3.3 OLS regression display

1983: TSP 4.0

Compared to the last display, in Fig. 7.3, notice that in Fig. 7.4 the diagnostics are right adjusted, aligned on the = sign, rather than left-adjusted. Obviously, this is purely a stylistic change. However, in this version of TSP, the substantive change is that the "Sum of Residuals" is no longer displayed, reflecting Bronwyn Hall's feeling that in almost all cases a constant term should be included.

The focus here is upon regression displays over time. However, the steadily broadening characteristics of the development of TSP are of course also relevant to any consideration of this program. Behind the scenes, during the 1970s, TSP changed from a program that provided its users with essentially linear, single equation techniques, which it had been until nearly the end of the 1960s, to one that provided also a variety of both linear and nonlinear techniques, including two and three stage least squares, SURE and FIML. An important characteristic of the nonlinear estimation algorithms of TSP since 1970 has been the use of analytic derivatives. Furthermore, during the 1970s, although use predominately on the IBM

```
                             EQUATION   1
                             ************

              METHOD OF ESTIMATION = ORDINARY LEAST SQUARES


                 DEPENDENT VARIABLE:  CONS

                    SUM OF SQUARED RESIDUALS =      3958.16
              STANDARD ERROR OF THE REGRESSION =    12.5828
                  MEAN OF DEPENDENT VARIABLE =      519.033
                        STANDARD DEVIATION =        147.459
                                 R-SQUARED =        .992999
                        ADJUSTED R-SQUARED =        .992719
  DURBIN-WATSON STATISTIC (ADJ. FOR  0 GAPS) =       0.4617
                    NUMBER OF OBSERVATIONS =        27
                     F-STATISTIC(    1,    25) =    3545.79
                  LOG OF LIKELIHOOD FUNCTION =     -105.645


                      ESTIMATED        STANDARD
  VARIABLE           COEFFICIENT        ERROR       T-STATISTIC

  C                    -17.80238       9.334954      -1.907067
  GNP                  .6339192       .1064578D-01   59.54652

         VARIANCE COVARIANCE OF ESTIMATED COEFFICIENTS

                            C              GNP
                     --------------------------
       C              87.14136
       GNP            -0.09598          0.00011
```

Fig. 7.4 TSP 4.0 OLS regression display

370, it was ported to a number of other mainframe and minicomputers, including those of Honeywell, Univac, DEC, ICL, Burroughs, and Siemens, in the process becoming more widely used.

1987: TSP 4.1

In 1987, the parameter Variance-Covariance matrix was removed from the TSP default display, as well as the "ADJ for 0 GAPS" label from the Durbin–Watson label. Significantly, this version of TSP is the first microcomputer version; this interactive version was produced by Rebecca Schnake. The program began to be converted from the mainframe in 1985, at about the time that Clint Cummins began to assume progressively the principal developer role, finally taking over in 1989 from Bronwyn Hall, who served in that capacity from 1970, first at Harvard and later, in 1977, in Palo Alto. Since that time, a number of additional procedures have been added including GMM, Probit, Logit, Poisson, Negbin, and the general purpose maximum likelihood routine, ML, yet further broadening the program's offerings.

1993: TSP 4.2

In 1993, TSP 4.2 left behind the mainframe look, adopting a new 2-column format for diagnostics, with mixed upper and lowercase text, in order to improve readability and save display space. Mainframe line printers characteristically could only produce uppercase text, resulting therefore in displays containing only capital letters. This look habitually continued to prevail in microcomputer versions of econometric software packages during the 1980s and was phased out rather gradually only during the 1990s, essentially a result of the adoption of the conventions of the graphical user interface.

The econometric change that is evident, when comparing Figs. 7.5 and 7.6, is the addition of (normalized) Schwarz Bayes Information Criteria. In addition, the sample date range was added to the header, as was also the number of observations, moved from among the diagnostic statistics.

1997: TSP 4.4 and Later

In 1997, as shown in Fig. 7.7, TSP 4.4 adopted the regression display format that it has today. Among the particular changes is the addition of three default tests: the LM het test, Jarque-Bera, and the RESET2. The SBIC statistic is now unnormalized. In addition, p-values were added for all statistics, including the Durbin–Watson. In order to make room for the p-values, the label size was reduced. For the Durbin–Watson the lower and upper bounds on p-value are displayed; the values in this case imply that the p-value is less than .0005.

```
                          EQUATION   1
                          ************

                METHOD OF ESTIMATION = ORDINARY LEAST SQUARES


 DEPENDENT VARIABLE:  CONS

             SUM OF SQUARED RESIDUALS =      3958.17
       STANDARD ERROR OF THE REGRESSION =    12.5828
          MEAN OF DEPENDENT VARIABLE =       519.033
                  STANDARD DEVIATION =       147.459
                           R-SQUARED =      0.992999
                  ADJUSTED R-SQUARED =      0.992719
             DURBIN-WATSON STATISTIC =        0.4617
                   F-STATISTIC(1,25) =       3545.79
          LOG OF LIKELIHOOD FUNCTION =      -105.645
              NUMBER OF OBSERVATIONS =            27

                 ESTIMATED       STANDARD
 VARIABLE       COEFFICIENT        ERROR       T-STATISTIC
 C                 -17.802       9.3350          -1.9071
 GNP               0.63392     0.10646E-01        59.547
```

Fig. 7.5 TSP 4.1 OLS regression display

```
                         Equation   1
                         ************

           Method of estimation = Ordinary Least Squares


 Dependent variable:  CONS
 Current sample:  1949 to 1975
 Number of observations:  27

 Mean of dependent variable = 519.033          Adjusted R-squared =.992719
 Std.dev. of dependent var. = 147.459       Durbin-Watson statistic = .461739
   Sum of squared residuals = 3958.17     F-statistic (zero slopes) = 3545.79
      Variance of residuals = 158.327    Schwarz Bayes. Info. Crit. = 5.23184
   Std. error of regression = 12.5828    log of likelihood function = -105.645
                  R-squared = .992999


                  Estimated          Standard
 Variable         Coefficient          Error            t-statistic
  C               -17.8024            9.33496            -1.90707
  GNP              .633919            .010646             59.5465
```

Fig. 7.6 TSP 4.2 OLS regression display

```
                     Equation   1
                     ============

            Method of estimation = Ordinary Least Squares


Dependent variable: CONS
Current sample:  1949 to 1975
Number of observations:  27

       Mean of dep. var. = 519.033      LM het. test = .783303 [.376]
    Std. dev. of dep. var. = 147.459     Durbin-Watson = .461739 [.000,.000]
Sum of squared residuals = 3958.17  Jarque-Bera test = 3.29859 [.192]
   Variance of residuals = 158.327   Ramsey's RESET2 = 16.7953 [.000]
Std. error of regression = 12.5828   F (zero slopes) = 3545.79 [.000]
               R-squared = .992999    Schwarz B.I.C. = 108.941
      Adjusted R-squared = .992719    Log likelihood = -105.645

          Estimated     Standard
Variable  Coefficient     Error        t-statistic   P-value
C         -17.8024      9.33496        -1.90707      [.068]
GNP        .633919      .010646        59.5465       [.000]
```

Fig. 7.7 TSP 4.4 and later OLS regression display

The most recent substantive change has been the updating of the Apple Mac version of TSP, which in 1990 originally involved the creation of a native version for the Apple. In addition, TSP has joined the Ox-metrics family of packages, using GiveWin as an interactive Windows shell. This change permits TSP to be run either as a traditional batch program or interactively. TSP also incorporates programming features, including support for matrix algebra and user-written procedures.

PcGive Regression Output 1985–2007

PcGive, like MODLER and TSP, is one of the longest-lived of the existing econometric software packages. Its earliest characteristics are described in David Hendry's dissertation (Hendry, 1970), where it began life in the late 1960s under the name AUTOREG. It was developed during the 1970s as a "computer program library for dynamic econometric models with autoregressive errors" (Hendry & Srba, 1980), before becoming in the mid 1980s a microcomputer program, since then known as PcGive.

Hendry describes its evolution (Hendry, 1993, p. 314) as mirroring "the story of the methodological developments [of the LSE or General-to-Specific Methodology]. PcGive is unusual in having this type of specific pedagogic rationale. Initially, the [progressive versions] focused on 'optimal estimation' based on the implicit assumption that the dynamic models under study were more or less correctly specified. The tests offered were mainly tests of model specification, where the null was just a special case of an already estimated more general model. Then an increasing number of diagnostic tests for misspecification were included (see for example, Mizon, 1977, Chap. 4) gradually leading to the implementation of 'model building' procedures. In parallel, the Monte Carlo simulation programs evolved to allow the properties of the various new methods to be studied in finite samples, as well as to embody advances in Monte Carlo methods."

1985: PcGive 2

The regression output shown in Fig. 7.8, which represents the first microcomputer version of PcGive, defines the sample size as including both the observations used to perform the regression, that for the years 1922 through 1939, and observations reserved for testing. This version and its use were described in 1987 by Hendry (1987, p. 87) as "an interactive, menu-driven econometrics program designed for modeling *time-series* data in the light of economic theory when the exact specification of the relationship of interest is not known for certain a priori." Deriving from GIVE in the AUTOREG library (Hendry & Srba, 1980), "PC-GIVE uses fast and tested FORTRAN subroutines for accurate numerical calculations, embedded in a user-friendly and highly protected environment which avoids having to learn any idiosyncratic or complicated command language."

At this time, the program was said to be easiest to use on a PC-XT (which then featured a 10 megabyte hard drive) but could (Hendry, 1993, p. 91) "function on even a single disk-drive PC. Present dimensions allow for 160 observations on up to 40 variables in a data file with (a possibly different set of) 35 variables in any equation and up to a further function of 40 variables used in misspecification tests. For scientific documentation all results are stored on disk as calculations proceed..."

The regression displayed in Fig. 7.8 uses the same data originally used to estimate Klein Model I (Klein, 1950, pp. 135–141), but the example itself consists of a

```
SELECTED SAMPLE IS:  1921  1 -  1941  1
OBSERVATIONS      1 To    21  WITH INITIAL SAMPLE SIZE OF   21

  MEANS OF VARIABLES
    C          LAG 1      I          LAG 1      CONSTANT
     53.1833    52.0889     1.0333      .9500     1.0000

  STANDARD DEVIATIONS OF VARIABLES
      4.9440     5.1475     3.6931     3.7036      .0000

  CORRELATION MATRIX
         C        LAG 1     I      LAG 1   CONSTANT
 C         1.0000
 LAG 1      .8322 1.0000
 I          .4580  .1344 1.0000
 LAG 1      .4426  .4696  .7376  1.0000
 CONSTANT   .0000  .0000  .0000   .0000   1.0000

 OBSERVATIONS    2  TO 19  USED FOR ESTIMATION PLUS   2  FOR PARAMETER CONSTANCY
    18 OBSERVATIONS    4 VARIABLES DEPENDENT VARIABLE IS    C

 VARIABLE     COEFFICIENT     STD ERROR     H.C.S.E.      T VALUE  PARTIAL r2
 LAG 1           .96685        .08584        .08210      11.26395     .9006
 I              1.01270        .15643        .20804       6.47359     .7496
 LAG 1          -.78503        .17508        .18447       4.48393     .5895
 CONSTANT       2.52066       4.45185       4.14673        .56620     .0224

 DURBIN WATSON STATISTIC =  1.968      S =       1.50330347
 RESID SUM SQS =  .316389D+02  SSQ =  .225992D+01  RSQ =  .92386  F =     56.62

 INDEX OF NUMERICAL PARAMETER CONSTANCY [CHI**2( 2)/ 2] =        .93
 Values > 2 Imply Poor ex ante Forecasts
 SCALED RESIDUALS
    -.07    -.40    1.03   -1.10     .04     .37     .47   -1.95    -.28    -.38
   -1.68     .13     .17     .44    1.62     .01    1.14     .45     .40    1.31

  TEST FOR AUTOCORRELATED ERRORS

 CHI-SQ. TEST FOR    1 ORDER SERIAL CORRELATION =      .0022


               F-FORM(  1,  13) =      .0016

  ERROR AUTOCORRELATION COEFFICIENTS

     .0140
  RESIDUALS SCALED BY :       1.0000 FOR ARCH
```

Fig. 7.8 PcGive 2 regression display

regression of consumption expenditures on consumption lagged 1 period, investment, and investment lagged 1 period. Compared to other regression output of the middle 1980s, PcGive included more tests of residuals properties reflecting the specification search methodology that stood behind the program's development. However, what is also interesting is the "Alternative Output Format," which restates the regression results in a format similar to that often used in published articles since the middle 1960s (see, for example, Hendry, 1993, Chap. 7). Among other things, this approach permits the direct capture of the results in a form that allows them to be integrated into a document for publication. Although document processing at that time had not reached the WYSIWYG stage in the way it has today, the word processing software available in 1985 did allow short files to be relatively easily incorporated into text (Fig. 7.9).

```
ARCH TEST [ AUTOCORRELATED SQUARED RESIDUALS ]
          CNST     1 LAG
COEFF.    2.4726   -.3338
S.E.'s     .7691    .2417

RESIDUAL SUM OF SQUARES =    .100752D+03     S =       2.59168     F =
1.91

      ANALYSIS OF SCALED RESIDUALS

       Sample Size     18
       Mean                          .000000
       Std.Devn.                     .907485
       Skewness                     -.506575
       Excess Kurtosis              -.072120
       Minimum                     -1.949483
       Maximum                      1.618355

CHI-SQUARED TEST FOR NORMALITY BASED ON 3rd AND 4th MOMENTS :
    CHISQ(2) = [T-K]*(Sk**2 + [Ek/2]**2)/6

               CHISQ(2) =          .602
F =       1.91

      ANALYSIS OF SCALED RESIDUALS

       Sample Size     18
       Mean                          .000000
       Std.Devn.                     .907485
       Skewness                     -.506575
       Excess Kurtosis              -.012619
```

Fig. 7.8 (continued)

```
EQ( 1) BASED ON 1922  1 TO 1941  1 With  2 Forecasts
 C        =    .9668 *LAG 1  +   1.0127 *I   +  -.7850 *LAG 1  +    2.5207
*CONSTANT
 H.C.S.E.  [  .08210 ]       [   .20804 ]  [   .18447 ]      [   4.14673 ]

 R**2 = .92386 F( 3, 14) =    56.62    S =  1.50330347   D.W. =  1.97
 Forecast Chi**2( 2) =     .93

    Mean =          53.183333             S.D. =        4.943950
    Chow  F [ 0.,   0. ] =    .00    Normality Chi**2(2) =       .60
    LM-AR F [ 1.,  13. ] =    .00    ARCH F  [ 1.,  11. ] =     1.91
```

Fig. 7.9 PcGive 2 alternative regression display

1992: PcGive 7

The 1992 version of PcGive (then still called PC-GIVE) is to a degree described in a postscript to Hendry's 1993 book *Econometrics: Alchemy or Science* (Hendry, 1993, Chap. 19). In line with his belief that "an important component of any modeling exercise is to estimate the most general model which it is reasonable to entertain a priori (see Pagan, 1987; Pagan, 1990)," PcGive at that point was intended to facilitate "formulating general linear dynamic models while offering protection against the possibility that the initial generality is in fact too specific to characterize the available data adequately..." Whereas, in contrast, "most econometrics packages focus on the estimation of econometric models of varying degrees of complexity assuming that their qualitative characteristics are known but the numeric values of the parameters need calibration...[and] while estimation represents a necessary

```
EQ( 1) Modelling C by OLS
       The present sample is:  1922 to 1941 less 2 forecasts
       The forecast period is: 1940 to 1941

Variable     Coefficient    Std.Error  t-value        HCSE  PartR2 Instab
Constant          2.5207       4.4519    0.566      4.1467  0.0224   0.29
C_1              0.96685     0.085836   11.264    0.082102  0.9006   0.30
I                 1.0127      0.15643    6.474     0.20804  0.7496   0.09
I_1             -0.78503      0.17508   -4.484     0.18447  0.5895   0.05

R2 = 0.923858  F(3, 14) = 56.622 [0.0000]  s = 1.5033  DW = 1.97
RSS = 31.6388913 for 4 variables and 18 observations

Variance instability test: 0.0541511  ;  Joint instability test: 0.899482
Information Criteria:  SC = 1.20632;  HQ = 1.03574;  FPE = 2.76213

       Seasonal means of differences are
   0.97647
R2 relative to difference+seasonals =    0.77667

       Analysis of 1-step forecasts
   Date        Actual       Forecast       Y - Yhat   Forecast SE    t-value
1940  1        65.0000       64.3998       0.600193      1.81615     0.330475
1941  1        69.7000       67.7373        1.96266      1.93311      1.01529

Tests of parameter constancy over:  1940 to 1941

Forecast Chi2(2)/2 = 0.93195
Chow   F( 2, 14) =     0.51569 [0.6080]

Testing for Residual Autocorrelation from lags 1 to 2
CHI2(2) = 0.68318  and   F-Form(2, 12) = 0.23671 [0.7928]

Error Autocorrelation Coefficients:
            Lag  1     Lag  2
Coeff.     0.07893     0.2335

Testing for ARCH from lags 1 to 1 (Residuals scaled by 1.503303e+000)
CHI2(1) = 1.9184  and   F-Form(1, 12) = 1.5264 [0.2403]

ARCH Coefficients:
          Constant     Lag  1
Coeff.       1.094    -0.3338
Std.Err     0.3805     0.2702

RSS = 19.7272     s = 1.28216

Normality test
Normality Chi2(2)=     0.60181

Testing for Heteroscedastic errors
CHI2(6) = 6.3885  and   F-Form(6, 7) = 0.64188 [0.6973]

V01=C_1          V02=I           V03=I_1
Heteroscedasticity Coefficients:
          Constant         V01          V02          V03        V01^2        V02^2     V03^2
Coeff.      -55.41       2.068      0.06825      -0.1602     -0.01852      0.09448  -0.07807
t-value    -0.5701       0.534       0.1924      -0.4005      -0.4864        1.257    -1.078

RSS = 75.3404     s = 3.28069

RESET test for adding Yhat^2
RESET  F( 1, 13) =      1.2667 [0.2807]
```

Fig. 7.10 PcGive 7 regression display

ingredient in econometrics research, it is far from sufficient for practical empirical modeling. PC-GIVE has been explicitly developed to aid the process of discovering 'good' models by offering a wide range of evaluation tools..."

The regression displays shown here do not convey the full effect of the Hendry approach, and should be regarded as representing merely a capsule view, however seen in the context of the early 1990s, PcGive was one of the few econometric software packages to offer the number and variety of tests shown. The displays shown are exactly as provided by David Hendry, including the omission of the p-value in Fig. 7.11.

2006: PcGive 11

PcGive 11 represents a significant expansion in scope that is not conveyed by Fig. 7.12, which simply indicates some further refinement in the way that test statistics are presented in the context of a regression display. However, during the

```
EQ( 1) Modelling C by OLS
       The present sample is:  1922 to 1941 less 2 forecasts
       The forecast period is: 1940 to 1941

             C =     +2.521          +0.9668 C_1         +1.013 I        -0.785 I_1
 [HCSE]        [     4.147]      [    0.0821]        [     0.208]    [    0.1845]

 R2 = 0.923858  F(3, 14) = 56.622 [0.0000]  s = 1.5033  DW = 1.97
 RSS = 31.6388913 for 4 variables and 18 observations

 Variance instability test: 0.0541511  ;  Joint instability test: 0.899482
 Information Criteria:  SC = 1.20632;  HQ = 1.03574;  FPE = 2.76213
 R2 relative to difference+seasonals =    0.77667

 Forecast Chi(2)/2 = 0.93195
 Chow    F( 2, 14) =    0.51569 [0.6080]

 AR 1- 2F( 2, 12) =    0.23671 [0.7928]
 ARCH 1 F( 1, 12) =    1.5264  [0.2403]
 Normality Chi(2) =    0.60181
 Xi^2    F( 6,  7) =    0.64188 [0.6973]
 RESET   F( 1, 13) =    1.2667  [0.2807]
```

Fig. 7.11 PcGive 7 alternative regression display

```
EQ( 1) Modelling C by OLS (using KLEIN.IN7)
       The estimation sample is: 1922 - 1939

                     Coefficient  Std.Error  t-value  t-prob Part.R^2
C_1                     0.966847    0.08584     11.3   0.000   0.9006
Constant                 2.52066      4.452    0.566   0.580   0.0224
I                        1.01270     0.1564     6.47   0.000   0.7496
I_1                    -0.785030     0.1751    -4.48   0.001   0.5895

sigma                     1.5033  RSS               31.6388913
R^2                     0.923858  F(3,14) =       56.62 [0.000]**
log-likelihood           -30.617  DW                      1.97
no. of observations           18  no. of parameters          4
mean(C)                  53.1833  var(C)               23.0847

1-step (ex post) forecast analysis 1940 - 1941
Parameter constancy forecast tests:
Forecast  Chi^2(2) =   1.8639 [0.3938]
Chow      F(2,14)  =  0.51569 [0.6080]

AR 1-2 test:       F(2,12)   =  0.23671 [0.7928]
ARCH 1-1 test:     F(1,12)   =   1.5264 [0.2403]
Normality test:    Chi^2(2)  =   1.7947 [0.4076]
Hetero test:       F(6,7)    =  0.64188 [0.6973]
Hetero-X test: not enough observations
RESET test:        F(1,13)   =   1.2667 [0.2807]
```

Fig. 7.12 PcGive 11 regression display

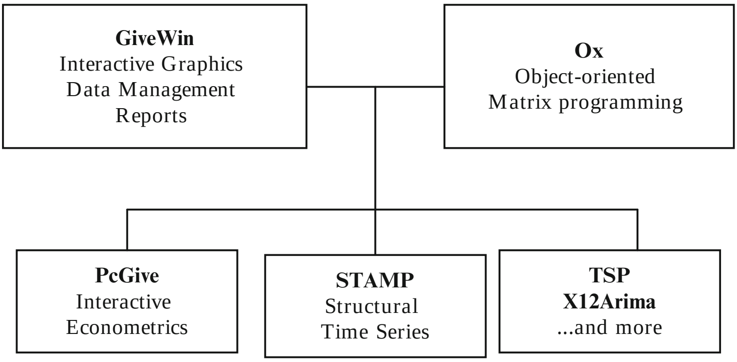


Fig. 7.13 Schematic of the OxMetrics family

14 years between 1992 and 2006, beginning with version 10, PcGive has been integrated into was is now described as the OxMetrics family of programs, a diagram of which is shown as Fig. 7.13. One of the aspects of this change is the development of the so-called GiveWin interface that provides an essentially menu and icon oriented facility that permits a variety of econometric software programs to be executed, including not only PcGive, but also STAMP and other programs, written in the Ox econometric programming language, and even essentially independent programs such as TSP.

Data Management Issues

It was suggested earlier that data acquisition is today software related. This assertion gains substance from the fact that the Internet has steadily become an increasingly important online economic data source, particularly during the past 10 years. As the years have gone by, economic data sets have become progressively more available as downloads from an increasing variety of websites, rather than needing to be keypunched laboriously by individual analysts from the hard copy publications of government agencies, trade associations, international organizations and other such originating data sources. However, this tendency for the data used in economic research to become distributed in machine-readable form actually dates from the 1960s. As mentioned before, economists were early developers of large-scale online data systems, initially containing time series macroeconomic data.

Michael McCracken, who appears to be the first of these, began to create a data base management program for the CDC 1604 in or about 1964, a program called DATABANK (McCracken, 1967a, b). From the first, DATABANK was designed to manage an economic data base of as many as 5,000 time series and some perspective on the ambition of this effort at that time is provided by the fact that the Association for Computing Machinery's journal *Transactions on Data Base Systems* only began to be published in 1976. In addition, in the middle 1960s, Charlotte Boschan and a group of other business economists at New York banks began to create what would soon be known as the NBER Time Series Data Bank (Boschan, 1972), the contents of which subsequently grew to multiple thousands of series during the 1970s.

Almost contemporaneously with these individually instigated efforts, government statistical agencies in Australia, Canada, and the United States, with only a slight lag, officially began to make plans for, and to a degree institute, large-scale online data distribution systems, including CANSIM and LABSTAT (Mendelssohn, 1980; Pilozzi, 1976; Renfro, 1980a, 1997a; Triandafillou, 1975). Starting at the end of the 1960s and extending to the present day, such firms as Chase Econometrics, Data Resources, Haver Analytics, and Wharton Econometric Forecasting Associates (all, except Haver, now merged together as IHS Global Insight), among others worldwide, have made a business of compiling data banks and distributing economic and financial time series online. In addition, other organizations and people, both commercially and as research-inspired products, have created and made available both micro and macro data libraries and economic and financial data sets of various sorts, often originally distributed using tape reels. Notwithstanding its ultimate particular importance as an economic data distribution mechanism, the Internet can be seen as representing the continuation of a long-term multi-faceted development.

The personal perspective of any given economist on the change in the state of the art that this evolution represents will depend upon circumstance, but as a general proposition, there is a starkly obvious difference between today and the 1950s or before, when economists such as Keynes, Kuznets, Meade, Stone, or Tinbergen might each need to compile individually the observations they intended to use for analysis, drawn observation by observation from possibly obscure documents. Exactly how the situation of the 1960s differs from the present may be less immediately evident, for, although any working economist will have experienced the expansion of the Internet during the past 10 years, not everyone will be able to recall that it was only in the later 1960s that government agencies and trade organizations began to produce numbers on anything like today's scale, and even then limited to a comparatively smaller number of relevant (generally post World War II) observations.

In addition, notwithstanding early papers by Guy Orcutt and others (Edwards & Orcutt, 1969; Orcutt, 1952, 1962), microeconomic data sets were comparatively rare even as late as the early 1970s (David, 1980; David & Robbin, 1981; Taeuber & Rockwell, 1982). For example, in the UK, the Social Science Research Council (SSRC) data facility at the University of Essex (now the UK Data Archive) was set up in 1967, whereas the Central Statistics Office was established at the behest of Winston Churchill in 1941. In contrast to those times, economic data are now not only more generally available but, in addition, also relatively highly accessible, in some cases even organized into intelligible, downloadable data sets, formatted in a way that is instantly analytically tractable given the use of an accommodating software package. Economists who have had the good fortune to download and import their data in this tightly coupled, relatively painless way are likely to feel that the ideal has been attained, whatever their historical recall.

However, this good fortune is not the universal experience. Someone working alone who, from a variety of independently created websites, gathers data day by day that pertain to multiple geographic areas, multiple economic concepts, and multiple observational frequencies, and who thus experiences first hand the need to select the nuggets from among large quantities of disparate time series, or else who otherwise often creates data samples as subsets that are extracted from large and varied data sets, is likely to have quite a different perspective. Such an experience contrasts strongly to that of anyone who uses well-organized data samples that are immediately downloaded in their entirety and in a format that is imported directly by his or her chosen software package. Furthermore, this latter type of instant access, or tight coupling, tends to obscure that, in its absence, the economist's ability to determine the characteristics of economic data not only may crucially matter to the particular ways in which they can be used but in many cases this ability can become attenuated, due to information loss in transmission.

In order to assess at least some of the implications, consider initially the fact that individual time series are defined by the geographic area or other economic and social characteristics of the concept they each represent, as well as by the characteristics of the date range of their availability, their observational frequency, and often even the revision cycle of each observation. In somewhat the same fashion, cross-section data samples are made up of variable observations that will have a variety of associated conceptual characteristics. Panel data samples are in addition time dated. When any of these data types are imported into an econometric software program, in the absence of tight coupling, the first general issue that must be faced is that of specifying their particular program-based organizational characteristics. In the first instance, to fix ideas, think of these data as being stored internally in a particular matrix form where the concept-identified variables are ordered by column and their individual observations by row. In the case of time series data, given a selected column, the observations must be located in the appropriate, time-dated, row cells of this column.

Utilizing this sort of matrix storage approach internally in an analytic program, the management differences between time series, cross section and panel data are not so very different, although the rows will not be time-indexed in the case of cross section data. Panel data will of course require the use of a third dimension of the matrix, either explicitly or by "stacking" in two dimensions. But apart from these conceptual differences, it is historically completely natural for the econometrician to think of the data, once loaded for analysis, to be organized into matrices. Furthermore, once this type of organization has been appropriately established, it is easy to think simply in terms of indexed variables and observations and thus to ignore any distinctions between time series and cross section observations. Whenever the data are represented in this form, as will be familiar to anyone who has used SAS, for example, they are quite manageable on much the same terms, whatever their specific type. In each case, the variables and observations can be addressed either by selecting or looping over the relevant indexed columns and rows. Of course this is not to say that the computational operations subsequently performed are necessarily the same irrespective of data type; the focus here is exclusively on pre-analytic data management.

In the context of this type of explicit data matrix representation, it is straightforward to define particular operations symbolically. For example, if variable 1 is identified as nominal Consumption Expenditures and variable 12 as the Personal Consumption Deflator, then real values of the expenditures can be specified using the Fortran-like notation:

$$\mathrm{X(I, 1)/X(I, 12)}$$

where I indexes the corresponding rows for both included variables. In the case of time series data, the observation date range and the observational frequency generally need to be evaluated in combination in order to define the range of this index or to select relevant cells, although in some contexts, such as when all the observations on all the variables will be used, it might be possible to dispense with this side calculation and simply define a counting index, $\mathrm{I} = 1,\ 2, \ldots, \mathrm{N}$, where N specifies a given number of observations common to each variable.

However, it is a data management limitation to have to import the data strictly as a rectangular data set, in much the same fashion as was common in the mid 1960s, in order to avoid dealing with this type of side calculation. An additional aspect of this particular representational approach, in which the data storage matrix is symbolically represented in the calculations, is that the user, effectively in a programmer's role, may in addition finally need to specify also the matrix column location of the evaluated result of such a transformation, if this vector is to be stored for later use in the same matrix. Of course, algebraic operations on the entire **X** matrix can also be considered, such as for the computation of sums of squares and cross products to be used in the calculation of parameter estimates. But notice that, in all cases, essentially what is being done is to use index values in order to be able to identify logically each set of observations, that is, each variable to be used in the analysis. Notice also that in principle the observations can be referenced even if they are not "nicely" organized – at least to the degree that some viable index algorithm can be formulated, possibly in the form of one or more pointer vectors that specify row and column locations. Notwithstanding that it is most intuitively natural to imagine the variables simply as matrix columns, in principle the indices potentially provide organizational flexibility.

Alternatively, these indices can be dispensed with, in favor of a condensed symbolic command notation, such as the familiar textbook-styled:

$$\mathbf{X}'\mathbf{X}$$

provided, of course, that the referenced data are located in a specific way that will permit this more condensed notation to be operationally meaningful algebraically. Generally, the observations will then need to be located in the upper left hand corner of the X matrix, or to completely fill this matrix. Otherwise, once the $\mathbf{X}'\mathbf{X}$ matrix is computed, it would then be necessary to remap particular cells in order to insure that this implied $\mathrm{K} \times \mathrm{K}$ matrix correctly contains the sums of squares and cross products of the K variables, where K should be interpreted as denoting the number of filled columns of **X**. In almost any econometrics textbook, it is a notational convention that **X** would in this case be a $\mathrm{N} \times \mathrm{K}$ matrix, but in the context of computer memory **X** as a storage array may in fact physically exceed these dimensions. These differences reflect the difference between the conventions of textbook notation and the behind-the-scenes housekeeping that will necessarily be associated with any actual attempt to make this type of textbook notation symbolically operative. Notice that the freedom to continue to think, in the context of the computer, in the same way that the textbooks teach is bought here at the price of a particular logical organization of the data once they have been imported.

The issue fundamentally being considered is the correspondence between the internal representation of the data and how the observations might be addressed externally, using a particular program's command language. But so far it has not really been necessary to call attention to this correspondence specifically, for the external references and the internal organization of the relevant matrices have been effectively the same, given their suitable organization. However, an alternative approach to external program control is instead to reference each individual set of observations (that is, the observations that constitute each conceptually identifiable variable) using a symbolic name or label of some sort, which when properly implemented in a free-format algebraic command language then permits the user to direct the program logically, employing a "natural language" notation like:

$$\mathrm{CE} = \mathrm{CE\$}/\mathrm{PDCE}$$

where CE$ should be understood to refer to nominal Consumption Expenditures, PDCE the Consumption Expenditures deflator and CE the resulting real values. Notice that this is a vector-based symbolic command notation.

An important difference, when using this type of notational and logical approach, is that the program must then assume responsibility for determining automatically the locations of the relevant observations in some type of behind-the-scenes workspace or data bank or Memory File that will need to be used for several implied data management operations: to store all imported observations on all variables, to find the observations referenced by CE$ and PDCE, and to locate the resulting observations on the "new" variable, CE, after these have been computationally created. The ability to permit the user to employ an algebraic free format language in much the same way that he or she might otherwise write or type on a sheet of paper, yet to cause calculations to be made, implies the need for the program to perform all the necessary background processing to make the symbolic operations mathematically meaningful, whenever the notation is interpreted exactly as it would be when displayed on a sheet of paper. Among the requirements for the program to be able to do this is that it "understand" all the relevant characteristics of the variables being used.

Obviously, even if the calculations themselves are coded correctly, in order for the program both to perform the calculations reliably behind the scenes and to communicate accurately to the user what it is logically doing, or what it did, a reasonably rich information set of data characteristics will need to be provided when the observations on each of the original variables are imported. In particular, notice that to perform this type of calculation, the program must, for example, correspondingly locate the relevant observations on each of the right-hand-side variables so that they are vector pair wise at the time that the division operation takes place. It must also deal with any missing interior observations appropriately, so that each corresponding resultant observation on CE, after the calculation, is not some bogus number. Whenever the variables involved are time series, it must cope with the possibility that the availability date range for CE$ may not match that for PDCE. Without considering in great detail the need for such protections, it is already evident that if a program is to take full responsibility for effectively managing the data internally, it is often first necessary to import all the observations on all the variables at the outset, make any and all subsequent revisions to these variables as members of the program's internal data base, and then, later, during any data related operations, to set the date range and related use characteristics; but in this case, the program obviously must "know" the dates of availability for each variable and often the individual observation frequencies.

It is demonstrably possible to design a program to keep track of all this information. However, initially, the basic definitional information must be provided whenever observations are imported. Additional complexity is also possible: for instance, some packages allow frequency conversions to be made on the fly, and in this case the individual observation frequencies of each variable obviously must be "known" by the program. It is evident from such considerations that, as a rule, the more sophisticated the internal data base management facilities of a software package, the more important it is to associate with each distinct set of variable observations the particular characteristics of that variable and those observational values. Of course, whenever variable transformations are made, additional variables will usually be created, which will then themselves need to be stored and managed, either temporarily or more or less permanently, possibly from one program use session to another.

One of the reasons to address in this way both the characteristics of the data and various logical issues of data base management is in order to bring out clearly the implied data import requirements that will be associated with particular types of program command language structures. Each of the several ways of representing algebraic operations that have been considered here are exhibited by one or more of the existing econometric software packages, with particular packages supporting more than one type of representation. Recall also that there is no requirement for a particular command language representation to bear a one-to-one correspondence either to the specific way that a program will organize its data storage, the algorithmic way that it performs calculations, or the data importation methods it might employ. Therefore it is not appropriate to make inferences about the internal architecture of a software package simply from its external command language representation. However, it is nonetheless possible to design a program so that the user can operate it as if the command language representation controls its operation in a seemingly isomorphic manner. Whenever this happens, the first type of language structure, featuring indexed matrices, allows the user, without consciously thinking about the process, to manage sets of conceptually related variables using matrix notation, and will permit individual cell-level operations to be performed at the user's specific command. The second type of language structure, in contrast, generally will establish requirements for the particular way that observations are automatically located in matrix cells so as to seemingly correspond with the notational conventions used in the econometrics literature. The third form of language structure makes explicit the operations that might be performed on named economic variables. As has been discussed, each of these forms tends to imply a level of information requirement about the characteristics of the economic data that are used, or else implicitly mandates their logical organization in a particular, possibly more or less rigid fashion. The purpose here is not to judge any command language form to be best, but rather to illustrate some of their implicit information requirements.

Of course, some econometric software packages can be operated using menus and icons and a package could be designed so as to be a modern looking analogue of a 1960s package. The classic 1960s package required the data to be imported in a specific card image fashion, usually a type of matrix representation. Generally variables were specified positionally. The operations performed were strictly limited. Modern packages generalize the possible operations and provide greater flexibility in use, essentially by providing some form of language interface, examples of which have just been considered. It is in the nature of econometric operations that users will wish to specify particular variables to be used in particular ways, so that today, even when a software package is operated in the style of a classic "windows" program, provision will still usually be made for the user to specify transformations and other operations using some type of algebraic natural language. As indicated earlier, the user of a modern econometric software package is buffered by the package from the computer's operating system, but even at one remove the user must still manage the way the program operates using the imported data.

It is highly relevant to this consideration of data importation that it has become increasingly common for government agencies and other originating data sources to supply online data in the context of Excel spreadsheet files, or at least Internet resident files that generally obey the conventions of these spreadsheet files, as well as by adopting particular, related browser interface structures, each of which data storage or transfer representations can affect the ability of economists to quickly download observations from the Internet and make them immediately usable in existing econometric software packages. In a number of cases, an economist can choose either to download an explicit Excel formatted file of the XLS type or else to exhibit its contents in a browser display that has many of the essential characteristics of an Excel worksheet display, including such features as the ability either to cut and paste observations, singly or in groups, or to otherwise transfer them directly to his or her desktop. Although these files and browser displays are becoming increasingly more XML friendly from one version of Excel to the next, which potentially has favorable implications for data downloads (Harrison & Renfro, 2004), whenever files and displays of this type are used for this purpose they nevertheless potentially pose problems, precisely because they generally suppress important content characteristics of the observations transferred. Ordinarily, the observations are transmitted in what is essentially a matrix format, with perhaps some descriptive information about the relevant economic variables, but not necessarily in a form that permits this critical information to be imported automatically. Furthermore, even when this information is immediately human intelligible, the variations in the precise way it is displayed makes its content often difficult to capture automatically on import. In addition, it is not necessarily true that these files' observation matrix formats will themselves be compatible with either the conventions of econometric textbook characterizations or other econometrically natural forms or formats. Therefore, particularly for certain downloaded data sets, a large amount of user editing may be required before the observations they contain can be imported reliably into any given econometric software package, together with program-usable information about the variables' characteristics.

As discussed, time series are not simply sets of observations, but have a frequency of observation, a range of availability, and a conceptual definition that in some way must be conveyed to the data-receiving program. Among other things, the extended conceptual definition can include its units of measurement and last revision date, and even the constructive formula for a series that relates it to other series, all of which information may be analytically pertinent. Similarly, cross-section and panel data observations constitute observations on particular variables and the specific characteristics of these variables have obvious relevance to the analyses performed. The inference that should be drawn from a consideration of these properties of the data is that the use of data transfer protocols of the type just considered generally imposes the requirement, each time a new set of observations is transferred, for some person to at least stand ready to edit the data, and in many cases to actually follow through, lest the specific supplementing definitional information becomes separated from the observations. One of the implications of this extra work, particularly when multiplied by the number of economists doing applied research, is a disincentive both in the aggregate and for the individual to make the best use of the data available from the various originating data sources.

Of course, individual econometric software developers are at liberty to react by designing their packages so as to be closely compatible with the download conventions established by certain originating data sources, and in a number of cases this has been done. However, the degree of protocol consistency among originating data sources is not great. For US data, for example, there are significant differences between the operating characteristics of the websites of the Bureau of Economic Analysis, the Bureau of Labor Statistics, and the Board of Governors of the Federal Reserve System. International protocol comparisons would reveal more variety, not limited to human language differences. Therefore, unless the developer of each econometric software package were to choose to match the individual conventions of each possible originating data source, a costly proposition, the result may increasingly be to foster selectivity and enforce research balkanization. Furthermore, it also needs to be borne in mind that, inasmuch as website conventions change with some frequency, it is not possible simply to adopt the conventions of a particular site once and for all. Therefore, even selective compatibility implies a treadmill for the econometric software developer. Of course, one of the pernicious implications of such selectivity is the possible creation of incentives for economists to use data disproportionately obtained from only certain originating sources.

However, there is also another underlying technological aspect that needs to be recognized: to its users, the browser effectively represents a "window" on the Internet, but as a piece of software it is nevertheless a desktop construct; that is, local, personal computer software. Because of this specific characteristic, and because of the prior widespread use of spreadsheet software since at least the early 1980s, from the Microsoft perspective (or that of any browser developer), it was natural for these two computer technologies to be made to interconnect, especially as this also involved a seemingly natural convergence of community interests. For spreadsheet users, a browser display that mimickes Excel may be desirable, as it might also be for at least some web page developers. At the same time, from the perspective of a government statistical agency or other data purveyor, the browser table display is itself an obvious extension of the printed table, the still-remembered previous data distribution technology. In addition, from the perspective of a data purveyor, a further attraction is that this choice of transfer mechanism also removes any requirement to consider the secondary use of the data: simply make an Internet-resident table and let the final user be responsible for its disposition.

Unfortunately, as should be evident from the earlier discussion, because of its summary nature, a table – that is, a matrix – also tends to make extraneous the supplementary information used to form it. Make no mistake: on the side of the data publisher, the information obviously exists to produce the "table." Economic statisticians fully understand the constructive characteristics of the data that they collect, compile, and publish, so that pouring the observations into a table and adding some descriptive garnish is easy to do. Simply as a fixed object, this table may admirably serve its display purpose. The problem is disassembly, which is a process of adding back the requisite information about the individual observations it displays, in many cases the same information that was used to create the table. Notice that, in addition, the more complex the table, the more detailed the disassembling "instructions" may need to be. Data publishers have all this information, and "in house" they utilize it. The problem is to permit this information to be shared effectively with the ultimate users of the data.

Why this particular thought is important is because, for a certain type of economic data, namely time series, a generalized, open-source content-rich easily-adopted data exchange protocol already exists, and predates the modern development of web pages. In a particular form, known as TSD (Time Series Data), it has been used for more than 25 years by a number of organizations and has been increasingly implemented in the existing econometric software packages (Harrison & Renfro, 2004). This protocol has been implemented stably over time, to the degree that a file created in 1986 can still be read today, and, although modern extensions have been proposed, certain of which are XML related in form, these extensions can all be effected in a relatively highly backwardly-compatible fashion. Once established they can be adopted easily by econometric software and website developers alike. These protocols, including the extensions, have the property of permitting observations and associated descriptive information to be transferred directly from the originating data sources to the economist performing research, without the need for data editing.

A further point also needs to be made, which is that the critical consideration is not necessarily the tabular, or matrix, nature of the existing data distributors websites. What is critical is *information content*. The cells of a spreadsheet or website table could actually be used to distribute a complete set of descriptive information about an economic variable and if *standards* could be established for the precise form and format of this information, as well as the particular format of the observations, there is no reason why an Internet-resident table cannot be used as an effective data distribution mechanism. Such standards could even include a variable key to automate reads of the contents of any given table. The fundamental problem is the lack of content-rich data distribution standards between original data sources and the end user.

If the attention of economists is sufficiently engaged, to the point of the creation of a lobby group, it is a relatively small matter for government statistical agencies and other online economic data publishers to modify their websites to support particular standards and specific data protocols. Obviously, the general implementation of content-rich economic data exchange protocols would make it possible for economists to both download and exchange observations without any need for extra editing. Inasmuch as the effect would be to permit existing information to be processed efficiently, these protocols do not per se imply additional distribution costs once implemented, but instead a substantial reduction in comparison with today's aggregate cost. Yet the inference to be drawn is not necessarily that there should be immediate wholesale adoption of any existing protocols, but rather the need in the first instance for economists to recognize that a problem exists and for its possible solution to be considered widely. If enough attention is paid, it may even be possible to create protocols for each of the types of economic data. If instead economists continue to jointly and severally adopt the ostrich posture, it is highly likely that the website protocols of the originating data source organizations around the world will progressively become individually more various, even tortuous.

The implications of the appropriate provision for the edit-free downloading and transfer of economic data files actually go beyond the mechanistic issues of data exchange. Years of obtaining National Income and Product Accounts data in a particular form should not be allowed to obscure the fact that, in the context of the distributional promise of the Internet, it would in principle be possible for economists to obtain different representations of the data that are each designed to support specific research needs. For instance, there are particular data provision initiatives (Anderson, 2006) under way that aim to provide economists with economic datasets that have a particular revision dating, for example, so that it would be possible to obtain observations representing the US National Income and Product Accounts as of particular release dates, such as at the end of January 2006, or February 2006, or March 2006. Similarly, it is in principle possible to construct a dataset containing observations that for each observational time period are second-month-after-the-end-of-the-quarter revisions. For many people, it may well be true that the most useful US National Income and Product Accounts dataset is the one that contains the most recent revision for each time series observation included, but it is easy to imagine research projects that could involve the use of observations with a different revision dating (Croushore & Stark, 2000; Howrey, 1978, 1996).

More generally, it also needs to be recognized that Income and Product Accounts, Flow of Funds Accounts, and other such data compilations can be presented in a wide variety of ways, not just the way they are presented by a given tabular display at a particular point in time. A particularly good example to consider is international transactions. For example, subsidiary accounts can be established specifically in order to portray either cross-border trade or trade between residents and non-residents of a particular country or in terms of foreign and domestic ownership (Landefeld, Whichard, & Lowe, 1993). Furthermore, accounts can be structured in a variety of ways, including those that provide satellite accounts (Carson, Grimm, & Moylan, 1994; Carson & Landefeld, 1994a, b; Eisner, 1989; 1995; Popkin, 1993; Renfro, 1998; Triplett, 1993). In the old days, when the data transfer mechanism took the form of printed documents displaying tables of data, it would have been too ambitious to consider radically alternative presentations of these data. Today and in the future, the Internet potentially provides the capability to present data in what can be called an *accounts independent* form (Renfro, 1997a). Similarly, micro datasets can potentially be provided to users in a way that exposes a much greater degree of research opportunity for the analyst. This is not the place to discuss these matters further; however, they have been addressed simply to convey that good economic data exchange standards are of much greater general importance than might be inferred from the earlier discussion alone.

The issues associated with the management of economic data have generally received less attention from both economists generally and many developers of econometric software packages than have other aspects of software design. There have been only a few attempts (Renfro, 1980a, 1997a) over the years to consider various aspects of economic data base management, including both time series data base management and relational data base systems as these might apply to economic data bases (David, 1980, 1989; Kendrick, 1982). As mentioned in passing earlier, economic consulting firms and economic data vendors have actively developed systems and mechanisms for large-scale data management, distribution and point-to-point transfer. In this context, as indicated, content rich economic data exchange formats have been created and adopted quite generally, but in contrast it is characteristic of the econometric software packages most likely to be used by academic economists that they more commonly provide minimal generalized data management. Reflecting the focus of econometric textbooks on particular econometric techniques, much of the development of econometric software has disproportionately focused on these techniques. It can be argued that, ultimately, all econometric software developers would do well to seek a greater degree of balance in the software facilities they offer.

A reason to consider such matters here is of course that the existence of disincentives to the use of available economic data can be expected to adversely affect the progress of economic research. There are certain identifiable macroeconomic datasets, for example, that have been relatively intensively used and it may be that some of this use, relative to the use of other datasets more difficult to import, or even to obtain, may reflect not only fashion but also possible data management limitations of econometric software packages. Obviously, what software developers concentrate on will in part reflect their own particular interests, for as mentioned earlier, these economists as a group tend not to be primarily motivated by commercial aspirations. At the same time, however, the characteristics of the existing packages reflect also the interests and concerns of economists generally. The facilities that these programs provide will, over time, tend to be motivated by their expected degree of use. Consequently, the future development of econometric software will depend not only upon the interests of those economists who develop this software, but those of economists in general.

Chapter 8
The Implications of the Findings

The present survey has already been criticized, before its completion, as over-narrowly focused, even within its own terms. The argument is that the common use of econometric programming languages during the past approximately 20 years, such as Gauss and Ox (Hill, 1989; Knusel, 1995; Renfro, 2004b, c), among others, has allowed economists to implement diagnostic tests in their day-to-day work, lessening their dependence on packages such as those surveyed here. Consequently, since the sample fails to properly address this possibility, the results are biased. In response to the narrow focus aspect of this indictment, the only defensible answer is "guilty as charged," if only in a limited sense. Clearly there is considerable scope, excluding the surveyed packages, for econometricians and applied economists to implement a wide range of tests, both now and in the past. Without any doubt, tests have often been implemented that are not described here, but would be relevant to the subject of this survey. If it were also the long-standing disciplinary convention, strong and invariably-honored, for each economist to publish his or her empirical results fully and carefully – or alternatively make them available on a web page or in some other public context in such a way that an interested later investigator could easily by replication determine exactly how to evaluate the computational process involved, then this guilty plea might be cause for chagrin and embarrassment. However, the unscientific nature of the way in which empirical results are much too commonly presented in the economics literature, with the calculations hidden, the particular observational values unidentified – and possibly unavailable publicly – among other problems, leading to well-known replication difficulties (Anderson, Greene, McCullough, & Vinod, 2007; Dewald, Thursby, & Anderson, 1986; McCullough & Vinod, 2003), makes it impossible to give immediate credence to the large majority of the implemented diagnostic tests that are not considered in the present study, which is a reason they have been ignored. This is not a justification per se, but only an explanation.

However, there is also a strong justification, which is that a critically important issue addressed by this survey is the learning process that is associated with both the implementation and further development of econometric methodology generally, and diagnostic tests specifically. For there to be progress, other than by chance, what should be done at each stage of development and implementation is to make each aspect public in a way that can be evaluated easily. It can be imagined that,

C.G. Renfro, *The Practice of Econometric Theory*, Advanced Studies in Theoretical and Applied Econometrics 44, DOI 10.1007/978-3-540-75571-5_8,

during the development of each of the software packages surveyed, someone – perhaps several people – would necessarily have been involved in the initial implementation of each of the test statistics. Once implemented, some attempt would then have been made to validate this implementation. However, it is not until this implementation has *both been made and evaluated in a public forum* that the general learning process begins. Implementations made privately and used privately (to make an inference) might in a particular case have an impact on the progress of economic knowledge, but this use does not provide any information to the profession about the way in which diagnostic tests should be implemented, how they are implemented, or in what way that implementation affects the progress of applied research. It is only to the degree that those economists who have used Gauss, Ox, or any other programming language come forward and present their results on the same basis as those presented here that these private results have any relevance to the learning process or to the conduct of the present survey. To consider the present survey only in terms of the information presented in this volume is too narrow a view of its purpose. The evidentiary process of which this survey is a part should be ongoing. There is no reason to stop presenting relevant results now.

Very early in the survey, confirmation of the need for it became obvious to the participants. The experience of those who have taken part most actively is that, whenever values generated by different packages for corresponding statistics were found to differ, it was at first sometimes surprisingly difficult to determine what in particular had been done by each developer – and therefore to explain exactly why these differences occurred. To a degree, differences occurred as a result of outright coding mistakes. That such mistakes might have been made is quite easily understood, and indeed to be expected, for the simple truth is that a single error in a single source code statement, out of many (perhaps a million or more in some cases), is sufficient to cause one program to generate a different result than another.

However, considered more reflectively there is a gray area between "correct" and "error." One example of a gray area result occurs as a consequence of the particular, somewhat eclectic names that econometric software developers have given to test statistics their programs display. For instance, as was demonstrated in Chap. 5, in the case of a statistic that is intended to provide the means to test against hetroscedasticity, commonly identified as the Bruesch-Pagan test statistic, the value displayed by a given econometric package can have any of several possible constructive characteristics. Depending on which program is used the value displayed might correspond algorithmically to a Bruesch-Pagan test (original form) or the so-called Koenker-Bassett variant of this test, or the White test variant. As discussed in that chapter, the numeric value that appears can depend upon whether the test value is computed as ESS/2, which is of course a χ^2 statistic; or as TR^2, also a χ^2 statistic; or as a joint parameter F-statistic; or even which regressors are used; or some combination of these circumstances.

However, what also needs to be generally recognized when considering this example is that the issue at stake concerns both the degree of behind-the-scenes computational detail and the usual paucity of information on the surface, when only a name is used to identify a number. To think otherwise is to be oblivious to the way in which econometrics has developed during the past nearly 80 years. The practitioner, staring at a printout or a screen, often may have nothing in front of him or her other than a single number to which some ostensibly clarifying name is attached and originally this nomenclature might have reflected that, in the beginning, the program was created for the use of one person or perhaps a closely affiliated group of people. Relatively few econometric software packages began as the result of a corporate effort with a substantial development budget and a well-funded marketing plan. Even in the case of those that did, display space on a screen is still limited, tending to force simple descriptions. Furthermore, even textbooks and journal articles vary in the names applied to all but the most established statistical tests.

The fundamental issue here is the communication of information. There is an understandable tendency for econometric theorists to split hairs, even to the point of making distinctions that can appear to have little practical relevance, when seen from the perspective of those who either implement or apply particular tests represented by one or more test statistics. However, an implementation failing of equal importance is when practitioners are not given sufficient information to make an informed inference. To a degree, the ability to make such inferences is a matter of transparency, namely the undistorted full transmission of information from an original source to the user. It is all too easy for the econometric software developer to become a technological fan-dancer, offering enticing glimpses amidst smoke and mirrors, yet hiding the essential truth from the spectator. The problem is how to avoid this opacity. It can be avoided if developers make the effort to provide clear descriptions of the characteristics of the displays they provide and if those who teach communicate to their students precisely what the statistics mean.

Perhaps the most significant finding of this survey is that, with the best will in the world, those who have set out to develop software to be used by the profession at large have produced a body of work that, objectively speaking, can be criticized both for some degree of inaccurate description and for its lack of transparency to the users of this software. This survey stands in testament to the fact that, whenever the diagnostic statistics implemented are constructively equivalent each developer has apparently obtained comparable results, at least for the cases investigated here. That is, the survey results suggest that when econometricians try to do the same thing, they succeed. This is the good news, for although it might have been expected, this expectation was not necessarily a priori justified. However, in the absence of this survey, it would not be true – has not been true – that each interested user of these software packages would have immediately be able to determine exactly how to interpret the statistics displayed. It is not enough to know that a particular value represents a χ^2 or an F-test value or other such evaluative quantification. In order for it to be possible to make knowingly an appropriate, if conditional, inference, it is clearly necessary to know the full facts of the implementation. The natural gulf between econometric theory and practice is too wide to be able to presume it to be of necessity comfortably spanned at the first attempt.

The non-transparency revealed by the survey can be regarded as representing a past failure on the part of the individual software developers, but it can also be regarded as a collective disciplinary failure. Over the years, because of a lack of information, it has often been difficult for individual econometricians who develop software, or even those who write textbooks, to determine on their own how what one developer has done compares with that done by another. Furthermore, there has been little individual incentive to overcome these difficulties. After all, econometric software development is a competitive enterprise, so that a developer may not always feel able to get in touch with another to inquire about a program's characteristics. As a consequence, what has apparently occurred, at least sometimes, is that individual developers have implemented tests that have represented their best judgment, but which might also be open to misinterpretation by software users. Others who may also have been misled include econometric theorists.

There are of course a few instances, such as in the case of the 1966 paper by Zellner and Thornber (1966), that econometricians have taken the initiative to publish evaluative information about the computational aspects of their work, but it has been *far* more common not to. One of the reasons, of course, has been not only the absence of any requirement to do this – for instance, as a condition of publication of some proposed new technique – but also the aforementioned outright reluctance on the part of journal editors to publish any empirical confirmation that a developer might proffer. In many respects, and on several levels, economists have failed to establish adequate ground rules and sufficient laboratory training for published applied research, especially as compared to the hard sciences or medicine. The price of this sin is ignorance and the result is a messy situation in which it is difficult for anyone who reads almost any article published in the journals that involves data or computation to give immediate credence to what is said. It is really not at all surprising that economists appear to prefer not to debate the facts but instead the theory, do not examine the reality of the assumptions made but instead investigate the logic of the proofs.

However, it is necessary to be careful when asserting that it is difficult to publishing articles about computing and the details of the research process in the economics and econometrics literature, for in the strictest sense this is not true. Such articles are often difficult to write, but once written they are actually easy to publish. Journals such as the *Journal of Econometrics* and the *Journal of Economic and Business Statistics* have historically been receptive to articles about software and the latter, in particular, to articles about data. The *Journal of Economic and Social Measurement* often publishes such articles, as does also the *International Journal of Forecasting*, *Computational Statistics and Data Analysis*, and *The American Statistician*. Furthermore, these are not the only accommodative journals. It is not difficult to reach an audience that essentially consists of software developers, statisticians, and econometricians. What may be difficult, as Hendry and others have previously suggested – and may be becoming more so – is to found a modern academic career on publications about software and data. Among those who have made notable contributions to the development of econometric software are many who currently hold academic appointments but few, if any, who owe their appointments either to their development of software or publications concerning this work. And yet the ability of the economist to comprehend and evaluate the actual activities of economic agents depends upon it, now and in the future.

There are few images more evocative of economic reasoning than that of a downward sloping demand curve intersected by an upward sloping supply curve. This composite construct depicts that quantity demanded decreases with an increase in price, and that supply increases. More than 80 years ago, Marshall (Marshall, 1920, Chap. 4, p. 86) rationalized the first result by appeal to the "universal law" that "a person's desire for a commodity ... diminishes, other things being equal, with every increase in his supply of that commodity." More recently, mathematical logic, resting upon a set of seemingly axiomatic assumptions, has provided a more intellectually satisfactory explanation for this demand behavior. However, it is not difficult to discover exceptional cases that threaten to subvert this ideal, such as the demand for luxury goods or for inferior goods. It is also possible to raise questions concerning the behavioral validity of the particular underlying assumptions. On the supply side, computer software provides an example of a good the marginal cost of which, although falling initially, does not then necessarily increase at higher levels of production, except perhaps at some extraordinary amount. But what is perhaps most interesting about the discovery of such "abnormalities" is their empirical provenance (Samuelson, 1984; Spiegel, 1994), in contrast to the "normal" case, which has characteristically arisen as a deductive consequence of the careful, on the face of it, seemingly axiom-based advance of economic theory. On reflection, the inference that might be drawn is that the word "normal," although sanctioned by usage historically, may not always be the best one to apply a priori to what, at least on occasion, could turn out be a special, even somewhat unusual case. It is not the purpose of this monograph to examine the way in which fact, in the form of empirical observation, might confront ideal theory. But it is useful to recognize that at least some of the findings it presents demonstrate that the complexity of economic phenomena, in conjunction with the difficulty associated with historical attempts to frame discriminating tests of theories, may explain both the attraction of the ideal for economists and the prevailing tendency to establish as standards exemplars that may not be quite as usual as their theoretical prominence would seem to imply.

Appendix A
Version Information for the Surveyed Econometric Software Packages

AREMOS	Version 5.3, Released 1 January 2003
AutoBox	Version 6.0, Released 1 October 2008
B34S	Version 8.11, Released 1 January 2008
Betahat	Version 6.0, Released 21 July 2005
EasyReg	Version 2.0, 25 February 2008
EViews	Version 6, Released March 2007
FP	Released August 2003
GaussX	Version 9.0, Released 1 June 2008
Gretl	Version 1.7.9, Released 28 September 2008
LIMDEP	Version 9.0, Released 1 January 2007
MicroFit	Version 4.11, Released 1 January 2004
Modeleasy^{+}	Version 6.0, Released 1 October 2008
MODLER	Version 11.1, Released 15 July 2008
NLOGIT	Version 4.0, Released 1 January 2007
PcGive	Version 12.1, Released October 2007
RATS	Version 7.1, Released August 2008
REG-X	Version 98.1, Released 1 June 1998
SHAZAM	Version 10.0, Released April 2008
SORITEC	Version 7.1, Released March 2006
Stata	Version 10.1, Released 11 August 2008
TROLL	Version 2.14, Released August 2008
TSP	Version 5.0, 20 February 2008
TSM 4	Version 4.28, Released 20 November 2008

Appendix B
Grunfeld's Investment Theory: Time Series Data on General Electric and Westinghouse (Theil, 1971, Table 7.1, p. 296)

	CGE(−1)	CW(−1)	FGE(−1)	FW(−1)	IGE	IW
1935	97.8	1.8	1,170.6	191.5	33.1	12.9
1936	104.4	0.8	2,015.8	516.0	45.0	25.9
1937	118.0	7.4	2,803.3	729.0	77.2	35.0
1938	156.2	18.1	2,039.7	560.4	44.6	22.9
1939	172.6	23.5	2,256.2	519.9	48.1	18.8
1940	186.6	26.5	2,132.2	628.5	74.4	28.6
1941	220.9	36.2	1,834.1	537.1	113.0	48.5
1942	287.8	60.8	1,588.0	561.2	91.9	43.3
1943	319.9	84.4	1,749.4	617.2	61.3	37.0
1944	321.3	91.2	1,687.2	626.7	56.8	37.8
1945	319.6	92.4	2,007.7	737.2	93.6	39.3
1946	346.0	86.0	2,208.3	760.5	159.9	53.5
1947	456.4	111.1	1,656.7	581.4	147.2	55.6
1948	543.4	130.6	1,604.4	662.3	146.3	49.6
1949	618.3	141.8	1,431.8	583.8	98.3	32.0
1950	647.4	136.7	1,610.5	635.2	93.5	32.2
1951	671.3	129.7	1,819.4	723.8	135.2	54.4
1952	726.1	145.5	2,079.7	864.1	157.3	71.8
1953	800.3	174.8	2,871.6	1,193.5	179.5	90.1
1954	888.9	213.5	2,759.9	1,188.9	189.6	68.6

Glossary

CGE	Capital Stock, General Electric
CW	Capital Stock, Westinghouse
FGE	Market Value of Outstanding Stock, General Electric
FW	Market Value of Outstanding Stock, Westinghouse
IGE	Investment, General Electric
IW	Investment, Westinghouse

Appendix C
U.S. Quarterly Macroeconomic Data Originally Used by Cecchetti & Rich (Greene, 2008; Cecchetti & Rich, 2001)

Year	qtr	GDP	realcons	realinvs	realgovt	realdpi	cpi_u	M1	tbilrate	unemp	pop	infl	realint
1950	1	1,610.5	1,058.9	198.1	361	1,186.1	70.6	110.2	1.12	6.4	149.461	0	0
1950	2	1,658.8	1,075.9	220.4	366.4	1,178.1	71.4	111.75	1.17	5.6	150.26	4.5071	−3.3404
1950	3	1,723	1,131	239.7	359.6	1,196.5	73.2	112.95	1.23	4.6	151.064	9.959	−8.729
1950	4	1,753.9	1,097.6	271.8	382.5	1,210	74.9	113.93	1.35	4.2	151.871	9.1834	−7.8301
1951	1	1,773.5	1,122.8	242.9	421.9	1,207.9	77.3	115.08	1.4	3.5	152.393	12.616	−11.216
1951	2	1,803.7	1,091.4	249.2	480.1	1,225.8	77.6	116.19	1.53	3.1	152.917	1.5494	−0.0161
1951	3	1,839.8	1,103.9	230.1	534.2	1,235.8	78.2	117.76	1.63	3.2	153.443	3.0809	−1.4542
1951	4	1,843.3	1,110.5	210.6	563.7	1,238.5	79.3	119.89	1.65	3.4	153.97	5.5874	−3.9374
1952	1	1,864.7	1,113.6	215.6	584.8	1,238.5	78.8	121.31	1.64	3.1	154.566	−2.5301	4.1701
1952	2	1,866.2	1,135.1	197.7	604.4	1,252	79.4	122.37	1.68	3	155.165	3.0341	−1.3575
1952	3	1,878	1,140.4	207.8	610.5	1,276.1	80	123.64	1.83	3.2	155.766	3.0113	−1.1813
1952	4	1,940.2	1,180.5	223.3	620.8	1,300.5	80	124.72	1.92	2.8	156.369	0	1.9233
1953	1	1,976	1,194.9	227.5	641.2	1,317.5	79.6	125.33	2.05	2.7	157.009	−2.005	4.0517
1953	2	1,992.2	1,202.5	228.5	655.9	1,336.3	80.2	126.05	2.2	2.6	157.652	3.0038	−0.8004
1953	3	1,979.5	1,199.8	222.8	647.6	1,330.2	80.7	126.22	2.02	2.7	158.298	2.486	−0.4627
1953	4	1,947.8	1,191.8	205	645.4	1,325.9	80.5	126.37	1.49	3.7	158.946	−0.9926	2.4792
1954	1	1,938.1	1,196.2	203.4	627.1	1,330.3	80.5	126.54	1.08	5.3	159.675	0	1.08
1954	2	1,941	1,211.3	203	606.1	1,327.9	80.7	127.18	0.81	5.8	160.407	0.9926	−0.1792
1954	3	1,962	1,227.3	213.3	591.2	1,344.2	80.4	128.38	0.87	6	161.142	−1.4898	2.3598
1954	4	2,000.9	1,252.6	223.3	587.4	1,373.6	80.1	129.72	1.04	5.3	161.881	−1.4953	2.532
1955	1	2,058.1	1,280.1	247.2	586.4	1,392.7	80.1	131.07	1.26	4.7	162.669	0	1.2567
1955	2	2,091	1,304.3	262.8	579.9	1,423.3	80.1	131.88	1.51	4.4	163.462	0	1.5133
1955	3	2,118.9	1,320.3	266.4	584	1,451.1	80.5	132.4	1.86	4.1	164.258	1.9925	−0.1292
1955	4	2,130.1	1,336.7	272	571.3	1,468.1	80.4	132.64	2.35	4.2	165.058	−0.4972	2.8439
1956	1	2,121	1,339.2	262.9	570.9	1,480.9	80.4	133.11	2.38	4	165.808	0	2.38
1956	2	2,137.7	1,343.7	260	582.6	1,497.8	81.4	133.38	2.6	4.2	166.561	4.9444	−2.3478

(continued)

(continued)

Year	qtr	GDP	realcons	realinvs	realgovt	realdpi	cpi_u	M1	tbilrate	unemp	pop	infl	realint
1956	3	2,135.3	1,346.8	257.1	577.3	1,504.1	82	133.48	2.6	4.1	167.318	2.9376	−0.3409
1956	4	2,170.4	1,365.3	254.4	592.5	1,526.5	82.7	134.09	3.06	4.1	168.078	3.4001	−0.3368
1957	1	2,182.7	1,374.2	250	604	1,527.5	83.3	134.29	3.17	3.9	168.848	2.8916	0.2784
1957	2	2,177.7	1,376.5	249.9	600.6	1,538.6	84.3	134.36	3.16	4.1	169.621	4.7733	−1.6167
1957	3	2,198.9	1,387.7	255.6	605.5	1,548.7	84.9	134.26	3.38	4.2	170.398	2.8369	0.5431
1957	4	2,176	1,388.8	234.1	616.6	1,543.1	85.2	133.48	3.34	4.9	171.178	1.4109	1.9324
1958	1	2,117.4	1,370.1	216.7	609.6	1,524.7	86.4	133.72	1.84	6.3	171.917	5.5945	−3.7578
1958	2	2,129.7	1,380.9	211.3	625	1,534.1	86.7	135.22	1.02	7.4	172.659	1.3865	−0.3665
1958	3	2,177.5	1,402.3	228.4	628.4	1,568.1	86.7	136.64	1.71	7.3	173.404	0	1.71
1958	4	2,226.5	1,418.8	249.6	641.5	1,588	86.7	138.48	2.79	6.4	174.153	0	2.7867
1959	1	2,273	1,445.2	263	651.5	1,599.5	86.7	139.7	2.8	5.8	174.894	0	2.8
1959	2	2,332.4	1,468.2	286.2	663.9	1,629.6	87.3	141.2	3.02	5.1	175.638	2.7586	0.2614
1959	3	2,331.4	1,483.8	266.6	668.1	1,627	87.7	141	3.53	5.3	176.385	1.8286	1.7048
1959	4	2,339.1	1,485.6	275.6	662.2	1,639.2	88	140	4.3	5.6	177.136	1.366	2.934
1960	1	2,391	1,499.2	305.3	648.8	1,657.7	88	139.8	3.95	5.1	177.841	0	3.9467
1960	2	2,379.2	1,518.1	274	657.4	1,666.5	88.7	139.6	3.09	5.2	178.548	3.1692	−0.0759
1960	3	2,383.6	1,512.1	272.4	665.9	1,667.7	88.8	141.2	2.39	5.5	178.259	0.4507	1.9426
1960	4	2,352.9	1,513.5	239.5	673.1	1,667.2	89.3	140.7	2.36	6.3	179.972	2.2459	0.1174
1961	1	2,366.5	1,512.8	245	680.4	1,680.6	89.3	141.9	2.38	6.8	180.718	0	2.3767
1961	2	2,410.8	1,535.2	263.3	687.2	1,705.4	89.4	142.9	2.33	7	181.468	0.4477	1.879
1961	3	2,450.4	1,542.9	285.5	694	1,729.4	89.9	143.8	2.32	6.8	182.22	2.2309	0.0924
1961	4	2,500.4	1,574.2	290.2	711.1	1,764.4	89.9	145.2	2.48	6.2	182.976	0	2.4767
1962	1	2,544	1,590.6	307.3	723.4	1,777.9	90.3	146	2.74	5.6	183.663	1.7758	0.9675
1962	2	2,571.5	1,609.9	304.5	731.7	1,799.3	90.5	146.6	2.72	5.5	184.352	0.885	1.835
1962	3	2,596.8	1,622.9	310	740.8	1,811.4	91.2	146.3	2.86	5.6	185.044	3.082	−0.2253
1962	4	2,603.3	1,645.9	299.5	744.2	1,825.5	91	147.8	2.8	5.5	185.739	−0.8782	3.6815
1963	1	2,634.1	1,657.1	315.4	740	1,838.9	91.3	149.2	2.91	5.8	186.409	1.3165	1.5935
1963	2	2,668.4	1,673	320.8	744.3	1,857.2	91.7	150.4	2.94	5.7	187.082	1.7486	1.1947

(continued)

Year	qtr	GDP	realcons	realinvs	realgovt	realdpi	cpi_u	M1	tbilrate	unemp	pop	infl	realint
1963	3	2,719.6	1,695.7	331.5	765.9	1,879.2	92.1	152	3.28	5.5	187.757	1.741	1.539
1963	4	2,739.4	1,710	335.2	759.2	1,910.5	92.5	153.3	3.5	5.6	188.434	1.7335	1.7632
1964	1	2,800.5	1,743.8	348.9	763.1	1,947.6	92.6	154.5	3.54	5.5	189.093	0.4322	3.1045
1964	2	2,833.8	1,775	347.5	772.9	1,999.4	92.9	155.6	3.48	5.2	189.755	1.2938	2.1862
1964	3	2,872	1,807.8	355.7	766.4	2,027.8	93.2	158.7	3.51	5	190.419	1.2896	2.217
1964	4	2,879.5	1,812.8	358.3	766.1	2,052.6	93.6	160.3	3.69	5	191.085	1.7131	1.9736
1965	1	2,950.1	1,852.5	394.9	765.5	2,071.8	93.7	161.5	3.9	4.9	191.675	0.4271	3.4729
1965	2	2,989.9	1,873.2	394.6	781.3	2,096.4	94.7	162.2	3.88	4.7	192.267	4.2463	−0.3663
1965	3	3,050.7	1,905.3	408.4	800.3	2,155.3	94.8	164.9	3.86	4.4	192.861	0.4222	3.4378
1965	4	3,123.6	1,959.3	410.1	817.2	2,200.4	95.4	167.8	4.16	4.1	193.457	2.5237	1.633
1966	1	3,201.1	1,988.6	444.1	832.5	2,219.3	96.3	170.5	4.63	3.9	193.965	3.7559	0.8774
1966	2	3,213.2	1,994	436.5	857.8	2,224.6	97.1	171.6	4.6	3.8	194.475	3.3092	1.2874
1966	3	3,233.6	2,016.6	432.7	870.1	2,254	98.1	172	5.05	3.8	194.986	4.0984	0.9516
1966	4	3,261.8	2,025.1	435.8	888	2,280.5	98.6	172	5.25	3.7	195.499	2.0336	3.2131
1967	1	3,291.8	2,037.3	424.9	925.60	2,312.6	98.9	174.8	4.54	3.8	195.966	1.2152	3.3215
1967	2	3,289.7	2,064.6	405	921.3	2,329.9	99.7	177	3.66	3.8	196.435	3.2226	0.4374
1967	3	3,313.5	2,075.2	415.2	926.8	2,351.4	100.7	180.7	4.34	3.8	196.904	3.992	0.3513
1967	4	3,338.3	2,087.9	423.6	934.8	2,367.9	101.6	183.3	4.79	3.9	197.375	3.5591	1.2276
1968	1	3,406.2	2,136.2	433.8	951.4	2,409.5	102.8	185.5	5.07	3.7	197.857	4.6967	0.3699
1968	2	3,464.8	2,169.6	451.8	956	2,451.2	104	189.4	5.51	3.6	198.341	4.6422	0.8711
1968	3	3,489.2	2,210.7	437.3	958.3	2,457.9	105.1	192.7	5.22	3.5	198.826	4.2086	1.0148
1968	4	3,504.1	2,220.4	442.2	960.5	2,474.3	106.4	197.4	5.58	3.4	199.312	4.9173	0.666
1969	1	3,558.3	2,244.8	470.8	956.9	2,477.5	108	200	6.14	3.4	199.807	5.9703	0.1697
1969	2	3,567.6	2,258.8	467.1	956	2,501.5	109.7	201.3	6.24	3.4	200.303	6.2473	−0.0039
1969	3	3,588.3	2,269	477.2	954.1	2,550.2	111.2	202.1	7.05	3.6	200.8	5.4324	1.6176
1969	4	3,571.4	2,286.5	452.6	943.1	2,568.1	112.9	203.9	7.32	3.6	201.298	6.0688	1.2512
1970	1	3,566.5	2,300.8	438	936.2	2,581.9	114.5	205.7	7.26	4.2	201.92	5.6289	1.6344
1970	2	3,573.9	2,312	439.4	927.3	2,626	116.3	207.6	6.75	4.8	202.545	6.2393	0.5107
1970	3	3,605.2	2,332.2	446.5	930.9	2,661.1	117.5	211.9	6.37	5.2	203.171	4.1061	2.2672

(continued)

(continued)

Year	qtr	GDP	realcons	realinvs	realgovt	realdpi	cpi_u	M1	tbilrate	unemp	pop	infl	realint
1970	4	3,566.5	2,324.9	421	929.9	2,650.9	119.1	214.3	5.36	5.8	203.799	5.4101	−0.0501
1971	1	3,666.1	2,369.8	475.9	918.6	2,703.5	119.8	218.7	3.87	5.9	204.549	2.3441	1.5226
1971	2	3,686.2	2,391.4	490.2	915.2	2,742.6	121.5	223.6	4.21	5.9	205.303	5.6362	−1.4296
1971	3	3,714.5	2,409.8	496.5	911.9	2,752.9	122.2	226.6	5.05	6	206.059	2.2979	2.7554
1971	4	3,723.8	2,449.8	480.6	909.4	2,782.1	123.1	228.2	4.23	5.9	206.818	2.9352	1.2981
1972	1	3,796.9	2,482.2	513.6	920.8	2,797.6	124	234.2	3.44	5.8	207.429	2.9138	0.5229
1972	2	3,883.8	2,527.5	544.9	921.9	2,822.9	125	236.8	3.75	5.7	208.043	3.2129	0.5338
1972	3	3,922.3	2,565.9	554.1	907.6	2,883.6	126.2	243.3	4.24	5.6	208.658	3.8217	0.4183
1972	4	3,990.5	2,626.3	559.4	909.1	2,993	127.3	249.1	4.85	5.4	209.275	3.4714	1.3819
1973	1	4,092.3	2,674.2	595.2	914.5	3,031.9	129.8	251.5	5.64	4.9	209.792	7.7793	−2.1393
1973	2	4,133.3	2,671.4	618.2	911.5	3,059.6	132.4	256.9	6.61	4.9	210.309	7.9331	−1.3231
1973	3	4,117	2,682.5	597.5	898.5	3,079.3	135.5	258	8.39	4.8	210.829	9.2576	−0.8676
1973	4	4,151.1	2,675.6	615.3	908.4	3,118.3	138.5	262.7	7.47	4.8	211.349	8.7595	−1.2928
1974	1	4,119.3	2,652.4	579.2	920	3,072.1	143.1	266.5	7.6	5.1	211.844	13.0693	−5.466
1974	2	4,130.4	2,662	577.3	927.8	3,045.5	146.9	268.6	8.27	5.2	212.339	10.4834	−2.2134
1974	3	4,084.5	2,672.2	543.4	924.2	3,053.3	151.7	271.3	8.29	5.6	212.836	12.8611	−4.5711
1974	4	4,062	2,628.4	547	927.4	3,036.7	155.4	274	7.34	6.6	213.334	9.639	−2.3024
1975	1	4,010	2,648.8	450.8	940.8	3,015	157.8	276.2	5.88	8.3	213.862	6.1304	−0.2537
1975	2	4,045.2	2,695.4	436.4	938.3	3,156.6	160.6	282.7	5.4	8.9	214.392	7.0354	−1.632
1975	3	4,115.4	2,734.7	474.9	941.8	3,114.9	163.6	286	6.34	8.5	214.924	7.403	−1.0664
1975	4	4,167.2	2,764.6	486.8	949.1	3,147.6	166.3	286.8	5.68	8.3	215.457	6.5476	−0.8643
1976	1	4,266.1	2,824.7	535.1	952.5	3,201.9	167.5	292.4	4.95	7.7	215.979	2.876	2.0773
1976	2	4,301.5	2,850.9	559.8	943.3	3,229	170.1	296.4	5.17	7.6	216.503	6.1613	−0.9879
1976	3	4,321.9	2,880.3	561.1	938.9	3,259.7	172.6	300	5.17	7.7	217.028	5.8361	−0.6661
1976	4	4,357.4	2,919.6	565.9	938.6	3,283.5	174.3	305.9	4.7	7.8	217.554	3.9205	0.7795
1977	1	4,410.5	2,954.7	595.5	945.3	3,305.4	178.2	313.6	4.62	7.5	218.104	8.8514	−4.2281
1977	2	4,489.8	2,970.5	635	955.1	3,326.8	181.8	319	4.83	7.1	218.655	8.0003	−3.1736
1977	3	4,570.6	2,999.1	670.7	956	3,376.5	184	324.9	5.47	6.9	219.207	4.8114	0.6586

(continued)

Year	qtr	GDP	realcons	realinvs	realgovt	realdpi	cpi_u	M1	tbilrate	unemp	pop	infl	realint
1977	4	4,576.1	3,044	656.4	954.5	3,433.8	186.1	330.5	6.14	6.7	219.761	4.5394	1.5973
1978	1	4,588.9	3,060.8	667.2	956.7	3,466.3	189.8	336.6	6.41	6.3	220.343	7.8747	−1.4647
1978	2	4,765.7	3,127	709.7	982.1	3,513	195.3	347.1	6.48	6	220.926	11.4264	−4.9431
1978	3	4,811.7	3,143.1	728.8	990.3	3,548.1	199.3	352.7	7.32	6	221.512	8.1098	−0.7898
1978	4	4,876	3,167.8	746.3	999.6	3,582.6	202.9	356.9	8.68	5.9	222.098	7.1608	1.5192
1979	1	4,888.3	3,188.6	746	990.6	3,620.7	209.1	362.1	9.36	5.9	222.713	12.0397	−2.6797
1979	2	4,891.4	3,184.3	745.7	1,000.5	3,607.1	216.6	373.6	9.37	5.7	223.330	14.0959	−4.7226
1979	3	4,926.2	3,213.9	732.1	1,002.4	3,628.8	223.4	379.7	9.63	5.9	223.948	12.3646	−2.7313
1979	4	4,942.6	3,225.7	717.8	1010.8	3,657.8	229.9	381.4	11.8	6	224.569	11.4722	0.3311
1980	1	4,958.9	3,222.4	711.7	1025.6	3,678.5	239.8	388.1	13.46	6.3	225.23	16.8643	−3.401
1980	2	4,857.8	3,149.2	647.4	1,028.7	3,612.2	247.6	389.4	10.05	7.3	225.893	12.8037	−2.7537
1980	3	4,850.3	3,181.2	599.8	1,015.4	3,637.6	251.7	405.4	9.24	7.7	226.558	6.5693	2.6673
1980	4	4,936.6	3,219.4	662.2	1,013.9	3,703.8	258.4	408.1	13.71	7.4	227.225	10.5083	3.2017
1981	1	5,032.5	3,233.1	726.3	1,027.5	3,713.5	265.1	418.7	14.37	7.4	227.783	10.2393	4.134
1981	2	4,997.3	3,235.5	693.4	1,030.1	3,969.6	271.3	425.5	14.83	7.4	228.342	9.2472	5.5795
1981	3	5,056.8	3,250.5	733.9	1,027.8	3,777	279.3	427.5	15.09	7.4	228.903	11.6245	3.4622
1981	4	4,997.1	3,225	708.8	1,034.8	3,777.2	281.5	436.2	12.02	8.2	229.466	3.1384	8.8849
1982	1	4,914.3	3,244.3	634.8	1,033.6	3,769.4	283.1	442.2	12.89	8.8	230.013	2.2671	10.6262
1982	2	4,935.5	3,253.4	631.6	1,039.5	3,791.4	290.6	447.9	12.36	9.4	230.562	10.459	1.901
1982	3	4,912.1	3,274.6,	623.5	1,046.8	3,799.4	293.3	457.5	9.71	9.9	231.113	3.6993	6.0107
1982	4	4,915.6	3,329.6	571.1	1,064	3,806.4	292.4	474.3	7.94	10.7	231.664	−1.2293	9.166
1983	1	4,972.4	3,360.1	590.7	1,069.8	3,831.2	293.4	490.2	8.08	10.4	232.195	1.3657	6.7143
1983	2	5,089.8	3,430.1	650.7	1,078.2	3,857.8	298.1	504.4	8.42	10.1	232.726	6.3569	2.0631
1983	3	5,180.4	3,484.7	691.4	1,097	3,928.6	301.8	513.4	9.19	9.4	233.258	4.9342	4.2525
1983	4	5,286.8	3,542.2	762.2	1,078.8	4,010.2	303.5	520.8	8.79	8.5	233.792	2.2468	6.5465
1984	1	5,402.3	3,579.7	845	1,091	4,103	307.3	530.8	9.01	7.9	234.299	4.9771	4.0362
1984	2	5,493.8	3,628.3	873.2	1,115.2	4,182.4	310.7	540.5	9.84	7.4	234.806	4.4013	5.442
1984	3	5,541.3	3,653.5	890.7	1,123.1	4,258.8	314.5	543.9	10.34	7.4	235.315	4.8625	5.4808

(continued)

(continued)

Year	qtr	GDP	realcons	realinvs	realgovt	realdpi	cpi_u	M1	tbilrate	unemp	pop	infl	realint
1984	4	5,583.1	3,700.9	876.9	1,144.2	4,286.1	315.5	551.2	8.97	7.3	235.825	1.2698	7.7035
1985	1	5,629.7	3,756.8	848.9	1,157.6	4,287.6	318.8	565.7	8.17	7.2	236.348	4.1621	4.0046
1985	2	5,673.8	3,791.5	862.8	1,180.5	4,368.7	322.3	582.9	7.52	7.3	236.872	4.3675	3.1558
1985	3	5,758.6	3,860.9	854.1	1,209.2	4,346.6	324.5	604.4	7.1	7.2	237.397	2.7211	4.3822
1985	4	5,806	3,874.2	887.8	1,214.7	4,388.3	327.4	619.1	7.15	7	237.924	3.5589	3.5878
1986	1	5,858.9	3,907.9	886.2	1,224	4,444.5	326	632.6	6.89	7	238.474	−1.7141	8.6008
1986	2	5,883.3	3,950.4	868.3	1,248	4,489.3	327.9	661.2	6.13	7.2	239.026	2.3245	3.8055
1986	3	5,937.9	4,019.7	838	1,277.4	4,507.9	330.2	688.4	5.53	7	239.579	2.7959	2.7374
1986	4	5,969.5	4,046.8	838.2	1,271.5	4,504.5	331.1	724	5.34	6.8	240.133	1.0888	4.2512
1987	1	6,013.3	4,049.7	863.4	1,278.4	4,556.9	335.9	732.8	5.53	6.6	240.670	5.7572	−0.2239
1987	2	6,077.2	4,101.5	863.9	1,289.1	4,512.7	340.1	743.5	5.73	6.3	241.208	4.9705	0.7629
1987	3	6,128.1	4,147	860.5	1,292.4	4,600.7	344.4	748.5	6.03	6	241.748	5.0256	1.0077
1987	4	6,234.4	4,155.3	929.3	1,310	4,659.6	345.7	749.4	6	5.8	242.289	1.507	4.4963
1988	1	6,275.9	4,228	884.6	1,300.1	4,724.1	349	761.1	5.76	5.7	242.84	3.8002	1.9598
1988	2	6,349.8	4,256.8	902.5	1,302.4	4,758.9	353.5	778.8	6.23	5.5	243.391	5.1246	1.1054
1988	3	6,382.3	4,291.6	907.5	1,300.3	4,801.9	358.9	784.6	6.99	5.5	243.945	6.0641	0.9292
1988	4	6,465.2	4,341.4	916.7	1,327.2	4,851.4	360.9	786.1	7.7	5.3	244.499	2.2228	5.4805
1989	1	6,543.8	4,357.1	952.7	1,319.3	4,903.5	366.2	782.7	8.53	5.2	245.077	5.8315	2.7018
1989	2	6,579.4	4,374.8	941.1	1,340.6	4,891	371.7	773.9	8.44	5.2	245.656	5.963	2.477
1989	3	6,610.6	4,413.4	929.3	1,353.5	4,902.7	374.6	782	7.85	5.2	246.237	3.1087	4.7413
1989	4	6,633.5	4,429.4	922.9	1,360.4	4,928.8	377.6	792.1	7.64	5.4	246.819	3.1907	4.4493
1990	1	6,716.3	4,466	934	1,381.2	5,001.6	385.5	800.8	7.76	5.3	247.478	8.2823	−0.5256
1990	2	6,731.7	4,478.8	933	1,384.7	5,026.6	389.1	809.7	7.77	5.3	248.138	3.7181	4.0486
1990	3	6,719.4	4,495.6	912.6	1,384.8	5,032.7	397.5	821.1	7.49	5.7	248.8	8.5434	−1.0501
1990	4	6,664.2	4,457.7	849.6	1,398.6	4,995.8	400.9	823.9	7.02	6.1	249.464	3.4068	3.6165
1991	1	6,631.4	4,437.5	815.1	1,404.7	4,999.5	404.3	838	6.05	6.6	250.134	3.3781	2.6753

(continued)

Year	qtr	GDP	realcons	realinvs	realgovt	realdpi	cpi_u	M1	tbilrate	unemp	pop	infl	realint
1991	2	6,668.5	4,469.9	808.8	1,408.9	5,033.3	407.3	857.4	5.59	6.8	250.805	2.9571	2.6362
1991	3	6,684.9	4,484.3	829.8	1,403	5,045.4	411.1	871.2	5.41	6.9	251.478	3.7146	1.6921
1991	4	6,720.9	4,474.8	864.2	1,397	5,053.8	413	895.9	4.58	7.1	252.153	1.8444	2.7389
1992	1	6,783.3	4,544.8	843.8	1,407.6	5,138.8	417.2	935.8	3.91	7.4	252.869	4.0473	−0.1373
1992	2	6,846.8	4,566.7	901.8	1,405.7	5,172.5	419.9	954.5	3.72	7.6	253.587	2.5803	1.143
1992	3	6,899.7	4,600.5	912.1	1,413.1	5,174.2	423.2	988.7	3.13	7.6	254.307	3.1313	−0.0013
1992	4	6,990.6	4,665.9	941.6	1,413.7	5,271.5	425.2	1,024	3.08	7.4	255.03	1.8859	1.1908
1993	1	6,988.7	4,674.9	964.8	1,396.4	5,181.2	430.1	1,038.1	2.99	7.1	255.715	4.5832	−1.5899
1993	2	7,031.2	4,721.5	967	1,398	5,258.6	432.4	1,075.3	2.98	7.1	256.402	2.1333	0.85
1993	3	7,062	4,776.9	964.1	1,398.4	5,266.8	434.7	1,105.2	3.02	6.8	257.092	2.122	0.898
1993	4	7,168.7	4,822.3	1,015.6	1,402.2	5,338.5	436.8	1,129.2	3.08	6.6	257.783	1.9277	1.1523
1994	1	7,229.4	4,866.6	1,057.3	1,388	5,293.2	441.1	1,140	3.25	6.6	258.416	3.9185	−0.6685
1994	2	7,330.2	4,907.9	1,118.5	1,390.4	5,381.2	443.3	1,145.6	4.04	6.2	259.052	1.9901	2.0466
1994	3	7,370.2	4,944.5	1,101.8	1,417.5	5,420.9	447.5	1,152.1	4.51	6	259.689	3.7719	0.7381
1994	4	7,461.1	4,993.6	1,150.5	1,404.5	5,493.4	448.4	1,149.8	5.28	5.6	260.327	0.8037	4.4797
1995	1	7,488.7	5,011.6	1,162.4	1,407.3	5,515.4	453.5	1,146.5	5.78	5.5	260.944	4.5238	1.2562
1995	2	7,503.3	5,059.6	1,128.5	1,414	5,509	456.7	1,144.1	5.62	5.7	261.562	2.8126	2.8107
1995	3	7,561.4	5,099.2	1,119.1	1,410.8	5,546.6	459	1,141.9	5.38	5.7	262.182	2.0094	3.3706
1995	4	7,621.9	5,132.1	1,152.4	1,393.5	5,585.3	459.9	1,126.2	5.27	5.6	262.803	0.7835	4.4865
1996	1	7,676.4	5,174.3	1,172.3	1,404.8	5,622	466.5	1,122	4.95	5.5	263.408	5.6996	−0.7496
1996	2	7,802.9	5,229.5	1,233.4	1,430.4	5,649.4	469.9	1,115	5.04	5.5	264.013	2.5641	2.4759
1996	3	7,841.9	5,254.3	1,281.4	1,422	5,709.7	472.7	1,095.8	5.14	5.3	264.62	2.7171	2.4196

(continued)

(continued)

Year	qtr	GDP	realcons	realinvs	realgovt	realdpi	cpi_u	M1	tbilrate	unemp	pop	infl	realint
1996	4	7,931.3	5,291.9	1,283.7	1,430.6	5,729.9	475	1,080.5	4.97	5.3	265.229	1.9415	3.0285
1997	1	8,016.4	5,350.7	1,325.4	1,434.6	5,771.8	479.3	1,072	5.06	5.2	265.865	3.6048	1.4586
1997	2	8,131.9	5,375.7	1,400.6	1,457	5,821.2	480.2	1,066.2	5.07	5	266.503	0.7504	4.3229
1997	3	8,216.6	5,462.1	1,408.6	1,464.8	5,877.3	483	1,065.3	5.06	4.9	267.143	2.3256	2.7311
1997	4	8,272.9	5,507.1	1,438.5	1,465.3	5,947.5	483.2	1,073.4	5.09	4.7	267.784	0.1656	4.9211
1998	1	8,396.3	5,576.3	1,543.3	1,456.1	6,064.5	485.8	1,080.3	5.08	4.6	268.398	2.1465	2.9301
1998	2	8,442.9	5,660.2	1,516.8	1,482.6	6,153.6	488.2	1,077.6	5.01	4.4	269.013	1.9713	3.0354
1998	3	8,528.5	5,713.7	1,559.7	1,489.9	6,209.9	490.1	1,076.2	4.88	4.5	269.63	1.5537	3.3263
1998	4	8,667.9	5,784.7	1,612.1	1,504.8	6,246.6	491	1,097	4.31	4.4	270.248	0.7339	3.5795
1999	1	8,733.5	5,854	1,641.8	1,512.3	6,268.2	494.4	1,102.2	4.42	4.3	270.857	2.7603	1.663
1999	2	8,771.2	5,936.1	1,617.4	1,516.8	6,300	497.9	1,099.8	4.46	4.3	271.467	2.8217	1.6383
1999	3	8,871.5	6,000	1,655.8	1,533.2	6,332.4	502.9	1,093.4	4.7	4.2	272.078	3.9968	0.6998
1999	4	9,049.9	6,083.6	1,725.4	1,564.8	6,379.2	504.1	1,124.8	5.06	4.1	272.691	0.9533	4.1067
2000	1	9,102.5	6,171.7	1,722.9	1,560.4	6,431.6	512.8	1,113.7	5.54	4	274.848	6.8445	−1.3012
2000	2	9,229.4	6,226.3	1,801.6	1,577.2	6,523.7	516.5	1,105.3	5.78	4	277.022	2.8758	2.9009
2000	3	9,260.1	6,292.1	1,788.8	1,570	6,566.5	520.3	1,096	6.03	4.1	279.213	2.9321	3.0979
2000	4	9,303.9	6,341.1	1,778.3	1,582.8	6,634.9	521.1	1,088.1	6.03	4	281.422	0.6146	5.4154

References

Adams, M. O. (1981). The Kentucky Economic Information System: A resource for government, academic, and private sector researchers and libraries. In M. E. Williams, & T. H. Hogan (Eds.), *National Online Meeting Proceedings - 1981*. Medford, NJ: Learned Information

Adams, M. O. (1986). The Kentucky Economic Information System. *In Exemplary systems in government awards '85–'86: The state of the art* (pp. 144–158). Urban and Regional Information Systems Association

Adams, M. O., & Ross, F. E. (1983). The energy data base within the Kentucky Economic Information System. *Review of Public Data Use, 11*(1), 75–78

Adelman, I. (2007). The research for the paper on the dynamics of the Klein-Goldberger model. *Journal of Economic and Social Measurement, 32*(1), 29–33

Adelman, F., & Adelman, I. (1959). The dynamic properties of the Klein-Goldberger model. *Econometrica, 27*, 596–625

Aitchinson, J. (1962). Large sample restricted parametric tests. *Journal of the Royal Statistical Society, Series B, 24*

Aitchison, J., & Brown, J. A. C. (1957). *The log normal distribution*. Cambridge: Cambridge University Press

Aitchinson, J., & Silvey, S. D. (1958). Maximum likelihood estimation of parameters subject to restraints. *Annals of Mathematical Statistics, 29*

Akaike, H. (1969). Fitting autoregressive models for prediction. *Annals of the Institute of Statistical Mathematics, 21*, 243–247

Akaike, H. (1973). Information theory and the extension of the maximum likelihood principle. In B. N. Petrov, & F. Csaki (Eds.), *2nd International Symposium on Information Theory* (pp. 267–281). Akailseooniai-Kiudo: Budapest

Aldrich, J. (1995). Correlations genuine and spurious in Pearson and Yule. *Statistical Science, 10*(4), 364–376

Alexander, L. A., & Jabine, T. B. (1980). Access to social security microdata files for research and statistical purposes. *Review of Public Data Use, 8*, 203–224

Allen, A. T. (Ed.). (1982). SAS/ETS *user's guide: Econometrics and Time Series Library* (1982 ed.). Cary, NC: SAS Institute, Inc

Altman, M., Gill, J., & McDonald, M. P. (2004). *Numerical issues in statistical computing for the social scientist*. Hoboken, NJ: Wiley

Amemiya, T. (1973). Regression analysis when variance of dependent variable is proportional to the square of its expectation. *Journal of the American Statistical Association, 68*, 928–934

Amemiya, T. (1980). Selection of regressors. *International Economic Review, 21*, 331–354

Amemiya, T. (1985). *Advanced econometrics* (Vol. 6, p. 521). Cambridge, MA.: Harvard University Press

Amman, H. M., & Kendrick, D. (1999). Programming languages in economics. *Computational Economics, 14*(1–2), 151–181

Anderson, R. G. (2006). Replicability, real-time data, and the science of economic research: FRED, ALFRED, and VDC. *Federal Reserve Bank of St. Louis Review, 88*(1), 81–93

Anderson, R. G., Greene, W. H., McCullough, B. M., & Vinod, H. D. (2008). The role of data and code archives in the future of economic research. *Journal of Economic Methodology*, *15*(1), 99–119

Andrews, D. W. K. (1993). Tests for parameter instability and structural change with unknown change point. *Econometrica, 61*, 821–856

Andrews, D. W. K., & Ploberger, W. (1994). Optimal tests when a nuisance parameter is present only under the alternative. *Econometrica, 62*, 1383–1414

Atkinson, A. C. (1970). A method for discriminating between models. *Journal of the Royal Statistical Society, Series B*, 32

Backus, J. W., Beeber, R. J., Best, S., Goldbereg, R., Haibt, L. M., Herrick, H. L., et al. (1957). *The fortran automatic coding system*, in *Proceeding of the Western Joint Computer Conference.* Los Angeles, CA

Banerjee, A., Dolado, J. J., Galbraith, J. W., & Hendry, D. F. (1993). *Co-integration, error correction, and the econometric analysis of non-stationary data*. New York: Oxford University Press

Barger, H., & Klein, L. R. (1954). A quarterly model for the United States economy. *Journal of the American Statistical Association, 49*, 413–437

Barker, T. S., Dada, W., & Peterson, W. (2004). Software developments in the Cambridge growth project 1960–1976: The origins of software for space-time economics. *Journal of Economic and Social Measurement, 29*, 173–182

Bartlett, M. S. (1955). *An introduction to stochastic processes with special reference to methods and applications*. Cambridge: Cambridge University Press

Barton, H. C. (1941). Adjustment for seasonal variation. *Federal Reserve Bulletin, 27*, 518–528

Begg, I., & Henry, S. G. B. (1998). *Applied economics and public policy*. Cambridge: Cambridge University Press

Belsley, D. A. (1974). Estimation of systems of simultaneous equations, and computational specification of GREMLIN. *Annals of Economic and Social Measurement, 3*, 551–614

Belsley, D. A. (1986). Centering, the constant, first-differencing, and assessing conditioning. In *model reliability* (pp. 117–152). Cambridge, MA: MIT

Belsley, D. A. (1991). *Conditioning diagnostics*. New York: Wiley

Belsley, D. A. (1999). Mathematica as an environment for doing economics and econometrics. *Computational Economics, 14*(1–2), 69–87

Belsley, D. A., & Kuh, E. (Eds.). (1986). *Model reliability*. Cambridge, MA: MIT

Belsley, D. A., Kuh, E., & Welsch, R. E. (1980). *Regression diagnostics*. New York: Wiley

Bera, A. K., & Jarque, C. M. (1981). An efficient large-sample test for normality of observations and regression residuals. *Australian National University Working Papers in Econometrics, 40*

Bera, A. K., & Jarque, C. M. (1982). Model specification tests. *Journal of Econometrics, 20*, 59–82

Bergstrom, A. R. (1967a). *The construction and use of economic models*. London: English Universities Press

Bergstrom, A. R. (1967b). *Selected economic models and their analysis* (Vol. x, pp. 131). New York: Elsevier

Berk, K. N. (1987). Effective microcomputer statistical software. *American Statistician, 41*(3), 222–228

Bernanke, B. S. (2004). Editorial statement. *American Economic Review, 94*, 404

Berndt, E. R. (1991). *The practice of econometrics: Classic and contemporary*. Reading, MA: Addison-Wesley

Berndt, E. R., Hall, B. F., Hall, R. E., & Hausman, J. (1974). Estimation and inference in nonlinear structural models. *Annals of Economic and Social Measurement, 3*, 653–665

Berry, D. A., Chaloner, K., Geweke, J., & Zellner, A. (1996). *Bayesian analysis in statistics and econometrics: Essays in honor of Arnold Zellner* (Vol. xxi, p. 577). Wiley series in probability and mathematical statistics. Applied probability and statistics. New York: Wiley

Bhargava, A. (1987). Wald tests and systems of stochastic equations. *International Economic Review, 28*(3), 789–808

Blinder, A. S., & Deaton, A. (1985). The time series consumption function revisited. *Brookings Papers on Economic Activity, 2*, 465–521

Bloch, R. M. (1948). Mark I calculator. In *Proceedings of a Symposium on Large-Scale Digital Calculating Machinery*. Cambridge, MA: Harvard University Press

Bodkin, R. G. (1999). Computation in macroeconometric model-building: Some historical aspects. In D. J. Slottje (Ed.), *Advances in econometrics, income distribution, and scientific methodology: Essays in honor of Camilo Dagum* (pp. 41–60). New York: Physica

Bodkin, R. G., Klein, L. R., & Marwah, K. (1991). *A history of macroeconometric model-building* (p. 573). Brookfield, Vermont: Edward Elgar

Bollerslev, T. (1988). On the correlation structure of the generalized autoregressive conditional heteroscedastic process. *Journal of Time Series Analysis, 9*, 121–131

Boschan, C. (1972). The NBER time series data bank. *Annals of Economic and Social Measurement, 1*, 193–209

Bowden, R. J. (1972). More stochastic properties of the Klein-Goldberger model. *Econometrica, 40*(1), 87–98

Box, G., & Jenkins, G. (1984). *Time series analysis: Forecasting and control* (2nd ed.). San Francisco: Holden Day

Box, G., & Pierce, D. (1970). Distribution of residual autocorrelations in autoregressive moving average time series models. *Journal of the American Statistical Association, 65*, 1509–1526

Bracy, D., Brown, B. W., Golderer, J. M., Proctor, M. O., Rowe, D. M., Schink, G. R., et al. (Eds.). (1975). *Computer programs available for distribution*. Philadelphia, PA: Wharton Econometric Forecasting Associates

Breusch, T. S. (1978). Testing for autocorrelation in dynamic linear models. *Australian Economic Papers, 17*, 334–355

Breusch, T. S., & Gray, E. (2006). Replicating a study of mothers' foregone earnings in Australia. *Journal of Economic and Social Measurement, 31*, 107–125

Breusch, T. S., & Pagan, A. (1979). A simple test for hetroscedasticity and random coefficient variation. *Econometrica, 47*, 1287–1294

Brillet, J.-L. (1989). Econometric modeling on microcomputers: A review of major software packages. *Journal of Applied Econometrics, 4*(1), 73–92

Brooks, C., Burke, S., & Persand, G. (2001). Benchmarks and the accuracy of GARCH model estimation. *International Journal of Forecasting, 17*, 45–56

Brown, B. W. (1975). *WEFA batch solution program, WEFA program usage guide #2*. In D. Bracy, et al. (Eds.), *Computer programs available for distribution*. Philadelphia, PA: Wharton Econometric Forecasting Associates

Brown, B., Durbin, J., & Evans, J. (1975). Techniques for testing the constancy of regression relationships over time. *Journal of the Royal Statistical Society, Series B, 37*, 149–172

Brown, J. A. C., Houthakker, H. S., & Prais, S. J. (1953). Electronic computation in economic statistics. *Journal of the American Statistical Association, 48*(263), 414–428

Bruno, G., & De Bonis, R. (2004). A comparative study of alternative econometric packages with an application to Italian deposit interest rates. *Journal of Economic and Social Measurement, 29*, 271–296

Burns, A. F., & Mitchell, W. C. (1946). *Measuring business cycles*. New York: National Bureau of Economic Research

Byte. (1984). Guide to the IBM Personal Computers. *Byte: The Small Systems Journal, 9*(9)

Byte. (1986). Inside the IBM PCs. *Byte, the Small Systems Journal, 11*(11)

Cameron, A. C., & Trivedi, P. K. (1990). *Conditional moment tests and orthogonal polynomials*. Department of Economics, Bloomington: Indiana University

Cameron, A. C., & Trivedi, P. K. (1993). Tests of independence in parametric models with applications and illustrations. *Journal of Business and Economic Statistics, 11*(1), 29–43

Campbell-Kelly, M., & Williams, M. R. (Eds.). (1985). *The Moore School lectures*. Cambridge, MA: MIT

Carson, C. (1975). The history of the United States national income and product accounts: The development of an analytic tool. *Review of Income and Wealth, 21*, 153–181

Carson, C. S., Grimm, B. T., & Moylan, C. E. (1994). A satellite account for research and development. *Survey of Current Business, 74*(11), 37–70
Carson, C. S., & Landefeld, J. S. (1994a). Integrated economic and environmental satellite accounts. *Survey of Current Business, 74*(4), 33–49
Carson, C. S., & Landefeld, J. S. (1994b). Accounting for mineral resources: Issues and BEA's initial estimates. *Survey of Current Business, 74*(4), 50–72
Cartwright, D. (1983). Establishment reporting in major administrative record systems. *Review of Public Data Use, 11*, 1–10
Cartwright, D. W., & Aarmknecht, P. A. (1980). Statistical uses of administrative records. *Review of Public Data Use, 8*, 1–10
Cecchetti, S., & Rich, R. (2001). Structural estimates of the U.S. sacrifice ratio. *Journal of Business and Economic Statistics, 19*, 416–427
Center for Computational Research in Economics and Management Science. (1979). *TROLL primer* (3rd ed.). Cambridge, MA: MIT Center for Computational Research in Economics and Management Science
Chapman, B., & Gray, M. (2006). Response to Breusch and Gray. *Journal of Economic and Social Measurement, 31*, 127–138
Chow, G. C. (1960). Tests of equality between sets of coefficients in two linear regressions. *Econometrica, 28*, 591–605
Christ, C. F. (1994). The Cowles Commission's contributions to econometrics at Chicago 1939–1955. *Journal of Economic Literature, 32*(1), 30–59
Cohen, S. B. (1983). Present limitations in the availability of statistical software for the analysis of complex survey data. *Review of Public Data Use, 11*(4), 338–344
Consigliere, I. (1981). The Chow test with serially correlated errors. *Rivista Internazionale dei Scienze Sociali, 89*, 125–137
Cooper, F. (1987). *Modeling the U.S. trade sector*. Paper presented to the 1987 MODLER users conference, October 21, 1987, Philadelphia, PA
Cordeiro, J. A. (2007). Logical operators of some statistical computing packages and missing values. *Computational Statistics and Data Analysis, 51*, 2783–2787
Costa, D., Demeulemeester, J. -L., & Diebolt, C. (2007). What is 'Cliometrica'? *Cliometrica, 1*(1), 1–6
Cox, D. R. (1961). Tests of separate families of hypotheses. In *Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability*. Berkeley: University of California Press
Cox, D. R. (1962). Further results on tests of separate families of hypotheses. *Journal of the Royal Statistical Society, Series B, 24*
Cramer, M. (2006). The working conditions of econometric scholars, then and now. *Statistica Neerlandica, 60*(2), 194–2005
Craven, P., & Wahba, G. (1979). Smoothing noisy data with spline functions: Estimating the correct degree of smoothing by the method of generalized cross-validation. *Numerishce Mathematik, 31*, 377–403
Cripps, T. F., & Fetherston, M. (1979). Cambridge economic policy group methodology. In Ormerod, P. (Ed.), *Economic modelling* (pp. 40–52). London: Heinemann
Critchley, F., Marriott, P., & Salmon, M. (1996). On the differential geometry of the Wald test with nonlinear restrictions. *Econometrica, 64*(5), 1213–1222
Croushore, D., & Stark, T. (2000). *A real-time data set for macroeconomists: Does data vintage matter for forecasting?* Working paper. Federal Reserve Bank of Philadelphia: Research Department
Cuff, R. N. (1980). On casual users. *International Journal of Man-Machine Studies, 12*, 163–187
Cummins, C. (2006). *Email re the Breusch-Godfrey LM*. C. G. Renfro, Recipient
D'Agostino, R. B., Balanger, A., & D'Agustino, R. B. Jr. (1990). A suggestion for using powerful and informative tests of normality. *American Statistician, 44*(4), 316–321
Darnell, A. C. (1994). *The history of econometrics*. Aldershot, Hants: E. Elgar
Darnell, A. C., & Evans, L. (1990). *The limits of econometrics*. Aldershot, Hants: E. Elgar

Darroch, N. N., & Silvey, S. D. (1963). On testing more than one hypothesis. *Annals of Mathematical Statistics, 34*
David, M. H. (1980). Access to data: The frustration and utopia of the researcher. *Review of Public Data Use, 8*, 327–337
David, M. H. (1989). Managing panel data for scientific analysis: The role of relational database management systems. In D. Kasprzyk, G. Duncan, G. Kalton, & M. P. Singh (Eds.), *Panel surveys* (pp. 226–241). New York: Wiley
David, M. H., & Robbin, A. (1981). The great rift: Gaps between administrative records and knowledge created through secondary analysis. *Review of Public Data Use, 9*(3), 153–166
Davidson, J. (2000). *Econometric theory*. Oxford: Blackwell Publishers
Davidson, J. (2006). *Email re core statistics table*. C. G. Renfro, Recipient
Davidson, J. (2008). *Time Series Modelling Version 4.28*. Exeter: University of Exeter
Davidson, J. (2009). When is a time series I(0)? in *The Methodology and Practice of Econometrics*, Jennifer Castle and Neil Shepherd (Eds.). Oxford University Press
Davidson, J. E. H., & Hendry, D. F. (1981). Interpreting econometric evidence: Consumers expenditure in the UK. *European Economic Review, 16*, 177–192
Davidson, J. E. H., Hendry, D. F., Srba, F., & Yeo, S. (1978). Econometric modelling of the aggregate time-series relationship between consumer's expenditure and income in the United Kingdom. *Economic Journal, 88*, 661–692
Davidson, R., & MacKinnon, J. G. (1981). Several tests for model specification in the presence of alternative hypotheses. *Econometrica, 49*, 781–793
Davidson, R., & MacKinnon, J. G. (1985). The interpretation of test statistics. *Canadian Journal of Economics, 18*(1), 38–57
Davidson, R., & MacKinnon, J. G. (1993). *Estimation and inference in econometrics*. New York: Oxford University Press
DeJong, D. N., Nankervis, J. C., Savin, N. E., & Whiteman, C. H. (1989). Integration versus trend stationarity in macroeconomic time series. *Working paper no. 89–99*. Department of Economics, University of Iowa, Iowa City, IA
de la Vina, L. Y. (1985). Econometric software for microcomputers. *Journal of Business & Economic Statistics, 3*(1), 89–91
de Marchi, N., & Gilbert, C. (1989). *History and methodology of econometrics*, Oxford: Clarendon
de Mello, J. (1988) Computable general equilibrium models for trade policy analysis in developing countries: A survey. *Journal of Policy Modeling, 10*(4), 469–503
Demeulemeester, J. -L., & Diebolt, C. (2007). How much could economics gain from history: The contribution of cliometrics. *Cliometrica, 1*(1), 7–17
Dennis, J. E., & Welsch, R. E. (1978). Techniques for nonlinear least squares and robust regression. *Communications in Statistics, 7*, 345–359
Dennis, J. E. J., Gay, D. M., & Welsch, R. E. (1981a). An adaptive nonlinear least-squares algorithm. *ACM Transactions on Mathematical Software, 7*, 348–368
Dennis, J. E. J., Gay, D. M., & Welsch, R. E. (1981b). Algorithm 573 NL2SOL–An adaptive nonlinear least-squares algorithm. *ACM Transactions on Mathematical Software, 7*, 369–383
Dent, W. T. (1980). Computation in econometric models. *Journal of Econometrics: Annals of Applied Econometrics, 12*
Desai, M. (2007). Memories of Monroe: Econometric computing in the early 1960s. *Journal of Economic and Social Measurement, 32*(1), 35–38
Desai, M., Hendry, D. F., & Mizon, G. E. (1997). John Denis Sargan: An obituary. *Economic Journal, 107*, 1121–1125
Dewald, W. G., Thursby, J. G., & Anderson, R. G. (1986). Replication in empirical economics: The Journal of Money, Credit and Banking Project. *American Economic Review, 76*(4), 587–603
Dhrymes, P. J., Howrey, E. P., Hymans, S. H., Kmenta, J., Leamer, E. E., Quandt, R. E., et al. (1972). Criteria for the evaluation of econometric models. *Annals of Economic and Social Measurement, 3*(1), 291–324
Dickey, D. A., Bell, W. R., & Fuller, W. A. (1986). Unit roots in time series models: Tests and implications. *The American Statistician, 40*, 12–26

Dickey, D. A. F, & Fuller, W. A. (1979). Distribution of the estimators for autoregressive time series with a unit root. *Journal of the American Statistical Association, 74*, 427–431

Diebold, J. (1962). The application of information technology. *Annals of the American Academy of Political and Social Science, 340*, 38–45

Diebold, F. X. (1998). The past, present, and future of macroeconomic forecasting. *Journal of Economic Perspectives, 12*(2), 175–192

Doornik, J. A., & Hansen, H. (2008). An omnibus test for univariate and multivariate normality, *Oxford Bulletin of Economics and Statistics, 70*, 927–939

Drud, A. (1983). A survey of model representations and simulation algorithms in some existing modeling systems. *Journal of Economic Dynamics and Control, 5*, 5–35

Duesenberry, J. S., Fromm, G., Klein, L. R., & Kuh, E. (1965). The *Brookings Quarterly Econometric Model of the United States*. Chicago: Rand McNally

Duesenberry, J. S., Fromm, G., Klein, L. R., & Kuh, E. (1969). *The Brookings model: Some further results*. Chicago: Rand McNally

Duesenberry, J. S., & Klein, L. R. (1965). Introduction: The research strategy and its application. In J. S. Duesenberry, G. Fromm, L. R. Klein, & E. Kuh (Eds.), *The Brookings Quarterly Econometric Model of the United States* (pp. 1–32). Chicago: Rand McNally

Durbin, J. (1954). Errors in variables. *Review of the International Statistical Institute, 22*, 23–32

Durbin, J. (1970). Testing for serial correlation in least squares regression when some of the regressors are lagged dependent variables. *Econometrica, 38*, 410–421

Durbin, J. (1988). Maximum likelihood estimation of the parameters of a system of simultaneous regression equations. *Econometric Theory, 4*, 159–170

Durbin, J., & Watson, G. S. (1950). Testing for serial correlation in least squares regression. *Biometrika, 37*, 409–428

Durbin, J., & Watson, G. S. (1951). Testing for serial correlation in least squares regression. *Biometrika, 38*, 159–178

Eastwood, A., & Godfrey, L. G. (1992). The properties and constructive use of misspecification tests for multiple regression models. In L. G. Godfrey (Ed.), *The implementation and constructive use of misspecification tests in econometrics* (pp. 109–175). Manchester: Manchester University Press

Edwards, J. B., & Orcutt, G. (1969). Should aggregation prior to estimation be the rule. *Review of Economics and Statistics, 51*(4), 409–420

Eisenpress, H. (1959). *Forecasting by generalized regression methods. Limited-information estimation procedure IBM 704 program IB LI*. New York: International Business Machines Corporation. Data Systems Division

Eisner, M. (1972). TROLL/1 – An interactive computer system for econometric research. *Annals of Economic and Social Measurement, 1*, 95–96

Eisner, M., & Pindyck, R. S. (1973). A generalized approach to estimation as implemented in the TROLL/1 system. *Annals of Economic and Social Measurement, 2*, 29–51

Eisner, R. (1989). Divergences of measurement and theory and some implications for economic policy. *American Economic Review, 79*, 1–13

Eisner, R. (1995). Expansion of boundaries and satellite accounts. In J. W. Kendrick (Ed.), *The new system of national accounts* (pp. 91–113). Kluwer: Boston

Elliott, G., Rothenberg, T. J., & Stock, J. H. (1996). Efficient tests for an autoregressive unit root. *Econometrica, 64*(4), 813–836

Engle, R. F. (1982). Autoregressive conditional hetroscedasticity with estimates of the variance of United Kingdom inflation. *Econometrica, 50*, 987–1008

Engle, R. F., & Granger, C. (1987). Co-integration and error correction: Representation, estimation and testing. *Econometrica, 55*, 251–276

Engle, R. F., & Granger C. W. J. (1991). Co-integration and error correction: Representation, estimation, and testing. In R. F. Engle, & C. W. J. Granger (Eds.), Long-run economic relationships. Readings in cointegration (pp. 81–111). Oxford: Oxford University Press

Epstein, R. J. (1987). *A history of econometrics*. Amsterdam: North Holland

Ericsson, N. R., & Irons, J. S. (1995). The Lucas Critique in practice: Theory without measurement. In K. D. Hoover (Ed.), *Macroeconometrics: Developments, tensions, and prospects* (pp. 263–311). Boston, MA: Kluwer

Ermini, L., & Hendry, D. F. (2008). Log income versus linear income: An application of the encompassing principle, *Oxford Bulletin of Economics and Statistics, 70*, 807–827

Evans, M. K. (1969). *Macroeconomic activity: Theory, forecasting, and control; an econometric approach*. New York: Harper & Row

Evans, M. K., & Klein, L. R. (1967). *The Wharton Econometric Forecasting Model. Programmed by George R. Schink*. Philadelphia: Economics Research Unit, University of Pennsylvania

Evans, M. K., & Klein, L. R. (1968). *The Wharton Econometric Forecasting Model. Programmed by George R. Schink (2nd enlarged ed.)*. Philadelphia: Economics Research Unit, University of Pennsylvania

Ewald, W. (1996). *From Kant to Hilbert*. Oxford: Oxford University Press

Farebrother, R. W. (1980). The Durbin-Watson test for serial correlation where there is no intercept in the regression. *Econometrica, 48*, 1553–1563

Farrell, M. J. (1957). The measurement of productive efficiency (Part III). *Journal of the Royal Statistical Society, Series A (General), 120*, 253–290

Fisher, F. M. (1970). Tests of equality between sets of coefficients in two linear regressions: An expository note. *Econometrica, 38*(2), 361–366

Fogler, H. R., & Ganapathy, S. (1982). *Financial econometrics for researchers in finance and accounting* (Vol. 12, p. 212). Englewood Cliffs, NJ: Prentice-Hall

Foschi, P., Belsley, D. A., & Kontoghiorghes, E. J. (2003). A comparative study of algorithms for solving seemingly unrelated regressions models. *Computational Statistics and Data Analysis, 44*, 3–35

Foschi, P., & Kontoghiorghes, E. J. (2003). Estimation of VAR models: Computational aspects. *Computational Economics, 21*, 3–22

Foss, M. F. (1983). *The U.S. national income and product accounts. Studies in income and wealth* (Vol. 47). Chicago: University of Chicago Press

Foster, D. P., & Stine, R. A. (2006). Honest confidence intervals for the error variance in stepwise regression. *Journal of Economic and Social Measurement, 31*(1), 89–102

Fox, K. A. (1956). Econometric models of the United States. *Journal of Political Economy, 64*(2), 128–142

Francis, I. (1973). A comparison of several analysis of variance programs. *Journal of the American Statistical Association, 68*, 860–865

Francis, I. (Ed.). (1981). *Statistical software: A comparative review*. New York: North Holland

Francis, I., Heiberger, R., & Velleman, P. F. (1975). Criteria and Considerations in the evaluation of statistical program packages. *American Statistician, 29*, 52–56

Fried, S. S. (1984). Evaluating 8087 performance on the IBM PC. *Byte, 9*(9), 197–208

Fromm, G., & Klein, L. R. (1975). *The Brookings model: Perspective and recent developments*. Amsterdam: North Holland

Fuller, W. A. (1976). *Introduction to statistical time series*. New York: Wiley

Galler, B. A. (1986). The IBM 650 and the Universities. *Annals of the History of Computing, 8*(1), 36–38

Ghysels, E., & Perron, P. (1993). The effect of seasonal adjustment filters on tests for a unit root. *Journal of Econometrics, 55*, 57–98

Giaquinto, M. (2004). *The search for certainty. A philosophical account of foundations of mathematics*. Oxford: Clarendon

Gilbert, C. L. (1986). Professor Hendry's econometric methodology. *Oxford Bulletin of Economics and Statistics, 48*, 283–307

Gilbert, C. (1988). *The development of econometrics in Britain since 1945*. Oxford: University of Oxford

Gilbert, C. L. (1989). LSE and the British approach to time series econometrics. *Oxford Economic Papers, 41*(1), 108–128

Gilbert, C. L. (1991). Richard Stone, demand theory and the emergence of modern econometrics. *Economic Journal, 101*, 288–302

Gilbert, C. L., & Qin, D. (2006). The first fifty years of modern econometrics. In T. C. Mills, & K. Patterson (Eds.), *Palgrave handbook of econometrics*. Basingstoke: Palgrave Macmillan

Glesjer, H. (1969). A new test for heteroscedasticity. *Journal of the American Statistical Association, 64*, 316–323

Gödel, K. (1992). *On formally undecidable propositions of principia mathematica and related systems*. New York: Dover Publications

Godfrey, L. G. (1978). Testing for multiplicative heteroskedasticity. *Journal of Econometrics, 8*(2), 227–236

Godfrey, L. G. (1988). *Misspecification tests in econometrics: The Lagrange multiplier principle and other approaches*. Econometric Society monographs (Vol. 16, xii, 12, p. 252) Cambridge: Cambridge University Press

Godfrey, L. G., & Orme, C. D. (1994). The sensitivity of some general checks to omitted variables in the linear model. *International Economic Review, 35*(2), 489–506

Goldberger, A. S. (1964). *Econometric theory*. New York: Wiley

Goldberger, A. S. (2004). Econometric computation by hand. *Journal of Economic and Social Measurement, 29*, 115–118

Goldberger, A. S., & Hofer, V. (1962). *A Users guide to the multiple regression program RGR*. Systems Formulation and Methodology Workshop Paper 6207. Madison, WI: Social Systems Research Institute, University of Wisconsin

Goldfeld, S. M., & Quandt, R. E. (1972). *Nonlinear methods in econometrics*. Contributions to economic analysis (Vol. 77, xi, p. 280). Amsterdam: North-Holland

Golub, G. H., & Van Loan, C. F. (1996). *Matrix computations*. (3rd ed.). Baltimore: Johns Hopkins University Press

Gong, G., & Semmler, W. (2006). *Stochastic dynamic macroeconomics. Theory and empirical evidence*. Oxford: Oxford University Press

Goodnight, J. H. (1979). A tutorial on the sweep operator. *American Statistician, 33*, 149–158

Gordon, R. A. (1975). Discussion. In G. Fromm, & L. R. Klein (Eds.), *The Brookings model: Perspective and recent developments* (pp. 31–34). Amsterdam: North-Holland

Gourieroux, C., & Jasiak, J. (2001). Financial econometrics: Problems, models, and methods. *Princeton series in finance* (Vol. 11, p. 513). Princeton, NJ: Princeton University Press

Granger, C. W. J. (1981). Some properties of time series data and their use in econometric model specification. *Journal of Econometrics, 16*, 121–130

Granger, C. W. J. (1991). *Developments in the study of cointegrated economic variables*. In R. F. Engle, & C. W. J. Granger (Eds.), Long-run economic relationships. Readings in cointegration (pp. 65–80). Oxford: Oxford University Press

Granger, C. W. J. (1992). Fellow's opinion: Evaluating economic theory. *Journal of Econometrics, 51*, 3–5

Granger, C., & Newbold, P. (1974). Spurious regressions in econometrics. *Journal of Econometrics, 2*, 111–120

Greene, W. H. (2000). *Econometric analysis*. Upper Saddle River, NJ: Prentice-Hall

Greene, W. H. (2003a). *Econometric analysis* (5th ed.). Upper Saddle River, NJ: Prentice-Hall

Greene, W. H. (2003b). *Email re randomly varying parameters*. C.G. Renfro, Recipient

Greene, W. H. (2004). Reply to "a comparative study of alternative econometric packages with an application to Italian deposit rates", by G. Bruno and R. De Bonis. *Journal of Economic and Social Measurement, 29*, 297–305

Greene, W. H. (2008). *Econometric analysis* (6th ed.). Upper Saddle River, NJ: Pearson Prentice-Hall

Gregory, A. W., & Veall, M. R. (1985). Formulating Wald tests of nonlinear restrictions. *Econometrica, 53*(6), 1465–1468

Gregory, A. W., & Veall, M. R. (1987). Formulating Wald test of the restrictions implied by the rational expectations hypothesis. *Journal of Applied Econometrics, 2*(1), 61–68

Grier, D. A. (1996). The ENIAC, the verb "to program" and the emergence of digital computers. *IEEE Annals of the History of Computing, 18*(1), 51–55

Griliches, Z. (1985). Data and econometricians – The uneasy Alliance. *American Economic Review, 75*(2), 196–200

Griliches, Z. (1986). Economic data issues. In Z. Griliches & M. D. Intriligator (Eds.), *Handbook of econometrics*. New York: North-Holland

Griliches, Z. (1994). Productivity, R&D and the data constraint. *American Economic Review, 84*, 1–23

Grunfeld, Y. (1958). The determinants of corporate investment. In *Economics*. Chicago: University of Chicago

Haavelmo, T. (1943). The statistical implications of a system of simultaneous equations. *Econometrica, 11*, 1–12

Haavelmo, T. (1944). The probability approach in econometrics. *Econometrica, 12*(Suppl. 3–6), 1–115

Haavelmo, T. (1958). The role of the econometrician in the advancement of economic theory. *Econometrica, 26*(3), 351–357

Hahn, F. H. (1970). Some adjustment problems. *Econometrica, 38*(1), 1–17

Haldrup, N., & Jansson, M. (2006). Improving size and power in unit root testing. In T. C. Mills, & K. Patterson (Eds.), *Palgrave handbook of econometrics* (pp. 252–277). New York: Macmillan

Hall, R. E. (2003). *Email re history of econometric software development*. B. H. Hall and C. G. Renfro, Recipients

Hall, S. (1995). Macroeconomics and a bit more reality. *Economic Journal, 105*, 974–988

Hall, S. G. (1989). Maximum likelihood estimation of cointegration vectors: An example of the Johansen procedure. *Oxford Bulletin of Economics and Statistics, 51*, 213–218

Hall, S. G. (2003). *Email message*. C.G. Renfro, Recipient

Hamilton, J. D. (1994). *Time series analysis*. Princeton: Princeton University Press

Hannan, E. J., & Quinn, B. (1979). Determination of the order of an autoregression. *Journal of the Royal Statistical Society, Series B, 41*, 190–195

Hansen, B. (1992). Testing for parameter instability in linear models. *Journal of Policy Modeling, 14*, 517–533

Hansen, B. E. (1995). Rethinking the univariate approach to unit root testing: Using covariates to increase power. *Econometric Theory, 11*, 1148–1171

Hansen, P. R., & Johansen, S. (1998). *Workbook on cointegration. Advanced texts in econometrics* (p. 161). Oxford: Oxford University Press

Harrison, T., & Renfro, C. G. (2004). TSX and TSE: A proposal for an extended time series data interchange and transfer format. *Journal of Economic and Social Measurement, 28*, 339–358

Hart, B. I., & von Neumann, J. (1942). Tabulation of the probabilities for the ratio of the mean square successive difference to the variance. *Annals of Mathematical Statistics, 13*, 207–214

Harvey, A. C. (1976). Estimating regression models with multiplicative heteroscedasticity. *Econometrica, 44*, 461–465

Harvey, A. C. (1981). *The econometric analysis of time series*. New York: Wiley

Harvey, A. C. (1990). *The econometric analysis of time series*. *LSE handbooks in economics* (2nd ed.). Cambridge, MA: MIT

Hausman, J. (1978). Specification tests in econometrics. *Econometrica, 46*: 1251–1272

Heckman, J. J. (2000). Microdata, heterogeneity and the evaluation of public policy. *Economic Sciences*, 255–322

Hendry, D. F. (1970). The estimation of economic models with autoregressive errors. Department of Economics, London School of Economics, University of London, London

Hendry, D. F. (1976). The structure of simultaneous equation estimators. *Journal of Econometrics, 4*, 51–88

Hendry, D. F. (1986). Empirical modeling in dynamic econometrics. *Applied Mathematics and Computation, 20*, 201–236

Hendry, D. F. (1987). Practitioners' corner: Using PC-GIVE in econometrics teaching. *Oxford Bulletin of Economics and Statistics, 48*(1), 87–98

Hendry, D. F. (1993). *Econometrics: Alchemy or science?: Essays in econometric methodology*. Oxford: B. Blackwell
Hendry, D. F. (1995). *Dynamic econometrics*. Oxford: Oxford University Press
Hendry, D. F. (2000). *Econometrics: Alchemy or science?: Essays in econometric methodology* (New ed.). Oxford: Oxford University Press
Hendry, D. F. (2003a). *Email re final version of software survey paper*. C. G. Renfro, Recipient, Oxford
Hendry, D. F. (2003b). J. Denis Sargan and the origins of the LSE econometric methodology. *Econometric Theory, 19*, 457–480
Hendry, D. F., & Doornik, J. A. (1999a). *Empirical econometric modelling using PcGive* (Vol. I). London: Timberlake Consultants Ltd
Hendry, D. F., & Doornik, J. A. (1999b). The impact of computational tools on time-series econometrics. In T. Coppock (Ed.), *Information technology and scholarship* (pp. 257–269). London: British Academy
Hendry, D. F., & Morgan, M. S. (1995). *The foundations of econometric analysis* (Vol. xvi, p. 558). Cambridge, New York: Cambridge University Press
Hendry, D. F., & Richards, J. F. (1989). Recent developments in the theory of encompassing. In B. Cornet, & H. Tulkens (Eds.), *Contributions to operations research and economics. The XXth Anniversary of CORE* (pp. 393–440). Cambridge, MA: MIT
Hendry, D. F., & Srba, F. (1980). AUTOREG: A computer program library for dynamic econometric models with autoregressive errors. *Journal of Econometrics, 12*, 85–102
Herbert, R. D. (2004). Modelling programming languages–appropriate tools? *Journal of Economic and Social Measurement, 29*, 321–338
Herwartz, H. (2006). Testing for random effects in panel data under cross sectional error correlation – A bootstrap approach to the Breusch Pagan test. *Computational Statistics and Data Analysis, 50*, 3567–3591
Hicks, J. (1990). The unification of macroeconomics. *Economic Journal, 100*, 528–538
Higham, N. (2002). *Accuracy and stability of numeric algorithms* (2nd ed.). Philadelphia: SIAM
Hill, R. C. (1989). *Learning econometrics using Gauss*. New York: Wiley
Hill, R. C., & Griffiths, W. E. (1993). *Learning SAS: A computer handbook for econometrics: A guide to programming for Griffiths/Hill/Judge learning and practicing econometrics, and Judge/Hill/Griffiths/Lèutkepohl/Lee Introduction to the theory and practice of econometrics* (2nd ed.). New York: Wiley
Hjort, N. L., & Koning, A. (2002). Tests for constancy of model parameters over time. *Journal of Nonparametric Statistics, 14*(1–2), 113–132
Hocking, R. R. (1983). Developments in linear regression methodology: 1959–1982. *Technometrics, 25*(3), 219–230
Holland, P. W., & Welsch, R. E. (1977). Robust regression using iteratively reweighted least-squares. Communications in Statistics, *6*, 813–827
Holt, C. A. (2003). Economic science: An experimental approach for teaching and research. *Southern Economic Journal, 69*(4), 754–771
Hood, W. C., & Koopmans, T. C. (1953). *Studies in econometric method. Cowles Commission Monograph No. 14*. New York: Wiley
Hoover, K. D. (2006). The methodology of econometrics. In T. C. Mills & K. Patterson (Eds.), *Palgrave handbook of econometrics* (pp. 61–87). Basingstoke: Palgrave Macmillan
Hoover, K. D., & Perez, S. J. (1999). Data mining reconsidered: encompassing and the general-to-specific approach to specification search. *Econometrics Journal, 2*, 167–191
Hotelling, H. (1943). Some new methods in matrix calculation. *Annals of Mathematical Statistics, 14*(1), 1–34
Householder, A. S. (1974). *Theory of matrices in numerical analysis*. New York: Dover
Householder, A. S. (2006). *Principles of numerical analysis*. Mineola, NY: Dover
Houthakker, H. S. (1951). Some calculations on electricity consumption in Great Britain Part III. *Journal of the Royal Statistical Society, Series A, 114*, 351–371

Howrey, E. P. (1971). Stochastic properties of the Klein-Goldberger model. *Econometrica, 39*(1), 73–87

Howrey, E. P. (1978). The use of preliminary data in econometric forecasting. *Review of Economics and Statistics, 60*(2), 193–200

Howrey, E. P. (1996). Forecasting GNP with noisy data: A case study. *Journal of Economic and Social Measurement, 22*(2), 181–200

Intriligator, M. D. (1978). *Econometric models, techniques, and applications* (Vol. xv, p. 638). Englewood Cliffs, NJ: Prentice-Hall

Jarque, C. M., & Bera, A. K. (1980). Efficient tests for normality, homoscedasticity, and serial independence of regression residuals. *Economics Letters, 6*, 255–259

Jeffreys, H., & Zellner, A. (1989). Bayesian analysis in econometrics and statistics: Essays in honor of Harold Jeffreys (p. x, 474). Malabar, FL: R. E. Krieger

Johansen, S. (1988). Statistical analysis of cointegrating vectors. *Journal of Economic Dynamics and Control, 12*, 231–254

Johansen, S. (1991). Estimation and hypothesis testing of cointegration in Gaussian vector autoregressive models. *Econometrica, 59*, 1551–1580

Johansen, S. (1995). *Likelihood-based inference in cointegrated vector autoregressive models. Advanced texts in econometrics* (p. x, 267). Oxford: Oxford University Press

Johansen, S., & Juselius, K. (1990). Maximum likelihood estimation and inference on cointegration, with applications to the demand for money. *Oxford Bulletin of Economics and Statistics, 52*, 169–210

Johansen, S., & Juselius, K. (1992). Testing structural hypotheses in a multivariate cointegration analysis of the PPP and the UIP for the UK. *Journal of Econometrics, 53*, 211–244

Johnston, J. (1963). *Econometric methods*. New York: McGraw-Hill

Johnston, J. (1984). *Econometric methods* (3rd ed., Vol. 8, p. 568). New York: McGraw-Hill

Johnston, J., & DiNardo, J. (1997). *Econometric methods* (4th ed.). New York: McGraw-Hill

Judge, G. G. (1985). The theory and practice of econometrics (2nd ed.). In *Wiley series in probability and mathematical statistics*. New York: Wiley

Keenan, T. A. (1963). *Sixth survey of university computing facilities, Report No. 12*. Rochester, NY: Computing Center, University of Rochester

Kelejian, H. H. (1982). An extension of a standard test for heteroscedasticity to a systems framework. *Journal of Econometrics, 20*, 325–333

Keller, W. J., & Wansbeek, T. (1983). Private consumption expenditures and price index numbers for the Netherlands. *Review of Public Data Use, 11*(4), 311–314

Kendrick, D. A. (1982). A relational database for the US economy. In C. P. Kindleberger, & G. di Tella (Eds.), *Economics in the long view: Essays in honor of W.W. Rostow, Volume 3: Applications and cases* (pp. 63–82, Part II, Chapter 3). London: Macmillan

Kendrick, D. A., & Meeraus, A. (1983). Special issue on modeling languages. *Journal of Economic Dynamics and Control, 5*(1)

Kendrick, J. W. (1995). *The new system of national accounts.* Boston: Kluwer

Kenessey, Z. (1994). *The accounts of nations.* Amsterdam: IOS

Keuzenkamp, H. A., & Magnus, J. R. (1995). On tests and significance in econometrics. *Journal of Econometrics, 67,* 5–24

Keynes, J. M. (1936). *The general theory of employment, interest, and money.* London: Macmillan

Keynes, J. M. (1940). How to pay for the war. *A radical plan for the chancellor of the exchequer.* London: Macmillan

Kim, K. S., Chung, D., Kim, J. H., & Han®guk Kaebal Yæon®guwæon. (1979). *Time Series Processor: A computer language for econometrics and statistics interfaced with KDI data base* (p. 5, 221, Rev. ed.). Seoul, Korea: Korea Development Institute

King, G. (1995). Replication, replication. *PS: Political Science and Politics, 28*(3), 444–452

Kirsch, J. (1979). *Troll primer* (3rd ed.). Cambridge, MA: MIT Center for Computational Research in Economics and Management Science

Kiviet, J. F. (1986). On the rigour of some misspecification tests for modelling dynamic relationships. *Review of Economic Studies, 53,* 241–261

Kiviet, J. F., & Phillips, G. D. A. (1986). Testing strategies for model specification. *Applied Mathematics and Computation, 20,* 237–269

Klein, L. R. (1946). A post-mortem on transition predictions of national product. *Journal of Political Economy, 54*(4), 289–308

Klein, L. R. (1950). *Economic fluctuations in the United States 1921–1941.* In *Cowles Commission Monograph* (Vol. 11). New York: Wiley

Klein, L. R. (1953). *A textbook of econometrics.* New York: Row, Peterson

Klein, L. R. (1960). Single equation vs. equation systems methods of estimation in economics. *Econometrica, 28,* 866–871

Klein, L. R. (1975). Research contributions of the SSRC-Brookings econometric model project–A decade in review. In G. Fromm, & L. R. Klein (Eds.), *The Brookings model: Perspective and recent developments* (pp. 11–29). Amsterdam: North-Holland

Klein, L. R., & Goldberger, A. S. (1955). An econometric model of the United States, 1929–1952. *Contributions to economic analysis* (p. 9). Amsterdam: North Holland

Klein, L. R., & Kosobud, R. F. (1961). Some econometrics of growth: Great ratios in economics. *Quarterly Journal of Economics, 75,* 173–198

Kleiner, M. M. (1980). Statistical uses of administrative records for labor market research. *Review of Public Data Use, 8*(1), 43–50

Kmenta, J. (1986). *Elements of econometrics* (2nd ed.). New York: Macmillan

Knusel, L. (1995). On the accuracy of statistical distributions in GAUSS. *Computational Statistics and Data Analysis, 20,* 699–702

Knuth, D. E. (1970). Von Neumann's first computer program. *ACM Computing Surveys, 2*(4), 247–260

Knuth, D. E. (1986). The IBM 650: An appreciation from the field. *Annals of the History of Computing, 8*(1), 50–54

Knuth, D. E. (1996). *Selected papers on computer science.* Cambridge: Cambridge University Press

Knuth, D. E. (1998). *The art of computer programming.* (3rd ed.). (Vol. 2). Reading, MA: Addison-Wesley

Koenker, R., & Bassett, G. (1982). Robust tests for heteroscedasticity based on regression quantiles. *Econometrica, 50*(1), 43–62

Kontoghiorghes, E. J. (2000). Inconsistencies in SURE models: Computational aspects. *Computational Economics, 16,* 63–70

Kontoghiorghes, E. J., & Dinenis, E. (1997). Computing 3SLS solutions of simultaneous equation models with a possible singular variance-covariance matrix. *Computational Economics, 10,* 231–250

Koopmans, T. C. (1947). Measurement without theory. *Review of Economics and Statistics, 29*(2), 161–172

Koopmans, T. C. (1950). *Statistical inference in dynamic economic models. Cowles Commission Monograph No. 10*. New York: Wiley

Kristensen, D., & Linton, O. (2006). Notes and problems: A closed form estimator for the GARCH(1,1) model. *Econometric Theory, 22,* 323–337

Krolzig, H.-M., & Hendry, D. F. (2001). Computer automation of general-to-specific model selection procedures. *Journal of Economic Dynamics and Control, 25,* 831–866

Kuan, C. -M., & Hornik, K. (1995). The generalized fluctuation test: A unifying view. *Econometric Reviews, 14,* 135–161

Kuh, E. (1972). Some notes on the research program at the NBER computer research center for economics and management science. *Annals of Economic and Social Measurement, 1*(2), 233–36

Kuh, E. (1974). Introduction. *Annals of Economic and Social Measurement, 3*(4), i–iii

Kuh, E., & Neese, J. (1982). Econometric model diagnostics. In G. C. Chow, & P. Corsi (Eds.), *Evaluating the reliability of macro-economic models.* New York: Wiley

Kuh, E., & Welsch, R. E. (1980). Econometric models and their assessment for policy: Some new diagnostics applied to the translog energy demand in manufacturing. In S. Gass (Ed.), *Proceedings of the Workshop on Validation and Assessment Issues of Energy Models* (pp. 445–475). Washington, DC: National Bureau of Standards

Kurabashi, Y. (1994). Keynes' how to pay for the war and its influence on postwar national accounting. In Z. Kenessey (Ed.), *The accounts of nations* (pp. 93–108). Amsterdam: IOS

Kwiatkowski, D., Phillips, P. C. B., Schmidt, P., & Shin, Y. (1992). Testing the null hypothesis of stationarity against the alternative of a unit root. *Journal of Econometrics, 54,* 159–178

Kydland, F. E., & Prescott, E. C. (1991). The econometrics of the general equilibrium approach to business cycles. *Scandinavian Journal of Economics, 93*(2), 161–178

Kydland, F. E., & Prescott E. C. (1996). *The computational experiment: An econometric tool. Journal of Economic Perspectives, 10*(1), 69–85

Landefeld, J. S., Whichard, O. G., & Lowe, J. H. (1993). Alternative frameworks for US international transactions. *Survey of Current Business, 73*(12), 50–61

Lane, T. (Ed.). (1979). *TROLL Reference Manual (Standard System)* (2nd ed.). Cambridge, MA: MIT Center for Computational Research in Economics and Management Science, MIT Information Processing Services

Lawson, T. (1989). Realism and instrumentalism in the development of econometrics. *Oxford Economic Papers, 41*(1), 236–258

Leamer, E. E. (1978). *Specification searches: Ad hoc inference with non-experimental data.* New York: Wiley

Leamer, E. E. (1983). Let's take the con out of econometrics. *American Economic Review, 73,* 31–43

Leamer, E. E. (1994). *Sturdy econometrics. Economists of the twentieth century* (Vol. xvi, p. 366). Aldershot: Brookfield

Leontief, W. (1971). Theoretical assumptions and nonobserved facts. *American Economic Review, 61*(1), 1–7

Leontief, W. W. (1936). Quantitative input and output relations in the economic system of the United States. *Review of Economic Statistics, 18,* 105–125

Leontief, W. W. (1941). *The structure of the American economy.* New York: Oxford University Press

Leontief, W. W. (1948). Computational problems arising in connection with economic analysis of industrial relationships. *Proceedings of a Symposium on Large-Scale Digital Calculating Machinery.* Cambridge, MA: Harvard University Press

Leontief, W. W. (1953). Preface to the second edition. In *The structure of the American economy*. New York: Oxford University Press

Levy, F. K. (1965). An adaptive production function. *American Economic Review, 55*(1/2), 386–396

Lilien, D. (2000). Econometric software reliability and nonlinear estimation in EViews: comment. *Journal of Applied Econometrics, 15*(1), 107–110

Ljung, G. M., & Box, G. E. P. (1979). On a measure of lack of fit in time series models. *Biometrika, 66,* 265–270

Lofgren, H., Harris, R. L., Robinson, S., Thomas, M., & El-Said, M. *A Standard Computable General Equilibrium (CGE) Model in GAMS.* 2002. Washington, DC: International Food Policy Research Institute

Longley, J. W. (1967). An appraisal of least squares programs for the electronic computer from the point of view of the user. *Journal of the American Statistical Association, 62,* 819–841

Lovell, M. C. (1983). Data mining. *Review of Economics and Statistics, 65*(1), 1–12

Lovell, M. C., & Selover, D. D. (1994). Econometric software accidents. *Economic Journal, 104,* 713–725

Mackie-Mason, J. K. (1992). Econometric software: A user's view. *Journal of Economic Perspectives, 6,* 165–188

MacKinnon, J. G. (1989). Heteroskedasticity-robust tests for structural change. *Empirical Economics, 14,* 77–92

MacKinnon, J. G. (1992). Model specification tests and artificial regressions. *Journal of Economic Literature, 30,* 102–146

Macon, N. (1963). *Numerical analysis.* New York: Wiley
Maddala, G. S. (1977). *Econometrics.* New York: McGraw-Hill
Maddy, P. (1988). Believing the axioms. I. *Journal of Symbolic Logic, 53*(2), 481–511
Mann, H. B., & Wald, A. (1943). On the statistical treatment of linear stochastic difference equations. *Econometrica, 11*(3/4), 173–220
Manser, M. E. (1992). Data needs for analyzing labor force status and transitions. *Journal of Economic and Social Measurement, 18,* 47–66
Marschak, J. (1953). Economic measurements for policy and prediction. In W. C. Hood & T. C. Koopmans (Eds.), *Studies in econometric method. Cowles foundation monograph* (Vol. 14, pp. 1–26). New York: Wiley
Marschak, J. (1966). Forward: A remark on econometric tools. In C. F. Christ (Ed.), *Econometric models and methods.* New York: Wiley
Marshall, A. (1907). The social possibilities of economic chivalry. *Economic Journal, 17,* 7–29
Marshall, A. (1920). *Principles of economics* (8th ed.). London: Macmillan
McAleer, M., & Pesaran, M. H. (1986). Statistical inference in non-nested econometric models. *Applied Mathematics and Computation, 20,* 271–311
McCarthy, M. D. (1992). The Cowles commission, the brookings project, and the econometric services industry: Successes and possible new directions: A personal view. *Econometric Theory, 8,* 383–401
McCloskey, D. (1985). *The rhetoric of economics.* Madison: University of Wisconsin Press
McCracken, M. C. (1966). *DATABANK system: Users' manual.* Ottawa: Economic Council of Canada
McCracken, M. C. (1967a). A computer system for econometric research. *Social Science Information, 5,* 151–158
McCracken, M. C. (1967b). *Data administration in an information system. Conference on Government Information Systems.* Ottawa: Economic Council of Canada
McCracken, M. C., & May, K. (2004). Issues in the development of early solution facilities for very large macroeconometric models. *Journal of Economic and Social Measurement, 29,* 167–172
McCracken, M. C., & Sonnen, C. A. (1972). *A system for large econometric models: Management, estimation, and simulation. Proceedings of the Association for Computing Machinery Annual Conference, August 1972. Boston, MA*: Association for Computing Machinery
McCullough, B. D. (1997). A review of RATS v4.2: Benchmarking numeric accuracy. *Journal of Applied Econometrics, 12,* 181–190
McCullough, B. D. (1998). Assessing the reliability of statistical software: Part I. *American Statistician, 52*(4), 358–366
McCullough, B. D. (1999a). Assessing the reliability of statistical software: Part II. *American Statistician, 53*(2), 149–159
McCullough, B. D. (1999b). Econometric software reliability: EViews, LIMDEP, SHAZAM, and TS. *Journal of Applied Econometrics, 14*(2), 191–202
McCullough, B. D. (2000a). Is it safe to assume that software is accurate? *International Journal of Forecasting, 16,* 349–357
McCullough, B. D. (2000b). The accuracy of Mathematica 4.0 as a statistical package. *Computational Statistics, 15,* 279–299
McCullough, B. D. (2004). Wilkinson's tests and econometric software. *Journal of Economic and Social Measurement, 29,* 261–270
McCullough, B. D., McGeary, K. A., & Harrison, T. D. (2006a). Lessons from the JCMB archive. *Journal of Money, Credit, and Banking, 38*(4), 1093–1107
McCullough, B. D., McGeary, K. A., & Harrison, T. D. (2006b). Lessons from the Archive(s) at the Fed. St. Louis Review (and Other Journals). Philadelphia: Drexel
McCullough, B. D., & Renfro, C. G. (1998). Benchmarks and software standards: A case study of GARCH procedures. *Journal of Economic and Social Measurement, 25,* 59–71
McCullough, B. D., & Renfro, C. G. (2000). Some numerical aspects of nonlinear estimation. *Journal of Economic and Social Measurement, 26,* 63–77

McCullough, B. D., Renfro, C. G., & Stokes, H. H. (2006). Re-examining 50-year-old OLS estimates of the Klein-Goldberger model. *Statistica Neerlandica, 60*(3), 181–193

McCullough, B. D., & Vinod, H. D. (1999). The numerical reliability of econometric software. *Journal of Economic Literature, 37*(2), 633–665

McCullough, B. D., & Vinod, H. D. (2003). Verifying the solution from a nonlinear solver: a case study. *American Economic Review, 93*(3), 873–892

McCullough, B. D., & Wilson, B. (1999). On the accuracy of statistical procedures in Microsoft Excel 97. *Computational Statistics & Data Analysis, 31,* 27–37

McCullough, B. D., & Wilson, B. (2002). On the accuracy of statistical procedures in Microsoft Excel 2000 and Excel XP. *Computational Statistics & Data Analysis, 40,* 713–721

McCullough, B. D., & Wilson, B. (2005). On the accuracy of statistical procedures in Microsoft Excel 2003. *Computational Statistics and Data Analysis , 49*(4), 1244–1252

McGuckin, R. H., & Nguyen, S. V. (1990). Public use microdata: Disclosure and usefulness. *Journal of Economic and Social Measurement, 16*, 19–40

McKelvey, M. J., & de Leeuw, F. (1982). Price reports by business firms: Comparison with official price indexes. *Review of Public Data Use, 10*(3), 141–152

McKenzie, C. (1998). A review of Microfit 4.0. *Journal of Applied Econometrics, 13*(1), 77–89

McKibben, W. J. (1998). Forecasting the World Economy Using Dynamic Intertemporal General Equilibrium Multi-Country Models. In *Business Symposium on Economic Forecasting*, 1 October 1998, Sydney

McLeod, A. I., & Li, W. K. (1983). Diagnostic checking ARMA time series models using squared-residual autocorrelations. *Journal of Time Series Analysis, 4*, 269–273

Meade, J. E., & Stone, R. (1941). The construction of tables of national income, expenditure, savings and investment. *Economic Journal, 51*, 216–233

Meeraus, A. (1983). An algebraic approach to modeling. *Journal of Economic Dynamics and Control, 5*, 81–108

Mendelssohn, R. C. (1980). The New on-line BLS database and information system. In J. D. Fischer (Ed.), *Readings in business and economic research management: Execution and enterprise* (pp. 86–89). Madison, WI: Office of Research Administration, School of Business, University of Wisconsin

Mills, T. C., & Patterson, K. (Eds.). (2006). *Palgrave Handbook of Econometrics*. Basingstoke: Palgrave Macmillan

Mirowski, P., & Sklivas, S. (1991). Why econometricians don't replicate (although they do reproduce). *Review of Political Economy, 3*(2), 146–163

Mizon, G. E. (1977). Model selection procedures. In M. J. Artis, & M. J. Nobay (Eds.), *Studies in modern economic analysis*. Oxford: Basil Blackwell

Mizon, G. E. (1984). The encompassing approach in econometrics. In D. F. Hendry, & K. F. Wallis (Eds.), *Econometrics and quantitative economics* (pp. 135–172). Oxford: Basil Blackwell

Moore, G. H. (1958). Forecasting industrial production: A comment. *Journal of Political Economy, 66*(1), 74–83

Morgan, G. H. T. (1978). *Topics in econometric model research* (p. 48). Wellington: Reserve Bank of New Zealand

Morgan, M. S. (1990). The history of econometric ideas. In *Historical perspectives on modern economics*. Cambridge, UK: Cambridge University Press

Morgenstern, O. (1960). *On the accuracy of economic observations*. Princeton, NJ: Princeton University Press

Muller, M. E., & Wilkinson, G. N. (1976). A standardized form for describing programs, packages and systems for statistical applications. *International Statistical Review, 44*(3), 349–353

Nagel, E., Newman, J. R., & Hofstadter, D. R., (2001) *Gödel's proof.* (Rev. ed.). New York: New York University Press

Nelson, C. R., & Plosser, C. I. (1982). Trends and random walks in macroeconomic time series: Some evidence and implications. *Journal of Monetary Economics, 10*, 139–162

Nerlove, M. (2004). Programming languages: A short history for economists. *Journal of Economic and Social Measurement, 29*, 189–203

Neumann, J. v. (1993/1945). First draft of a report on the EDVAC. *IEEE Annals of the History of Computing, 15*(4), 27–75

Newey, W. K., & West K. D. (1987). A simple, positive semi-definite, heteroskedasticity and autocorrelation consistent covariance matrix. *Econometrica, 55*(3), 703–708

Orcutt, G. (1952). Toward partial redirection of econometrics. *Review of Economics and Statistics, 34*, 195–213

Orcutt, G. H. (1962). Microanalytic models of the United States economy: Need and development. *American Economic Review, 52*, 229–240

Ozturk, F., & Akdeniz, F. (2000). Ill-conditioning and multicollinearity. *Linear Algebra and its Applications, 321*, 295–305

Pagan, A.R. (1987) Three econometric methodologies: a critical appraisal. *Journal of Economic Surveys, 1*, 3–24

Pagan, A. R. (1990). Three econometric methodologies: a critical appraisal. In C. W. J. Granger (Ed.), *Modelling economic series: Readings in economic methodology* (pp. 97–120). Oxford: Clarendon

Pagan, A. R., & Hall, A. D. (1983). Diagnostic tests as residual analysis. *Econometric Reviews, 2*, 159–218

Patil, G. P., Box, G. E. P., & Widen, M. L. (1970). Statistical training and research. *Review of the International Statistical Institute, 38*(3), 402–408

Perron, P. (1989). The great crash, the oil price shock, and the unit root hypothesis. *Econometrica, 57,* 1361–1401

Perron, P. (2006). *Dealing with structural breaks.* In T. C. Mills & K. Patterson (Eds.), *Palgrave handbook of econometrics*. Basingstoke: Palgrave Macmillan

Pesaran, M. H. (1997). The role of economic theory in modelling the long run. *Economic Journal, 107,* 178–191

Pesaran, M. H., & Pesaran, B. (1987). *Microfit. A interactive econometric software package. User manual*. Oxford: Oxford University Press

Pesaran, M. H., & Pesaran, B. (1991). *Microfit 3.0. An interactive econometric software package*. User Manual, Oxford: Oxford University Press

Pesaran, M. H., & Pesaran, B. (1997). *Working with MicroFit 4.0. Interactive econometric analysis*. Oxford: Oxford University Press

Pesaran, M. H., & Slater, L. J. (1980). *Dynamic regression: Theory and algorithms*. Chichester: Ellis Horwood

Phillips, A. W. (1957). Stabilization policy and the time forms of lagged responses. *Economic Journal, 67*, 265–277

Phillips, P. C. B. (1987). Time series regression with a unit root. *Econometrica, 55*, 277–301

Phillips, P. C. B. (2001). The ET interview: Professor Clive Granger. In N. R. S. E. Ghysels, & M. W. Watson (Eds.), *Essays in econometrics* (pp. 28–81). (Collected Papers of Clive W. J. Granger). New York: Cambridge University Press

Phillips, P. C. B., & Hall, V. (2004). Early development of econometric software at the University of Auckland. *Journal of Economic and Social Measurement, 29*, 127–134

Phillips, P. C. B. & Hansen, B. E. (1990). Statistical inference in instrumental variables regression with I(1) processes. *Review of Economic Studies, 57*, 99–125

Phillips, P. C. B., & Perron, P. (1988). Testing for a unit root in time series regressions. *Biometrika, 75,* 335–346

Phillips, P. C. B., & Xiao, Z. (1998). *A primer on unit root testing. Journal of Economic Surveys, 12*(5), 7–54

Pilozzi, G. (1976). A new approach to the dissemination of CANSIM data. *Canadian Statistical Review, 51*, 5–8

Plosser, C. I., Schwert, G. W., & White, H. (1982). Differencing as a test of specification. *International Economic Review, 23*, 535–552

Plott, C. R. (1991). Will economics become an experimental science? *Southern Economic Journal, 57*(4), 901–919

Popkin, J. (1993). An alternative framework for analyzing industrial output. *Survey of Current Business, 73*(11), 50–56

Prais, S. J., & Aitchison, J. (1954). The grouping of observations in regression analysis. *Review of the International Statistical Institute, 22*(1/3), 1–22

Prais, S. J., & Houthakker, H. S. (1955). *The analysis of family budgets*. Cambridge: Cambridge University Press

Press, S. J. (1980). Bayesian computer programs. In A. Zellner (Ed.), Bayesian analysis in econometrics and statistics: *Essays in honor of Harold Jeffreys* (pp. 429–442). Amsterdam: North Holland

Preston, R. S. (2006). *Personal Reflections on the Historical Development and Use of Computer Software Needed to Undertake Econometric Research at the Economics Research Unit, University of Pennsylvania – 1960 to 1969*

Qin, D. (1993). *The formation of econometrics: A historical perspective*. Oxford: Clarendon

Qin, D., & Gilbert, C. L. (2001). The error term in the history of time series econometrics. *Econometric Theory, 17*, 424–450

Quandt, R. E. (1983). Computational Problems and Methods. In Z. Griliches, & M. D. Intriligator (Eds.), *Handbook of econometrics* (pp. 699–764). Amsterdam: North Holland

Ralston, A., & Rabinowitz, P. (2001). *A first course in numerical analysis*. Mineola, New York: Dover

Ramsey, F. P. (1925). The foundations of mathematics. *In Proceedings of the London Mathematical Society*, Series 2, *25*(5), 338–384

Ramsey, F. P. (1928). A mathematical theory of savings. *Economic Journal, 38*, 543–559

Ramsey, J. B. (1969). Tests for specification error in classical linear least squares analysis. *Journal of the Royal Statistical Society, Series B, 31*, 350–371

Renfro, C. G. (1970). *An interactive regression program for the PDP-10. Description and User Manual*. Washington, DC: Brookings Quarterly Model Project, Brookings Institution

Renfro, C. G. (1980a). Economic data base systems: Some reflections on the state of the art. *Review of Public Data Use, 8*(3), 121–140

Renfro, C. G. (1980b). Production, distribution, and use of data: An editorial. *Review of Public Data Use, 8*(4), 295–306

Renfro, C. G. (1981). The Computer as a factor of production in economic data provision and research. Presented at *American Economic Association Annual Meeting,* Washington, DC, 30 December 1981

Renfro, C. G. (1996). On the development of econometric modeling languages: MODLER and its first twenty-five years. *Journal of Economic and Social Measurement, 22*(4), 241–311

Renfro, C. G. (1997a). Economic data base systems: Further reflections on the state of the art. *Journal of Economic and Social Measurement, 23*(1), 43–85

Renfro, C. G. (1997b). Normative considerations in the development of a software package for econometric estimation. *Journal of Economic and Social Measurement, 23*(4), 277–330

Renfro, C. G. (1998). Macroeconometric models and changes in measurement concepts: An overview. *Journal of Economic and Social Measurement, 24*(2), 63–82

Renfro, C. G. (2003). *Building and using a small macroeconometric model: Klein model I as an example*. Philadelphia: Modler Information Technologies Press

Renfro, C. G. (2004a). The early development of econometric modeling languages. *Journal of Economic and Social Measurement, 29*, 145–166

Renfro, C. G. (2004b). Econometric software: The first fifty years in perspective. *Journal of Economic and Social Measurement, 29*(1–3), 9–108

Renfro, C. G. (Ed.). (2004c). A compendium of existing econometric software packages. *Journal of Economic and Social Measurement, 29*, 359–402

Renfro, C. G. (Ed.). (2004d). *Computational econometrics: Its impact on the development of quantitative economics.* Amsterdam: IOS

Renfro, C. G. (2006). Economic and social measurement, research methodology, and computational technologies: An editorial. *Journal of Economic and Social Measurement, 31*(1), 1–32

Renfro, C. G., & Stokes, H. H. (2009). Testing for heteroscedasticity using the Goldfeld-Quandt test and other approaches: The practitioner's perspective. In preparation

Rice, J. (1984). Boundary modification for kernal regression. *Communications in Statistics, Series A, 13*, 893–900

Ritter, J. A. (2000). Feeding the national accounts. *Federal Reserve Bank of St. Louis Review, 82*(2), 11–20

Robinson, S., & Roland-Holst, D. W. (1988). Macroeconomic structure and computable general equilibrium models. *Journal of Policy Modeling, 10*, 353–375

Rosen, S. (1969). Electronic computers: A historical survey. *ACM Computing Surveys, 1*(1), 7–36

Royston, P. (1991). *sg3.2: Shapiro-Wilk and Shapiro-Francia tests. Stata Technical Bulletin, 3*, 19

Rudebusch, G. D. (1998). Do measures of monetary policy in a VAR make sense. *International Economic Review, 39*(4), 907–931

Rushton, S. (1951). On least squares fitting by orthonormal polynomials using the Choleski method. *Journal of the Royal Statistical Society, Series B, 13*(1), 92–99

Rutherford, T. (1999). Applied general equilibrium modeling with MPSGE as a GAMS Subsystem: An overview of the modeling framework and syntax. *Computational Economics, 14*, 1–46

Rycroft, R. S. (1989). Microcomputer software of interest to forecasters in comparative review. *International Journal of Forecasting, 5*, 437–462

Rycroft, R. S. (1993). Microcomputer software of interest to forecasters in comparative review: An update. *International Journal of Forecasting, 9*, 531–575

Rycroft, R. S. (1999). Microcomputer software of interest to forecasters in comparative review: Updated again. *International Journal of Forecasting, 15*(1), 93–120

Sage, A. P., & Melsa, J. L. (1971). *Estimation theory, with applications to communication and control*. New York: McGraw-Hill

Said, E. S., & Dickey, D. A. (1984). Testing for unit roots in autoregressive-moving average models of unknown order. *Biometrika, 71*, 599–607

Saikkonnen, P. (1989). Asymptotic relative efficiency of the classical test statistics under misspecification. *Journal of Econometrics, 42*, 351–369

Saikkonen, P. (1991). Asymptotically efficient estimation of cointegration regressions. *Econometric Theory, 7*, 1–21

Sall, J. P. (1979). SAS econometric and time-series library. Preliminary test version for SAS 79.2. In *SAS Institute Technical Report*. Raleigh, NC: SAS Institute, Inc

Samuelson, P. A. (1970). What makes for a beautiful problem in science. *Journal of Political Economy, 78*(6), 1372–1377

Samuelson, P. A. (1984). Second thoughts on analytical income comparisons. *Economic Journal, 94*, 267–278

Sargan, J. D. (1964). Wages and prices in the United Kingdom: A study in econometric methodology. In: P. E. Hart, G. Mills, & J. K. Whitaker (Eds.), *Econometric analysis for national economic planning* (pp. 25–63). London: Butterworths

Sargan, J. D., & Bhargava, A. (1983). Testing residuals from least squares regression for being generated by the Gaussian random walk. *Econometrica, 51*, 153–74

Sargent, T. J., & Sims, C. A. (1977). Business cycle modeling without pretending to have too much a priori economic theory. In *New Methods in Business Cycle Research* (pp. 45–109). Minneapolis, MN: Federal Reserve Bank of Minneapolis

SAS. (1993). *SAS/ETS user's guide, version 6* (2nd ed.). Cary, NC: SAS Institute

SAS. (1999). *SAS/ETS user's guide, version 8*. Cary, NC: SAS Pub

SAS Institute. (2000). *SAS/ETS software: Changes and enhancements: Release 8.1*. Cary, NC: SAS Institute

Schink, G. R. (2004). Simulation with large econometric models: The quest for a solution. Originally a paper presented to the association for computing machinery meetings, Denver, Colorado, 12 June 1970. *Journal of Economic and Social Measurement, 29*, 135–144

Schwarz, G. (1978). Estimating the dimension of a model. *Annals of Statistics, 6,* 461–464

Schwert, G. W. (1989). Tests for unit roots: A monte carlo investigation. *Journal of Business and Economic Statistics, 7*, 147–159
Shapiro, S. S., & Francia, R. S. (1972). An approximate analysis of variance test for normality. *Journal of the American Statistical Association, 67*, 215–216
Shapiro, S. S., & Wilk, M. B. (1965). An analysis of variance test for normality (complete samples). *Biometrika, 44*(4)
Sheldon, J. W., & Tatum, L. (1973). The IBM card-programmed electronic calculator. In B. Randell (Ed.), *The origins of digital computers. Selected papers* (pp. 229–235). New York: Springer
Shibata, R. (1981). An optimum selection of regression variables. *Biometrika, 68*, 45–54
Shiller, R. (1973). A distributed lag estimator derived from smoothness priors. *Econometrica, 41*(4), 775–788
Shiskin, J., & Eisenpress, H. (1957). Seasonal adjustments by electronic computer methods. *Journal of the American Statistical Association, 52*, 415–449
Simon, H. A. (1955). A behavioral model of rational choice. *Quarterly Journal of Economics, 69*(1), 99–118
Simon, S. D. L. S., & James, P. (1988). Benchmarking numerical accuracy of statistical algorithms. *Computational Statistics and Data Analysis, 7*, 197–209
Sims, C. A. (1980). Macroeconomics and reality. *Econometrica, 48*, 1–48
Slater, L. J. (1962). Regression analysis. *The Computer Journal, 4*(4), 287–291
Slater, L. J. (1967). *Fortran programs for economists. Department of applied economics occasional paper 13*. Cambridge: Cambridge University Press
Slater, L. J. (1972). *More Fortran programs for economists. DAE occasional paper 34*. Cambridge: Cambridge University Press
Slater, L. J. (2004). Recollections on early days in the Cambridge computing laboratory. *Journal of Economic and Social Measurement, 29*, 119–122
Slater, L. J., & Barker, T. S. (1967). *Regression program for Titan*. Cambridge: Department of Applied Economics, University of Cambridge
Slesnick, D. T. (1998). Are our data relevant to the theory? The case of aggregate consumption. *Journal of Business & Economic Statistics, 16*, 52–61
Smith, R. (1998). The development of econometric methods at the DAE. In I. Begg & S. G. B. Henry (Eds.), *Applied economics and public policy* (pp. 88–103). Cambridge: Cambridge University Press
Society for Economic Dynamics and Control. (1981). News Item: Third Conference of the Society for Economic Dynamics and Control (Program). *Journal of Economic Dynamics and Control, 3*, 393–395
Spanos, A. (1986). *Statistical foundations of econometric modelling* (Vol. 23, p. 695). Cambridge; New York: Cambridge University Press
Spanos, A. (1995). On theory testing in econometrics with nonexperimental data. *Journal of Econometrics, 67*, 189–226
Spanos, A. (2006). Econometrics in Retrospect and Prospect. In T. C. Mills & K. Patterson (Eds.), *Palgrave handbook of econometrics* (pp. 3–60). Basingstoke: Palgrave Macmillan
Spiegel, U. (1994). The case of a "Giffin Good". *Journal of Economic Education, 25*(2), 137–147
Spitzer, J. J. (1976). The Keynesian demand for money function revisited. *Journal of Monetary Economics, 2*, 381–387
Stigler, G. J., & Sherwin, R. A. (1985). The extent of the market. *Journal of Law and Economics, 28*, 555–585
Stock, J. H. (1987). Asymptotic properties of least squares estimators of cointegrating vectors. *Econometrica, 55*, 1035–1056
Stock, J. H., & Watson, M. W. (2006). *Introduction to econometrics* (2nd ed.). Boston, MA: Addison-Wesley
Stoer, J., & Bulirsch, R. (1993). *Introduction to numerical analysis* (2nd ed.). New York: Springer
Stokes, H. H. (1997). *Specifying and diagnostically testing econometric models* (2nd ed.). New York: Quorum Books

Stokes, H. H. (2004a). Econometric software as a theoretical research tool. *Journal of Economic and Social Measurement, 29*, 183–188

Stokes, H. H. (2004b). The evolution of econometric software design: A developer's view. *Journal of Economic and Social Measurement, 29*, 205–259

Stokes, H. H. (2004c). On the advantage of using two or more econometric software systems to solve the same problem. *Journal of Economic and Social Measurement, 29*, 307–320

Stokes, H. H. (2005). The sensitivity of econometric results to alternative implementations of least squares. *Journal of Economic and Social Measurement, 30*(1), 9–38

Stokes, H. H. (2008). A benchmark-free approach to access the accuracy of an OLS model calculation in an econometrically uncertain world. *Journal of Economic and Social Measurement, 33*(1), 1–17

Stone, R. (1997). The accounts of society. Nobel Memorial Lecture. *American Economic Review, 87*(6), 17–29

Suits, D. B. (1963). *The Theory and Application of Econometric Models* (Vol. xiii, p. 147). Athens: Center of Economic Research

Summers, L. H. (1991). The scientific illusion in empirical macroeconomics. *Scandinavian Journal of Economics, 93*(2), 129–148

Summers, R. (1965). A capital intensive approach to the small sample properties of various simultaneous equation estimators. *Econometrica, 33*(1), 1–41

Szroeter, J. (1978). A class of parametric tests for heteroscedasticity in linear econometric models. *Econometrica, 46*, 1311–1328

Taeuber, R. C., & Rockwell, R. C. (1982). National social data series: A compendium of brief descriptions. *Review of Public Data Use, 10*(1–2), 23–111

Theil, H. (1954). *Linear aggregation of economic relations*. Amsterdam: North-Holland

Theil, H. (1957). Specification errors and the estimation of economic relationships. *Review of the International Statistical Institute, 25*(1/3), 41–51

Theil, H. (1971). *Principles of econometrics*. New York: Wiley

Thomas, J. J. (1989). The early econometric history of the consumption function. *Oxford Economic Papers, 41*(1), 131–149

Thursby, J. G. (1989). A comparison of several specification error tests for a general alternative. *International Economic Review, 30*, 217–230

Thursby, J. G. (1992). A comparison of several exact and approximate tests for structural shift under heteroscedasticity. *Journal of Econometrics, 53*, 363–386

Tinbergen, J. (1939). *Statistical testing of business cycle theories*. Geneva: League of Nations

Tinbergen, J. (1940). Econometric business cycle research. *Review of Economic Studies, 7*(2), 73–90

Tinbergen, J. (1951). *Econometrics*. London: Allen & Unwin

Tintner, G. (1952). *Econometrics*. New York: Wiley

Triandafillou, P. N. (1975). CANSIM/SIS – A new phase of development. *Canadian Statistical Review, 50*, 4–6

Triplett, J. E. (1991). The federal statistical system's response to emerging data needs. *Journal of Economic and Social Measurement, 17*, 155–178

Triplett, J. E. (1993). Economic concepts for economic classifications. *Survey of Current Business, 73*(11), 45–49

Tsay, R. S. (1986). Nonlinearity tests for time series. *Biometrika, 73*(2), 461–466

Tsay, R. S. (2002). Analysis of financial time series. In *Wiley series in probability and statistics*. New York: Wiley

Turing, AM, (1936). On computable numbers, with an application to the Entscheidungsproblem. *Proceedings of the London Mathematical Society*, *42*(2), 230–265

Turing, A. M. (1937). On computable numbers, with an application to the Entscheidungsproblem: A correction. *Proceedings of the London Mathematical Society*, *43*(2), 544–546

Vanoli, A. (2005). *A history of national accounting*. Amsterdam: IOS

Veall, M. R. (1991). SHAZAM 6.2: A review. *Journal of Applied Econometrics, 6*(3), 317–320

Velleman, P. F., & Welsch, R. E. (1976). On evaluating interactive statistical program packages. *Communications in Statistics, Part B–Simulation and Computation, 5*, 197–208

Vining, R. (1949). Koopmans on the choice of variables to be studied and of methods of measurement. *Review of Economics and Statistics, 31*(2), 77–86

von Neumann, J. (1941). Distribution of the ratio of the mean square successive difference to the variance. *Annals of Mathematical Statistics, 12*(4), 367–395

von Neumann, J. (1942). A further remark concerning the distribution of the ratio of the mean square successive difference to the variance. *Annals of Mathematical Statistics, 13*, 86–88

Waldman, D. (1983). A note on the algebraic equivalence of White's test and a variant of the Godfrey/Bruesch-Pagan test for heteroscedasticity. *Economic Letters, 13*, 197–200

Wallis, K. F. (1972). Testing for fourth order autocorrelation in quarterly regression equations. *Econometrica, 40*, 617–636

Waugh, F. V. (1961). The place of least squares in econometrics. *Econometrica, 29*(3), 386–396

Wetherill, G. B., Curram, J. B., Burman, J. P., Duncombe, P., Hailstone, J. E., & Kempson, R. E. (1985). The design and evaluation of statistical software for microcomputers. *The Statistician, 34*, 391–427

Whistler, D., White, K. J., Wong, S. D., & Bates, D. (2004). *SHAZAM econometrics software. Version 10.* User's reference manual. Vancouver, BC: Northwest Econometrics

White, H. (1980). A hetroscedasticity-consistent convariance matrix estimator and a direct test for hetroscedasticity. *Econometrica, 48*, 817–838

White, H. (1987). Specification testing in dynamic models. In T. F. Bewley (Ed.), *Advances in econometrics* (pp. 1–58). Fifth World Congress. Econometric Society. Cambridge: Cambridge University Press

White, H., Engle, R. F., & Granger, C. W. J. (1999). *Cointegration, causality, and forecasting: A festschrift in honour of Clive W.J. Granger* (Vol. vi, p. 497). Oxford: Oxford University Press

White, K. J. (1978). A General computer for econometric models–SHAZAM. *Econometrica, 46*(1), 239–240

Wickens, M. R. (1997). Comments: A systems approach with the CBS model. *Journal of Applied Econometrics, 12*, 523–527

Wickens, M. R., & Pagan, A. R. (1989). A survey of some recent econometric methods. *Economic Journal, 99*, 962–1025

Wilcox, D. W. (1998). The construction of US consumption data: Some facts and their implication for empirical work. *American Economic Review, 82*, 922–941

Wilkes, M. (1985). *Memoirs of a computer pioneer*. Cambridge, MA: MIT

Wilkes, M. V. (1956). *Automatic digital computers*. London: Methuen & Co

Wilkes, M. V., & Renwick, W. (1949). The EDSAC, an electronic calculating machine. *Journal of Scientific Instruments, 26*, 385–391

Wilkes, M. V., Wheeler, D. J., & Gill, S. (1951). *The preparation of programs for an electronic digital computer: With special reference to the EDSAC and the use of a library of subroutines*. Cambridge, MA: Addison Wesley

Wilkinson, J. H. (1961). Error analysis of direct methods of matrix inversion. *Journal of the Association for Computing Machinery, 8*(3), 281–330

Wilkinson, J. H. (1963). *Rounding errors in algebraic processes*. Englewood Cliffs, NJ: Prentice-Hall

Woytinsky, W. S. (1947). What was wrong in forecasts of postwar depression. *Journal of Political Economy, 55*(2), 142–151

Wu, D. -M. (1973). Alternative tests of independence between stochastic regressors and disturbances. *Econometrica, 41*, 733–750

Yule, G. U. (1926). Why do we sometimes get nonsense-correlations between time series? A study in sampling and the nature of time series. *Journal of the Royal Statistical Society, Series A, 89*, 1–64

Zeileis, A. (2006). Implementing a class of structural change tests: An econometric computing approach. *Computational Statistics and Data Analysis, 50*, 2987–3008

Zeileis, A., & Hornik, K. (2007). Generalized M-fluctuation tests for parameter instability. *Statistica Neerlandica, 61*(4), 488–508
Zellner, A. (1962). An efficient method of estimating seemingly unrelated regressions and tests for aggregation bias. *Journal of the American Statistical Association, 57*, 348–368
Zellner, A. (1997). *Bayesian analysis in econometrics and statistics: The Zellner view and papers. Economists of the twentieth century* (p. xvi, 554). Cheltenham: Brookfield
Zellner, A. (2006). *Email re the Durbin-Watson Statistic*. C. G. Renfro, Recipient
Zellner, A., & Huang, D. S. (1962). Further properties of efficient estimators for seemingly unrelated regression equations. *International Economic Review, 3*(3), 300–313
Zellner, A., & Theil, H. (1962). Three stage least squares: Simultaneous estimation of simultaneous equations. *Econometrica, 30*(1), 54–78
Zellner, A., & Thornber, E. H. (1966). Computational accuracy and estimation of simultaneous equations econometric models. *Econometrica, 34*(3), 727–729
Zellner, A., Stroud, A., & Chau, L. C. (1963a). *Program for computing seemingly unrelated regressions estimates*. Madison, WI: Social Systems Research Institute, Department of Economics
Zellner, A., Stroud, A., & Chau, L. C. (1963b). *Program for computing two and three stage least squares and associated statistics*. Madison, WI: Social Systems Research Institute, Department of Economics
Zivot, E., & Wang, J. (2002). *Modelling financial time series with S-PLUS*. May 22, 2002 (ed.), Seatle, WA: Insightful Corporation