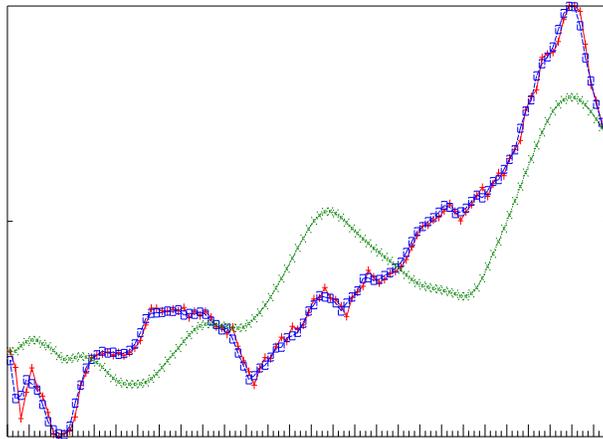


The Craft of Economic Modeling

Part 1



Clopper Almon

FIFTH EDITION
January 2014

The Craft of Economic Modeling

Part 1

Clopper Almon

Department of Economics
University of Maryland
College Park, MD 20742

January 2014

Fifth Edition

January 2014
Copyrighted by the Interindustry Economic Research Fund, Inc.
P.O. Box 451, College Park, Maryland, 20740
Telephone 301-405-4609

CONTENTS

ACKNOWLEDGEMENTS.....	iii
INTRODUCTION.....	v
CHAPTER 1. WHAT IS AN ECONOMIC MODEL AND WHY MAKE ONE?.....	1
1. An Accelerator-Multiplier Interaction Model of an Imaginary Economy.....	2
2. An Introduction to Identity-Centered Modeling	7
Model 1: The Pure Accounting Model.....	9
Model 2. A Behavioral Ratio for Personal Consumption.....	16
Model 3. More Behavioral Ratios and Shortcutting Exogenous Variables.....	20
Model 4. Seeing Past Inflation.....	23
Model 5. Modeling Behavior with Regression.....	28
CHAPTER 2. BASIC CONCEPTS OF LEAST SQUARES REGRESSION.....	37
1. What is the Method of Least Squares and Why Use It?.....	37
2. How to Calculate the Least Squares Regression Coefficients.....	39
3. Some Measures of How Well the Equation Fits.....	42
4. Measures of Autocorrelation and their Uses.....	44
5. Statistics for Each Independent Variable.....	46
6. Matrix Notation for Regression.....	48
7. A Shortcut to the Sum of Squared Residuals.....	52
8. Mexvals and Derivatives -- Measures of the Importance of Each Variable.....	55
9. Leverage, a Measure of the Influence of Individual Observations.....	57
CHAPTER 3. INTRODUCTION TO G 7.3.....	59
1. Tutorial of the Basics.....	59
2. Digression on Graphs in G.....	64
3. Building a Model.....	66
4. Using Help.....	70
CHAPTER 4. FORECASTS AND EXPERIMENTS.....	73
1. Counter-historical Simulation.....	74
2. Forecasting.....	77
3. Debugging Models	81
CHAPTER 5. TECHNIQUES FOR IMPROVING A MODEL.....	85
1. Working with the @cum Function.....	85
2. Rates of Change with Continuous Compounding.....	86
3. Inflation I — The Wolf in Sheep's Clothing.....	87
4. Soft Constraints.....	90
5. Inflation II — A Rough-hewn Timber.....	92
6. Inflation III — Accounting for External Shocks.....	93

7. Inflation IV: Using Distributed Lags.....	94
8. Inflation V: Working in Levels instead of Rates of Change.....	98
9. Ways to Improve AMI.....	100
Improve the Investment Equations.....	100
Improve the Savings Function.....	100
Expand the Government Sector.....	101
Improve the Foreign Sector.....	101
CHAPTER 6. GOOD ADVICE AND BAD EXAMPLES.....	103
1. Good Advice.....	103
2. Bad Examples.....	105
3. The Worst Example.....	114
CHAPTER 7. THE THEORY OF LINEAR SYSTEMS.....	119
1. The General Solution of a System of Linear Difference Equations.....	119
2. Frequency Response.....	125
3. Explaining the Emergent Cycle Phenomenon.....	130
Index.....	135

ACKNOWLEDGEMENTS

This book has been in the writing for twenty years or so, but the activity has been particularly lively in the past ten years. The material has been used by a score or so of instructors and by several thousand students. I am grateful to all of them for their patience with its shortcomings and suggestions for improvements. The software which accompanies the book, *G7*, was written primarily by the author but with many contributions and corrections by students and associates. In particular, I mention with gratitude those who have helped in the development of the programs. This group includes David Robison, Jane-Marie Mulvey, Pamela Megna, Douglas Meade, Kang Chen, Jeffrey Werling, Qiang Ma, Qisheng Yu and Ronald Horst. Teaching associates John Sabelhaus, Peter Taylor, William Stephens, Craig Hiemstra, Lesley Cameron, Lorraine Sullivan Monaco, and Maureen Sevigny have made many useful suggestions. Douglas Meade and Daniel Marcin have helped to port the newest version to Open Office 3.4 and update the text. Many, many others too numerous to mention have made valuable suggestions.

Special thanks go to Wang Yinchu and a team at the Economic Information Agency of Jiangsu province, who produced a Chinese translation of the second edition published by the University of Nanjing Press. Similar thanks go to Paul Salmon of the University of Rennes who translated the second edition into French and produced a French-speaking version of the programs. Likewise, Alberto Ruiz-Moncayo and Luis Sangines have produced a Spanish-speaking version of the programs while the first edition of the text was translated into Spanish under the direction of Javier Lantero at Fundacion Tomillo in Madrid. Maurizio Grassini has been active in making the programs known in Italy. His work led ISTAT to become the first central statistical office to use G data banks for official dissemination of statistics, a purpose for which they are well suited. I am indebted to Wladislaw Welfe of the University of Lodz in Poland and Maurizio Ciaschini of the University of Urbino in Italy for the opportunity to conduct in their institutions week-long intensive seminars on the use of G. Yasuhiko Sasai has produced a Japanese translation with an application to Japanese data. Georgy Serebryakov and his staff produced a hardcover Russian edition in 2012.

Finally, I am grateful to all my colleagues at INFORUM who have both encouraged the work on this project and liberated me to pursue it.

INTRODUCTION

This is a book about how to build models of a business, an industry, or the whole economy. It explains techniques used both in simple, single-equation models for forecasting the sales of a single product of a single company and also in complex, many-equation models of an entire economy or of the world. The principal example in the first two parts of the book is a macroeconomic model of the USA. The title, *The Craft of Economic Modeling*, emphasizes that the book does not stop with the theory or even with a few examples. Rather, it leads the reader directly into practice, for it is accompanied by computer software and data that enable the user to apply to practical problems every technique described. Though some of the exercises are just drills in the techniques, many call for originality and creativity in handling real-life problems. Of course, as in any craft, a book can only help, not replace, a living teacher.

The computer software, the *G* regression package, version 7.3, (referred to as *G7*) and the *Build* model builder, are comprehensive, easy-to-use programs that run under Windows XP and Windows 7 or 8. They are designed for work with time-series data. Public domain versions accompany the book or are available via Internet (www.inforum.umd.edu), where thousands of economic time series are also available as data banks for *G7*. Assembling equations into a model requires the use of the Borland C++ compiler, which is also available for free download from

<http://www.embarcadero.com/products/cbuilder/free-compiler>.

All software and data are also available on a CD-ROM from the author at nominal cost.

Simply put, an economic model is a set of equations which describe how the economy or some part of it functions. In my view, a model should incorporate and test our understanding of how the economy works. Its equations should make sense. And it should be possible to test how adequate our understanding is by running it over the past and seeing how well it can reproduce history. By changing some of its assumptions and rerunning history with the changed assumptions, it is possible to analyze the effects of policies. Finally, it should be useful not only for policy analysis but also for forecasting. By studying the errors of the forecast, the builder of the model may hope to improve his or her understanding of the economy.

I must warn the reader that these simple views are by no means shared by all -- or even by most -- practitioners of model building. The vector-autoregression (VAR) school gives little or no weight to the equations expressing any sort of understanding of the economy; ability to reproduce the past, however, is of great importance. The computable general equilibrium (CGE) school gives great weight to the equations making sense but has little interest in testing the dynamic properties (if any) of its models or in the equations fitting more than one point. In my view, each of these schools is right in what it values but remiss in what it neglects.

Some of the equations in our models have constants which describe the behavior of firms, consumers, or other economic agents. These constants, often called "parameters", must somehow be estimated. The most frequently used way is by choosing them so that the equation describes accurately the behavior of those agents in the past. Thus, estimating the parameters is just a way to sum up the lessons of the past to forecast the future or examine how the past itself might have been different had different decisions been made. A large part of this book is about how to do that

summing up. But it is just as important to choose judiciously the variables to be included in the model. Unfortunately, there is little theory about how this choice should be made. I hope that the discussion in this book will at least help to make the choice a conscious one.

Model building is a serious business because models are playthings. If that sounds paradoxical, consider that we can, fundamentally, learn in three ways: from our own real-life experiences, by hearing or reading about the experiences of others, and by play. Indeed, zoologists have observed that one of the distinguishing characteristics of human beings is the adult's retention of the ability to play. Therefore, even adults can learn. In economic policy, no one has enough personal experience to be of much value; and evaluating the experiences of others is difficult because many influences were at work in producing the actual historical outcome. That leaves play. But in the past it has been impossible to "play economy" without playing with the real economy. Models and micro computers, the super playthings of adults, open up the possibility of learning about how the economy works through play. Personally, I readily admit that nearly everything I think I know about how the economy works I have learned through play with models. When I read books or papers about the economy written by people who have not played in this way, I am often struck by how one-sided their knowledge is. Yes, I think that play with models is the best way to raise the level of public understanding of how the economy works, and that the quality of the economic policy we actually adopt depends on that public understanding. The attempts to develop that understanding by exposing college students to clearly written texts and objective multiple-choice examinations have failed. Maybe play can succeed.

When computers first made economic modeling feasible, the mystique of the machine raised expectations that models built with it would prove as infallible as the machine's arithmetic. In fact, the record of forecasters using models looks very human. That is, of course, exactly what one should expect. But expectations are seldom rational, and when it became clear that models were not the panacea that would cure all the ills of a business, some disillusion set in. Many bad models have been built, some of them by large commercial firms in the modeling business. Sometimes the salesman got the better of the scientist in the claims their builders made for them. However, I believe that those who know how sensitive a model can be to the judgmental decisions made in building it have not been disappointed in models, for they have not expected what could not be delivered. Working through this book will offer the first-hand experience with models that enables one to judge what can and what cannot be expected of them.

I should perhaps advise you that a large fraction of the economics profession regards such modeling as we will do here as *passé*. This widespread opinion is based on a misreading – or non-reading – of a well-known article by Robert Lucas which *seemed* to say that models such as those we will be building “can, in principle, provide *no* useful information as to the actual consequences of alternative economic policies.”¹ In fact, Lucas had in mind a particular group of models in use at that time (1975), and he actually went on to show how they could be modified to give what he considered correct answers to such policy questions. His real point was not that models are useless for policy analysis but simply that the fact that a particular model has a good “track record” in

1 Robert E. Lucas, “Econometric Policy Evaluation: A Critique” in *The Phillips Curve and Labor Markets*, Karl Brunner and Allan H. Meltzer, editors, Amsterdam, North Holland, 1975.

forecasting does not mean that it will give correct answers to questions involving policy changes that invalidate some of its equations. That point is surely accepted by all model builders and users, both then and now. It by no means invalidates the use of models; in fact, quite to the contrary, it shows that models are necessary so that the required changes in policy can be explicitly made.

Any policy analysis requires a model, for it requires that we compare how the world would be with and without the change of policy. The only question is whether the model by which we make that comparison is explicit, subject to scrutiny and testing, or implicit, vague, held somewhere in our head and subject to change as our mood changes. Historical facts never replace a model for policy analysis, for history gives us results only with or without the policy, never both with nothing else changed.

I can hardly imagine how any systematic progress can be made in economic understanding, forecasting or policy analysis without use of quantitative models. I hope that this book can contribute to lifting the level of applied modeling and to a recognition of the judgment and skill needed to build and use them well.

Over forty years of experience as an applied model builder have affected my choice of material for this book and its presentation. As it is usually taught, econometrics is a series of exercises in applied probability theory. I made a significant investment in learning this material in the expectation that it would prove useful in building meaningful models. That expectation has been largely but not quite totally disappointed. Nor am I alone in this experience; most applied model builders with whom I have dared to discuss the matter have heartily agreed. Such a statement is a plain declaration that "the emperor has no clothes" and is not likely to persuade teachers of econometrics to adopt this book as a text, unless, of course, they want the students to know the truth.

This book emphasizes the simple, direct techniques and the common-sense analysis that I have come to trust. These and only these are presented in Part 1 of the text. As a teacher, I have found that it is fairly trivial to get a student to learn and apply some "sophisticated" technique; it is a major accomplishment to get him or her to *think* about whether the equation and the estimated parameters make sense. I have tried to coach the reader in such thinking.

The principal value of the probabilistic theory has been to show how, in certain situations, the methods of Part 1 may lead to deceptive results. Understanding those possibilities is important, so in Part 2 we look at the probabilistic theory and present some of the widely-used techniques, such as generalized least squares and two-stage least squares, which are based on the probabilistic assumptions. Through applications of these techniques to actual problems with actual data, the student can form his own opinion of their value.

Part 3 turns to multisectoral models, models that distinguish different industries, such as chemicals or textiles. All of the techniques developed in Parts I and II remain useful, but must be supplemented by ways of handling the relations among industries.

A Guide to the Book

Before plunging into a theoretical discussion of any subject, one should begin with acute and sensitive observation of the phenomena it concerns. In the case of macroeconomic modeling, that means looking at the time series of economic data such as the National Income and Product Accounts (NIPA), employment, interest rates, money supply, exchange rates, and prices. The "nipashow" file, which comes with the software for this book, provides a good opportunity to practice such observations. Once the *G7* program is running, it is only necessary to say "add nipashow" to begin this display of economic data. At first, nipashow asks questions about the graphs it shows, then it simply shows other graphs and asks what the viewer sees interesting in them. Finally, it leaves both the drawing and the asking of questions up to the user. After the experience of viewing these graphs, one may well ask, How can the movement of these variables be explained? And that is where the book proper begins.

Chapter 1 illustrates the idea of a model and shows how a simple model of the economy can generate business cycles. It first uses a hypothetical model for which the reader can easily compute the solution. It then introduces a series of five models of increasing usefulness. This series illustrates identity-centered modeling, modeling that puts definitions and simple "behavioral ratios" in center stage. Model 1 incorporates only the definitional equations of the NIPA and affords a good review of the concepts in these accounts. Models 2 through 4 demonstrate the use of behavioral ratios to make the models richer and more useful. Model 5 introduces a few equations with parameters estimated with regression analysis from data on the American economy. Model 5 illustrates a fundamental point: it is not enough to have explanations of all of the parts of an economy to have an explanation of the whole. The dynamic interaction of individually sensible parts can generate quite surprising -- indeed, totally nonsensical -- results. Building a good model is far more than putting together equations which individually satisfactorily explain each component of the economy. The first section of chapter 1 does not require the use of computers. It can be discussed in class during the same week in which the "nipashow" plots are being studied.

Chapter 2 explains least-squares computations and the display of results shown by *G7*. Chapter 3 concerns how to use *G7* and particularly, how to estimate the parameters of model equations by regression. Chapter 4 shows how to combine the equations into a model and has you do some experiments and a forecast. It also explains how to use the numerous diagnostic techniques made available by the software for spotting errors in your model. The experiments reveal some weaknesses of the model, and Chapter 5 discusses ways to eliminate them. It closes with a number of suggestions on how you can use the techniques now at your disposal to make significant improvements to the model of Chapter 1. Chapter 6 gives some simple-minded maxims for getting reliable, trustworthy regression equations. Most of these maxims are too simple to be mentioned in a self-respecting textbook of econometrics. But they are the result of commenting on the work of hundreds of students over the years. I found myself saying the same thing over and over until I finally put pride aside and wrote down what I had been saying. Probably you will say to yourself as you read it, "That is obvious; I don't need to be told that." Fine, but it is terribly easy to let these errors slip in. Some very distinguished members of the economics profession have done so. I urge you to measure every equation of your model against these simple criteria.

Chapter 7 gives explains the basic mathematical theory of linear models. Its role is to help understand some of the properties that the student may find appearing in his models, particularly their tendency to generate cycles.

Part 2 begins with a fairly comprehensive quarterly aggregate model of the United States. In building it, I have, I believe, learned a lot about how the economy works. I have even come to the slightly arrogant opinion that no one should trust his understanding of how the economy works who has not put that understanding to the test of building a model. In this model, I have endeavored to get at structure and avoid reliance on lagged values of dependent variables. Thus, the model is called the Quarterly Econometric Structural model, or *QUEST*. Actually, *QUEST* connotes more than it abbreviates. Because all of the files for estimating and building *QUEST* are given, you may use it as the starting point for your own quest to build a model of the economy. Many people have felt that they learned to understand the human face or figure by modeling it. I have found the same to be true of modeling the economy. I hope that in working with *QUEST* you will also feel that you have joined in the search for a better understanding of the economy. *QUEST* also provides good examples of equations to work with in the rest of Part 2.

Part 2 continues with an explanation of optimization in models in Chapter 9. This optimization may be used either to improve the fit of the model in the historical period or to design optimal economic policy. The important technique of nonlinear regression is explained and illustrated in Chapter 10. Chapter 11 outlines some of the relations between probability theory and least-squares regression. This relationship of rather modest importance to the applied model builder has been magnified by the academic establishment to formidable proportions. The Datamaker fable, by which this subject is here introduced, is one of the best-loved sections of the book by many readers to whom this material has been presented with a straight face as if it ought to be believed. Basically, this fable says that if we are in an ideal case, the least squares we have used in Part I is, in fact, the best way to estimate equations. The rest of the chapter offers some methods, based on probability ideas, that can be tried when the ideal conditions are clearly not met. These techniques include the Hildreth-Lu procedure for dealing with auto-correlated residuals and "seemingly unrelated" regression and stacked regression with constraints across equations. Methods of estimating equations with moving-average errors are explained. The old subject of simultaneous equation bias and the new subjects vector autoregression, cointegration, and unit roots get just enough attention to show how they relate to the modeling approach used here.

Chapter 12 explains how to run a model many times with artificially introduced random errors. It helps to answer the question of how much confidence we can have in a particular model.

Part 3 extends our scope to multisectoral models. It first explains basic input-output analysis in the ideal case and then some of the practical problems arising in actual application. Special techniques for modeling consumer behavior, investment, and imports are then considered. The *Interdyme* software for building multisectoral models is then explained and illustrated.

CHAPTER 1. WHAT IS AN ECONOMIC MODEL AND WHY MAKE ONE?

The tax cuts and high defense spending of the Bush administration, combined with relatively slow economic growth led to growing federal deficits from 2001 to 2007. Over the same period, US exports shrank and imports ballooned, spelling hard times for many manufacturing industries. Was there a connection between these two developments? Specifically, did the tax cuts cause manufacturing's problems? Or were they caused by a loss of competitive ability or drive?

The argument that the federal deficit was responsible for the difficulties of industry is simply that the deficit led to high real interest rates, which led to a rise in the dollar, which made American goods expensive abroad and foreign goods cheap here, and consequently stifled exports and promoted imports.

Such reasoning points to theoretical possibilities. Whether or not these possibilities have been historically important is another question, one that theory cannot decide. To answer it, one must *quantify*, in the first case,

- the effect of the federal deficit on interest rates
- the effect of interest rates on exchange rates
- the effect of exchange rates on exports and imports
- the effect of interest rates on investment, personal income, and saving.

Moreover, though the tax cuts may have had a negative effect on industry through these effects, they increased after-tax income of households, who undoubtedly spent it in part on products of industry. When all of these effects are quantified and combined into a system of equations, one has a model.

That model -- that is, those equations -- can then be solved under alternative assumptions about the tax rates. First, one may use the actual tax rates and solve the model. The result is called a historical simulation. It will not give exactly the historical values of the variables in the model, for the equations which describe the behavior of people will not be exact. However, if the historical simulation is not a pretty good approximation of reality, we had better take our model back to the drawing board before going further. Assuming that it passes the historical simulation test, we then change assumptions about the tax rates -- maybe this time we leave them unchanged from the 1990s levels -- and again solve the model. This time we get what might be called a counter-historical simulation. Comparison of the historical simulation and the counter-historical simulation will answer the question; "Did the tax cuts stifle manufacturing?"

In economics, it is, in general, impossible to answer such questions of cause and effect without some kind of model. The natural sciences allow for controlled experiment, a luxury we economists can seldom afford. Building and running models under various assumptions is our version of the controlled experiment. It must be emphasized that history alone can never answer a question of cause and effect, for history gives us only one "run" of the economy. To speak of cause and effect in economics, one must have in mind some kind of model. It may be an entirely subconscious model, or a model spelled out in theory but never quantified or tested, or an explicit, quantified model that has been subjected to careful testing and scrutiny. I need hardly add in which sort I have the greater

confidence.

In Part II, we will build a fairly comprehensive aggregate model of the US economy and will apply it to such questions as we have been discussing. But the model will also have much wider uses. In particular, it will be useful for forecasting and for analyzing the effects of policies *in advance*. In 2001, for example, we could have studied whether the controversial Bush tax cuts were likely to stimulate the economy or choke off recovery just as it was getting started. We could ask: "Given expected future real federal spending, what level of taxation is necessary to balance the federal budget on average over the next five years?" We can examine the effects of tight money or easy money; we can see what happens if productivity is retarded or accelerated; we can look at the effects of stronger or weaker exports. We can ask what effects the stock market has on the economy, and vice versa. In short, we can examine most of the major questions of macroeconomics.

If you, like me, find these questions both fascinating and important, then come along on the path that leads to a way to answer them.

1. An Accelerator-Multiplier Interaction Model of an Imaginary Economy

To illustrate what is meant by a model, let us begin with one so simple that we can easily calculate it by hand. To keep the numbers simple, we will just make up the constants in the equations, textbook style, out of thin air with a cursory glance at the American economy in the early 2000's. To describe even this simple model, it will be necessary to use two rather long words of Greek origin. Namely, we must distinguish between the *exogenous* variables and the *endogenous* variables. The exogenous variables are used by the model but are not determined by it; the endogenous variables are both used in the model and are determined by it. (The Greek root *gen* means "birth" or "born". The exogenous variables are "born" outside the model; the endogenous, within it.) Population and government spending are exogenous variables in many models; income and investment are typical endogenous variables in macroeconomic models. A system of equations which relates this quarter's values of the endogenous variables to the current values of the exogenous variables and previous values of the endogenous variables is, in economic terms, a model as we shall use that word. In more precise mathematical terms, it is a system of difference equations. By calculating the simple model we present below with a pocket calculator for a few quarters, you should get a good idea of how a system of difference equations can generate a time path of the endogenous variables that looks at least reminiscent of the actual course of the economy. You will also discover the remarkable fact that the endogenous variables may show a pronounced cyclical behavior even though there is no trace of a cycle in the exogenous variables.

Consumption in this simple economy can be described by the equation

$$(1) \quad C = .6*Y[1] + .35*Y[2]$$

where C is consumption, Y is disposable income, and Y[1] is disposable income one period earlier, read as "Y lagged once."

Fixed investment follows the “accelerator” theory of investment which makes investment for expansion depend on past growth in peak output. Thus, to continue our simple economy,

$$(2) \quad I = R + 1.0 \cdot \Delta PQ[1] + 1.0 \cdot \Delta PQ[2] + .1 \Delta Q[1],$$

where I is gross investment, Δ indicates a first difference ($\Delta Q = Q - Q[1]$), R is replacement investment, and PQ is the current or previous peak output, thus,

$$(3) \quad \begin{aligned} PQ &= Q && \text{if } Q > PQ[1] \text{ or} \\ PQ &= PQ[1] && \text{otherwise.} \end{aligned}$$

The first term of equation 2 represents, as already said, replacement investment. The next two terms represent expansion of fixed capital when output rises above its previous peak, the output level for which we already have adequate capital. The final term in equation 2 represents inventory investment. Expansion investment -- from the way peak output is defined -- cannot be negative; inventory investment, on the other hand, can be either positive or negative.

Imports, M, are given by

$$(4) \quad M = -380 + .2(C + I + X)$$

where X is exports, which are exogenous.

The output variable, Q, is

$$(5) \quad Q = C + I + G + X - M$$

where G is government spending. Finally, disposable income is roughly Q less taxes, which we take as 28 percent of Q, so

$$(6) \quad Y = .72 \cdot Q.$$

The exogenous variables in this simple model are:

- G government spending
- X exports
- R replacement investment.

All others are endogenous. If in some period the exogenous variables and the required lagged values of the variables Y, PQ, and Q are known, then equations (1) - (6) can be applied, one after the other in the order given, to get the values of all the endogenous variables for that period. Then one can go on to the next period, and the next, as long as the exogenous variables are known. Thus, given starting values and exogenous variables, the equations can be used to calculate the course of the economy over time. This simple fact is so fundamental that it is a good idea to impress it upon yourself by seeing it happen in your hands, rather than inside a somewhat mysterious computer.

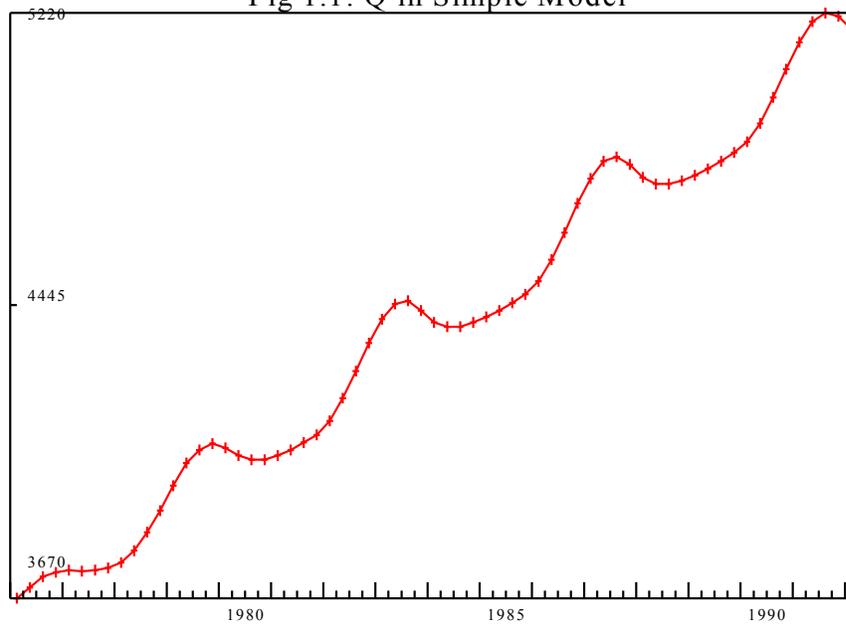
Table 1.1 therefore provides a work-sheet for calculation of this simple model. At this point, you should get out your pocket calculator and compute at least a few lines. Since you need three lagged values of PQ to get started, the first line that you can calculate is line 4. All values, however, are shown for lines 4, 5 and 6, so you can check that you have understood the formulas. (Do not worry if your calculated values are different after the decimal from those shown in the table. The table was calculated by a computer carrying more decimal places than shown here, and the numbers were then truncated in printing.) By line 6, this economy is in recession, as you can see from the Q column. In line 7, some values are missing and need to be computed by you, but the total output, Q, is left as a check. By line 8, you are entirely on your own except for the exogenous variables. Their values, by the way, are straight-line approximations of their values from 1979 to 1992 in the American economy. Places for you to put calculated values are marked by a decimal point.

I urge you as strongly as possible to calculate a few periods of Table 1.1. until the next peak is reached and the economy turns down. Endeavor to explain why the economy turned up and why it turned down. If you understand how this simple model works, you will have no great difficulty with the more complicated models which follow. If you miss the point here, trouble lies ahead.

The economy is in recession when you begin. As you calculate, you will see it grow modestly until it passes its previous peak. Then the accelerator terms in the investment equation will drive the economy up steeply. Then growth slows; but as soon as it slows down, investment drops, and slow growth is turned into decline. This model will thus show a significant cycle. Figure 1.1 shows the course of Q which it generates. Notice, however, that there is no cycle in any of the exogenous variables. They all grow along perfectly straight lines, as you can see by noting that they rise by the same amount each period.

We have thus discovered the important fact mentioned above: the equations of economic behavior can generate cycles where there are none in the exogenous variables. From Figure 1.1, you will have noticed that the cycle is persistent and of about a constant amplitude. This is a characteristic of a non-linear model. Equations (1) - (2) and (4) - (6) are linear, that is, the dependent variable on the left is just a sum of terms each of which is just one of the other variables possibly multiplied by a constant. Equation (3) for peak output, PQ, is, however, a horse of another color and not at all linear. If all equations of a model are linear, the model itself is said to be linear, but so much as one non-linear equation and the model becomes nonlinear. This distinction is important because it may be shown mathematically that in a linear model any cycle will, in all probability, either fade away or grow explosively. Only a most particular constellation of values of the constants in the equations can give the sort of steady, persistent cycling behavior shown in Figure 1.1. Thus, in the construction of models, we should pay particular attention to the non-linearities.

Fig 1.1: Q in Simple Model



You should not conclude from this example that *all* such examples will necessarily show this sort of cyclical behavior. In models with strong accelerator investment equations, however, cycles are not unlikely.

Exercise

- 1.1.(a) Calculate the course of the imaginary economy until Q turns down again. (b) Explain why this economy turns down when it has been growing, and why it begins to grow after a few periods of decline. Remember that everything there is to know about this economy is contained in the equations. Do not bring into your explanation factors like interest rates or inflation which play no role in this economy.

Table 1.1: Worksheet for Hand-calculated Examples

Line	X	G	C	R	I	M	Q	PQ	Y
1	303.9	714.9	2653.6	518.7	631.9	337.9	3966.6	3966.6	2855.9
2	308.8	718.5	2696.7	522.5	653.6	351.8	4026.0	4026.0	2898.7
3	313.8	722.2	2738.8	526.2	648.2	361.8	4061.2	4061.2	2924.0
4	318.7	725.9	2769.0	530.0	628.1	363.1	4078.6	4078.6	2936.6
5	323.6	729.5	2785.3	533.7	588.0	359.4	4067.2	4078.6	2928.4
6	328.5	733.2	2784.8	537.5	553.7	353.4	4047.0	4078.6	2913.8
7	333.5	736.9	.	541.2	.	.	4033.7	4078.6	2904.2
8	338.4	740.5	.	545.0
9	343.3	744.2	.	548.7
10	348.3	747.8	.	552.5
11	353.2	751.5	.	556.2
12	358.1	755.2	.	560.0
13	363.0	758.8	.	563.7
14	368.0	762.5	.	567.5
15	372.9	766.1	.	571.2
16	377.8	769.8	.	575.0
17	382.8	773.5	.	578.7
18	387.7	777.1	.	582.5
19	392.6	780.8	.	586.2
20	397.6	784.5	.	590.0
21	402.5	788.1	.	593.7
22	407.4	791.8	.	597.5
23	412.3	795.4	.	601.2
24	417.3	799.1	.	605.0

2. An Introduction to Identity-Centered Modeling

As we move from a simple imaginary economy to a real economy, we immediately encounter two complications:

1. The accounting system becomes much more complicated with many more variables and identities relating them.
2. The behavioral relations are no longer simply made up but must be induced from an interaction of our imagination with historical data provided by the national accounts and other sources.

In the rest of this chapter, we will go through a sequence of five models of increasing usefulness, each built from the previous one.

The first four models are very similar to those that have found wide use in business in the form of spreadsheet calculations such as those performed by Excel, Lotus and similar programs. Today, nearly anyone planning a conference, a school budget, or a complex business deal wants to see a spreadsheet to judge how the finances will come out. These spreadsheets are what may well be called *tautological* models. *Given their assumptions*, they are sure to be correct. They express their assumptions usually as simple ratios and employ simple addition or subtraction to total up the revenues and the costs and arrive at the bottom line with utmost clarity. Our first four models are patterned on these spreadsheet models, and their great virtue is their simplicity and ease of understanding. They stress, however, the part of model which is most certainly correct, the accounting identities. For this reason, I speak of this approach as *identity-centered* modeling.

The fifth model goes beyond what it is easy to do in the spreadsheets, namely, it introduces behavioral equations estimated by linear regression, that is, by fitting a mathematical equation to historical data. Relations found in this way can express much more complicated economic behavior than can the simple ratios used in previous models. Through them we can hope to find, for example, the way that investment is related to numerous factors such as growth in output, wear out of capacity, and costs of labor and capital. Models that use these equations can give a much richer description of economic behavior than can the tautological models. But the behavioral equations will not fit perfectly, so -- unlike the tautological models -- these models do not perfectly reproduce the past. How well they can reproduce it, therefore, can become a criterion in judging them.

Model 1 just sets out the variables and identities that are central to the system of national accounts used in the United States. We may call it a pure-accounting model because it makes connections among variables based only on accounting definitions, not on economic behavior. It is just a skeleton and would give strange results if we tried to make it walk. Moreover, it has many, many exogenous variables so that it would be awkward to use in practice.

Model 2 introduces a single, obvious but centrally important *behavioral* relation into the model, namely, a relation between Personal disposable income and Personal consumption expenditure. The relation which we introduce is extremely simple, namely a ratio, fixed in any one period but variable

over time, between consumption and income. Because such a ratio expresses economic behavior, it is natural to call it a *behavioral ratio*. This ratio then becomes an exogenous variable while the quantity in its numerator becomes an endogenous variable. The introduction of a behavioral ratio does not reduce the number of exogenous variables. Rather it *replaces* one exogenous variable by another which is likely to be more constant, perhaps without a trend, and easier to forecast. For example, if you ask me what Personal consumption expenditure will be in 2050, off hand I would have no idea. But if you ask me what the ratio of Personal consumption expenditure to Personal disposable income will be, I could answer with some assurance, “Oh, about what it is now, somewhere between .9 and 1.0.” Model 2 will also allow us to make a meaningful multiplier calculation, that is, to answer a question like, If government spending goes up by 10 billion, how much will GDP go up? The Model 1 answer would be 10 billion; Model 2 has something more interesting to say.

Model 3 carries further the process of *exogenous variable replacement* by means of behavioral ratios. It also shows how certain exogenous variables can be lumped together to reduce the total number.

Model 4 introduces another useful method of exogenous variable replacement, namely to make the variable exogenous in constant prices. Models 1, 2, and 3 worked only in current prices; model 4 introduces *deflators* so that we can also talk about both the rate of inflation and variables measured in real, constant-price terms. Forecasting of exogenous variables is often easier in real rather than in nominal (current-price) terms. For example, Social Security benefits can be calculated in real terms from the age structure of the population and the payment schedule. The dollar value of the payments, however, will depend on the increases in the Consumer Price Index (CPI) between now and the time they are actually paid. That increase is something the model should eventually forecast endogenously. Thus it makes good sense to break down the forecasting of these payments into two steps, first the constant-price or real value and then the current-price or nominal value.

Every equation in Models 1 through 4 is an identity, so our approach has earned its label as *identity-centered* modeling. Essentially, we have tried to use identities to replace variables that are hard to think about by others that are easier to grasp intuitively. Good use of identities is essential for good modeling. This recognition of the central role of identities in modeling is in stark contrast to the way that they are usually dismissed in econometric texts with the comment that an identity can be used to eliminate a variable. Of course it can, and then one is left with a variable that is hard to think about without the handle — the identity — that gives us an easy way to think about it.

Model 5, as already noted, introduces an extremely useful method of reducing the number of exogenous variables by finding behavioral relations through regression analysis. Application of this technique will occupy us through much of this book.

Model 1: The Pure Accounting Model

Table 1.2 shows the outline of the U.S. system of National Income and Product Accounts (NIPA) as a sequence of equations. The first column (op) shows the operation required to form each identity. The second column (code) shows the code name for the variable in the Quip data bank. The third column gives the verbal description of the data series, as shown in the NIPA. Finally, the value for the variable for a recent quarter is shown, in billions of dollars.

The first equation defines Gross domestic product as

$$\text{gdp} = \text{c} + \text{v} + \text{x} - \text{m} + \text{g}$$

where the variables have the meanings shown in the table. The variable names shown in the *code* column are the ones used in the Quarterly Income and Product (Quip) data bank which goes with the G software which we use in this book. Each equation after the first starts from the bottom line of the preceding one and adds or subtracts (as shown in the *operation* column) the indicated items down to the = sign, where the result is recorded and named. Thus, the second equation is

$$\text{gnp} = \text{gdp} + \text{irrow} - \text{iprow}$$

We will refer to the first equation as the the "product side" of the accounts. The remaining equations form the "income side" and may be less familiar. They show the transition from Gross domestic product to Personal disposable income and Personal saving. (I follow the NIPA practice of capitalizing only the first letter of the first word of each line in the accounts.)

Since these accounts will form the framework for much of our modeling, it is important that we understand what they represent. The product-side equation for GDP defines it as total *final* product of the economy. The word *final* here means that products used in making other products are not counted. Thus, one farmer may grow corn and sell it to another, who feeds it to pigs, which he sells to a packer, who sells the pork to a grocer, who sells it to the final consumer. Only the value of the pork sold by the grocer goes into GDP. The word "Gross", however, means that there has been no subtraction for the wear out of capital.

We now need to follow carefully through the transition from GDP to Personal disposable income. The first step, as already noted, is to convert from Gross domestic product to Gross national product. The basic idea is that dividends or interest paid by a U.S. company to a foreign resident are part of the *domestic* product of the U.S. but of the *national* product of the other country. More completely, Income receipts from the rest of the world consist of receipts by U.S. residents of foreign interest and dividends, of reinvested earnings of foreign affiliates of U.S. corporations, and of compensation paid to U.S. residents by foreigners. Income payments to the rest of the world consist of payments to foreign residents of U.S. interest and dividends, of reinvested earnings of U.S. affiliates of foreign corporations, and of compensation paid to foreigners by U.S. residents.

Thus, to get GNP from GDP, we add U.S. rest of world income receipts and subtract U.S. income payments.

Table 1.2: U.S. National Accounts in Outline

op	code	Description	2013.3
+	c	Personal consumption expenditures	11522.8
+	v	Gross private domestic investment	2732.6
+	x	Exports of goods and services	2265.9
-	m	Imports of goods and services	2767.8
+	g	Government consumption expenditures and gross investment	3137.4
=	gdp	Gross domestic product	16890.8
+	irrow	Income receipts from the rest of the world	822.7
-	iprow	Income payments from the rest of the world	558.6
=	gnp	Gross national product	17154.9
-	ncca	Consumption of fixed capital	2655.7
=	nnp	Net national product	14499.2
-	nbctpn	Business current transfer payments (net)	120.1
+	pibtp	Business current transfer payments to persons	44.9
-	niprf	Corporate profits with capital consumption and inventory valuation adjustments	2125.7
+	pidiv	Personal dividend income	795.9
-	netintm	Net interest and miscellaneous payments on assets	462.7
+	piint	Personal interest income	1234.4
-	nsd	Statistical discrepancy	-97.5
=	pibg	Personal income before government action	13963.4
+	pigsb	Personal current transfer receipts from government	2421.1
-	nisurp	Current surplus of government enterprises	-41.5
+	nisub	Subsidies (net)	59.1
-	nitpi	Taxes on production and imports	1142.7
-	nconsi	Contributions for government social insurance	1109.1
=	pi	Personal income	14233.3
-	piptax	Personal current taxes	1664.6
=	pidis	Disposable personal income	12568.7
-	c	Personal consumption expenditure	11522.8
-	piipcb	Personal interest payments	253.6
-	piptt	Personal current transfer payments	163.3
=	pisav	Personal savings	629.1

The next step is to subtract from GNP an estimate of capital consumption to get Net national product. This estimate of Consumption of fixed capital is not depreciation as reported by firms in their accounts and tax returns, for that depreciation is in historical prices, prices of the years in which the assets were purchased. If prices have been increasing, the resulting number understates the current cost of replacing what is wearing out. The Bureau of Economic Analysis (BEA) which makes the NIPA therefore constructs a series of capital consumption in comparable prices by putting investment in various physical assets into constant prices, say those of 2009, then applying depreciation at rates appropriate to the different kinds of assets, then summing over assets invested in various years, then summing over the various assets, and finally converting back to current year prices.

In case you got lost in those words and really want to know what I was trying to say, here is the same thing in a formula:

$$C(t) = \sum_{i=1}^n p_i(t) \sum_{\tau=-\infty}^t w_i(t-\tau) v_i(\tau) / p_i(\tau)$$

where $C(t)$ is Consumption of fixed capital in year t , n is the number of different types of physical assets (buildings, trucks, machine tools, and so on), $p_i(t)$ is the price index of product i in year t , $v_i(\tau)$ is the investment in assets of type i in year τ , and $w_i(k)$ is the fraction of the initial value of an asset of type i that depreciates in year k after it was bought. The difference between $C(t)$ and the depreciation reported for accounting purposes is called the Capital consumption adjustment (CCAdj). If this difference is positive, as it usually is, then profits have been overstated in conventional accounting, but they are corrected in the NIPA. The point of all this calculation is to provide figures for Net national product and Corporate profits that are not improperly influenced by past inflation or by modifying depreciation rules in the tax code.

Now we come to a big step, the conversion of Net national product to Personal income. We will take it in two smaller steps. In the first, everything relates to the private economy, while in the second everything is a government action. In the private-sector step, there are three smaller stages. The first is:

- nbctpn Business current transfer payments (net)
- + pibtb Business current transfer payments to persons

Business current transfer payments (net) include legal settlements and theft losses. We first subtract the entire amount and then add back in the portion which went to persons. The stolen case of Bourbon is counted as income to the thief and, by the way, appears in Personal consumption expenditure and thus in GDP.

The second stage is

- niprf Corporate profits with capital consumption adjustment and inventory valuation adjustment
- + pidiv Dividend payments to persons

Corporate profits, as such, do not go to Personal income; Dividends, however, do. Notice that the profits here have been adjusted by the same capital consumption adjustment that was used in the Capital consumption allowance. The net effect of this adjustment on Personal income is therefore exactly zero. The inventory valuation adjustment is an adjustment for the change in the values of inventory due to price changes. If a store is using “first in first out” inventory accounting and prices have been increasing, then the store has been charging itself for inventory at less than the replacement costs of the goods sold and thus exaggerating profits. This adjustment removes this effect. Since the adjustment is also made to the amount of inventory change included in investment on the product side of the accounts, the net effect of this adjustment on personal income is also exactly zero.

Finally comes the puzzling pair

- netintm Net interest and miscellaneous payments on assets
- + piint Personal interest income.

Net interest is all interest paid by business less interest received by business. It is also all the interest that is included in GDP. Interest paid by consumers on credit cards, automobile loans, or installment credit is not counted as part of Personal consumption expenditure and therefore is not part of GDP. (Do you think it should be?) Similarly, interest on the national debt is not part of Government purchases of goods and services. (Should it be?) Interest paid by business -- and thus included in the price of goods -- is, however, included in GDP. In particular, interest on home mortgages is included because home owners have been converted into a fictional business in the NIPA; they rent their homes to themselves and the imputed space rental value of these homes, including the interest costs, is part of Personal consumption expenditure. This pair of lines, therefore, removes all interest that is in GDP and adds back all the interest received by persons. The net effect is positive because of the large interest payments of the government to persons.

If we may, for the moment, skip the Statistical discrepancy, which should be zero conceptually, we have reached Personal income before government action. This item is not in the official accounts, but I find it a convenient point of reference. All of the transition from GDP to Personal disposable income down to this point has been the result of events in the private economy; the rest of the way is dominated by the government.

The first of the government actions, a huge one, is to add in Personal current transfer receipts from government. These include Social Security and other government retirement programs, Medicare and Medicaid payments, unemployment insurance, and all welfare programs. A small item, Current surplus of government enterprises, is subtracted. These are part of net national product, but do not go to persons. Then the Subsidies, such as those to farmers, are added in. Then the indirect taxes (Taxes on production and imports) are removed. These are taxes such as property taxes on business property or business franchise taxes or licenses. These charges are included in the prices of the products which compose GDP, but they go to governments, not persons, so they must be removed before we can get Personal income.

This would seem to me to be the logical place to draw a line and call the result Personal income.

The official definitions, however, remove one more item, Contributions for government social insurance. These include Social Security contributions from both the employer and employee and other similar programs. To me, they seem more akin to personal taxes than to the other previously deducted items. In any event, once they are deducted we reach the official Personal income.

From it, deduction of Personal current taxes (mainly income and property taxes but also estate taxes and nontaxes such as fines or admission to parks) leaves Disposable personal income. (I find it peculiar that the nontaxes are not part of Personal consumption.) To reach Personal savings, we must deduct not only Personal consumption expenditures but also Personal current transfer payments, which include transfers to foreigners (mainly American residents sending money to family in the Old Country) and transfers to government. We also must remove personal interest payments, which are mostly interest paid by consumers to business. Here at last we have the interest on credit cards and consumer loans which, as mentioned above, are not counted into Personal consumption expenditure. But of course they are real outlays, so they must be removed before reaching Personal saving by the last of the identities.

We have still not explained, however, the strange little item called Statistical discrepancy. It arises because there are two different ways of measuring GDP. One is by measuring what is produced; the other measures the income generated by producing it. The first, the product measure, is given by $gdp = c + v + x - m + g$. The second, the income definition, begins with National income defined conceptually as earned income and statistically as the sum

+	nice	Compensation of employees
+	niprop	Proprietors income with CCadj and IVA
+	niren	Rental income with CCadj
+	niprf	Corporate profits with CCadj and IVA
+	netintm	Net interest and miscellaneous payments on assets
+	nitpi	Taxes on production and imports
+	nisurp	Current surplus of government enterprises
-	nisub	Subsidies
+	nbctpn	Business current transfer payments (net)
=	ni	National income.

To get from National income to Gross domestic product, we just need to add or subtract:

+	ncca	Consumption of fixed capital
+	nsd	Statistical discrepancy
=	gnp	Gross national product
+	iprow	Income payments to rest of world
-	irrow	Income receipts to rest of world
=	gdp	Gross domestic product.

Now notice that this second, income-based route to GDP does not require estimating Personal consumption expenditure, nor Gross private domestic investment, nor anything else in the product-side definition of GDP. Conceptually, the two approaches should give exactly the same answer.

Will they in practice? You can bet they won't. Many of the items in the accounts must be deduced indirectly from other data. Consider just the problem of figuring out personal consumption of gasoline. Suppose that you have reliable data on retail sales of gasoline, but how much goes to vehicles being driven for business, and therefore not part of Personal consumption, and therefore not part of GDP? There is certainly ample room for error in such estimates. The income side is no less subject to error; just think of the incentives to avoid reporting proprietor income.

The difference between the two estimates of GDP is known as the Statistical discrepancy. The statisticians making the accounts strive to keep it as close to zero as possible. The American accountants, however, claim that they will not distort one of the components from what they believe is their best estimate of it just to get a low Statistical discrepancy. In most other countries, the accountants adjust the components until the discrepancy is eliminated. In the U.S. NIPA, National income and Personal income are based on the income measurement, while Gross domestic product is based on the product measurement. The Statistical discrepancy slips between the two so that the tables balance.

I must confess to you that I find the working out of the concepts of the NIPA and the devising of ways to estimate all the components the most impressive and important contribution that economics has made to understanding the world in which we live. Today, we take these accounts more or less for granted, but it is well to remember that in 1930 they did not exist.

We can now summarize this first model in the following *master* file which shows the actual commands to the software we use to compute this model.

```
# Master File for Model 1
# 31 variables, 8 identities, 23 exogenous variables
# Gross domestic product
f gdp = c + v + x - m + g
# Gross national product
f gnp = gdp + irrow - iprow
# Net national product
f nnp = gnp - ncca
# Personal income before government action
f pibg = nnp - nbctpn + pibtp - nprf + pidiv - netintm + piint - nsd
# Personal income
f pi = pibg + pigsb - nisurp + nisub - nitpi - nconsi
# Personal disposable income
f pidis = pi - piptax
# Personal savings
f pisav = pidis - c - piipcb - piptt
# Addendum: National income
f ni = nnp - nsd
end

# Graphical checks of the identities
ti GDP
gr gdp a.gdp
ti Personal income
gr pi a.pi
ti Personal savings
gr pisav a.pisav
```

```
ti National income
gr ni a.ni
```

Lines that begin with a # are just comments for the benefit of the human reader and are ignored by the computer. Lines that begin with an “f” form the variable on the left by the expression on the right. The “end” command signals the end of the master file for the model building program. Lines below it do not go into the model but have another function.

The lines below the “end” command provide graphical checks of the correctness of the identities. If you start *G7* and on the main menu click Model | Master, you will see this master file in the green edit window to the right. Normally, you do not “run” a master file, but with this model of only identities it makes sense to do so precisely to check the identities. Click the Run button on the main menu of the green editor, and all the commands will be executed by the *G7* program. The variables on the left of the “f” commands will be formed and placed in a *workspace* bank. When *G7* needs a variable, it looks first in this workspace; if it finds no variable of the required name, it looks in the assigned bank “a”. When *G7* has been started in the \ami directory, the bank assigned as “a” is Quip, the Quarterly Income and Product bank. Thus, when this file has been run through *G7*, the variables created by the “f” commands will be in the workspace bank. If the identities are correct, variables like *gdp*, *gnp*, *pi*, *pisav*, and *ni* should have the same values in both banks. To check that they do – that is, that we have made no mistake in writing down the identities – we can graph what should be the same series from the two banks. We can force *G7* to go to assigned bank “a” for a variable by putting an *a.* in front of the variable name. Thus, the command

```
gr gdp a.gdp
```

will graph *gdp* from the workspace in the first (red) line and *gdp* from the assigned bank “a” with the second (blue) line. If our identities are correct, the two lines will coincide. In the present case, passing this check is relatively simple. To see how the test works, however, you may want to deliberately mess up one of the identities, “run” the false Master file, and note how the graphs reveal that something is wrong. Be sure to return the Master file to its correct state and run it again.

There is absolutely nothing in this model to object to. It will reproduce the historical data exactly. And yet it is a rather strange model. If we increase government purchases, *g*, by \$100 billion, Personal income will increase by \$100 billion, Personal consumption will be unchanged, and Personal savings will increase by \$100 billion. That is surely an implausible result. Are consumers utterly indifferent to their income in deciding how much to consume? Certainly not. That observation leads us to Model 2. Because we can completely analyze Model 1 so easily, we will not actually build and run it. That we will do first with Model 2.

Before moving to it, however, I should add that if perhaps you feel these NIPA accounts are somewhat complicated, you should see those used by nearly all other countries. They use a structure known as the System of National Accounts (SNA) developed by the United Nations and other international organizations. In the SNA, the basic pure-identity model requires nearly four times as many variables and three times as many identities as does our model above. In models based on the SNA, the identity checks are very important. I have on several occasions spent several days getting identities correct in models based on these accounts.

Model 2. A Behavioral Ratio for Personal Consumption

We have noted the peculiarity in Model 1 that consumption is independent of Personal disposable income. Let us introduce a ratio between them, a ratio that will be constant in any one period but which may vary over time. Because it expresses the behavior of consumers, it is natural to call it a *behavioral ratio*. The command in our software for forming this exogenous ratio is

```
fex cBR = c/pidis
```

The “fex” stands for “form the exogenous variable”. It is important to realize that the “fex” command forms the variable on the left and puts it into the model’s data bank, but it *does not put the equation into the model*. In this respect, it differs fundamentally from the “f” command which not only forms the variable but also puts the equation into the model.

Once we have formed the behavioral ratio, we use it in the equation

```
f c = cBR*pidis
```

The Master file is changed at the end. As far down as the definition of Personal disposable income, it is unchanged. Starting from that point, the *end* of the new master file for Model 2 is:

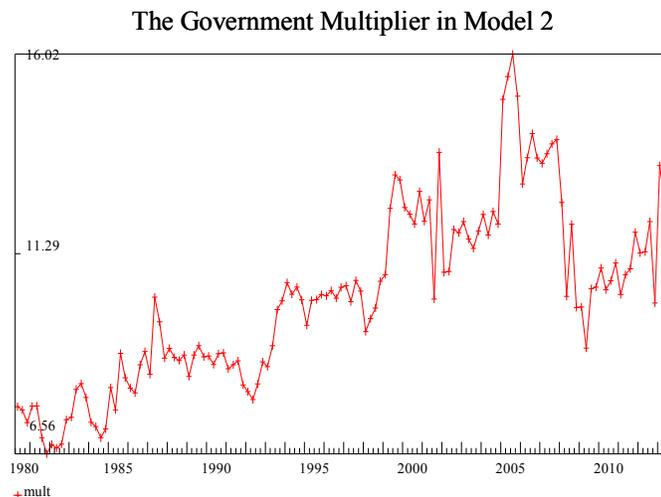
```
# Personal disposable income
f pidis = pi - piptax
# Personal consumption expenditure
fex cBR = c/pidis
f c = cBR*pidis
# Personal savings
f pisav = pidis - c - piipcb - piptt
# Addendum: National income
f ni = nnp - nsd
check c .2
end
```

This procedure of introducing *cBR* at first looks entirely circular. You probably wonder, what is the purpose of introducing *cBR* if we only use it to derive *c*? *c* was used to derive *cBR* in the first place! However, it is important to remember that this new variable, *cBR*, is exogenous to the model, and allows us further insight into the workings of our model economy. We can determine *cBR* for all years for which we already have economic data. For any future years, we will have to supply it, which you will see should not be very hard. Having *cBR* as an exogenous variable allows us to run counter-historical simulations which ask the question: “What if the consumption ratio had been 5% higher in all years than it was observed to be?”

(From here on, we omit the commands to check the identities.) The key new commands, lines 4 and 5 and the next to last, have been set in boldface type. Notice that Personal consumption expenditure, *c*, has now become an *endogenous* variable and that we have introduced a new exogenous variable, *cBR*, so the number of exogenous variables has not been changed. Furthermore, the model remains tautological in the sense that -- given the historical values of the exogenous

variables -- it will reproduce exactly the historical values of the endogenous variables.

If, however, we do a *counter-historical simulation* by running the model over the historical period but with one (or more) of the exogenous variables changed from the historical values, then we get a big difference from Model 1. For example, we can increase the value of Government purchases, g , by 20 in all periods. The ratio of the increase in gdp to this increase in g is called the “ gdp multiplier of g ” or just the “GDP/G multiplier”. For Model 1, that ratio was exactly 1.0 for all periods. The figure below shows its values for Model 2.



The key to understanding the big difference between Model 2 and Model 1 is the next to last line of the master file, the *check c .2*. Note that a value of c was assumed at the top of the master file for the calculation of gdp . At the end, just above the “check” command, the value of c implied by all the equations is calculated. Will it be the same as the value used at the top? If all of the other variables are at their historical values, then the answer is Yes. But if one of them, say g , has been changed to some other value, then the answer on the first try is No. Our “check” command checks to see if the difference between the initial and the implied value is less than .2. If so, it allows the model to go on to the next period; if not, it sends the calculations back to the top of the master file, but this time with the value of c implied on the previous pass as the value assumed at the top. This method of solving the equations of the model, which is known as Seidel iteration, is not certain to converge. In practice, however, it almost always does so in a well formulated model.

Clearly the economic properties of Model 2 are widely different from those of Model 1. We now have, almost certainly, too much multiplier, for we have taken no account of the fact that taxes should also go up as income rises. That and other matters will be handled in Model 3.

Before leaving Model 2, however, we should perhaps ask why there has been such a rise over time in the multiplier. You may perhaps recall from a macroeconomic course that the multiplier should be one over the savings rate. In our terms that would be $1/(1 - cBR)$. In fact, calculating the multiplier by this formula gives exactly the same graph as that shown above. Thus, the cause of the rise in the multiplier is the decline in the savings ratio, or more properly, the rise in the ratio of consumption to

income.

Exercise

1.2. Build and run Model 2. Run it first with all exogenous variables at their historical values. Then run it again with one or more of them changed. Compare the results graphically. Compute and graph the multiplier. Here are the steps:

1. Start the *G7* program by double-clicking its icon. If working on your own machine, you will first need to install the software which accompanies this book, including the Borland C++ free compiler. Follow the directions in the `readme.txt` file. If you are working on a network for a course, your instructor will explain how to find it. *G* immediately asks you to select the folder and configuration file. You want to “look in” `c:\ami` and the file you want to “open” is `g.cfg`.
2. Prepare the Master file. On the main menu bar, click `Model | Edit Master File`. (The use of the | symbol will be our shorthand way to say: click `Model` and then on the drop-down menu, click `Edit Master File`.) An editor window opens in the upper right corner of your screen and displays the Master file of Model 1. Do “`File | Save as ..`” to save it as `Master1` in case you need it again. Edit the file to transform it into the Master file of Model 2. The editor works pretty much the way that any Windows-based word processor works. Again click “`File | Save as ..`” on the Editor’s menu and save the file as `Master` (*not* `Master2`).
3. Build the model. On the main menu of the blue window to the left, click `Model | Build Macro Model`. This simple step calls a console application program to build an executable model from the current version of the Master file. If all goes well, all you have to do is tap a key at the end of the operation to return to *G7*. This step, it must be emphasized, works from the file presently called `Master` – not `Master2`, or `Master3`, or anything else.
4. Run a historical simulation. On the main menu, click `Model | Run Macro Model (F7)`. A window will request the starting date, ending date, results file name, and fix file name. To say that you want the model to run from the first quarter of 2000 to the third quarter of 2013, enter the dates as follows: `2000.1` and `2013.3`. (Run your model up to the most recent period for which you have data. It may be more recent than the example I have used.) Let us call the results file “`histsim2`” (for “historical simulation of model 2”) and use “`rroadj`” for the fix file. Exactly what that strange name means we will explain soon. Then click `OK`. You should get a black screen informing you that you are running the *AMI* model and giving you a `] prompt`. Respond with “`ti xxx`” where `xxx` is up to 80 characters giving your title for this run of the model. Take the time to give a meaningful title such as “`Historical Simulation of Model 2.`” You then get the `] command` again; reply with “`run`”. The model runs, some printout goes by too fast to read, and the program stops with the message “`Tap any key to continue.`” This pause allows you to see error messages. With good luck, there will be none.
5. Graph the results of the historical simulation. First you must adapt the `hist.shw` file to the current model. Open it in the green editor window. Commands at the top of the file open

the banks from which the data will be taken. In the present case, we should make them

```
bank histsim2 b
bank bws c
```

6. On the editor's main menu, click File | Save as ... and save the file as hist2.shw. Then click Run on the editor's menu. A graph and a run control panel should appear. Click the "continue" button on this panel, or tap the letter 'c', to view all the graphs, one after another. Since the model is tautological, all the graphs should show only one line, though if you look closely you will see that there are red crosses inside the blue squares. On the control panel, you can click the "Max" button to maximize the size of the graph. Click the "Normal" button to return the graph to its size before hitting the "Max" button.
7. Now to make a run with government purchases of goods and services, g , increased by \$100 billion, repeat step 4 but call the results "highgov2" and at the] prompt give a title like "ti Government increased by \$100 billion." Then, still at the] prompt, do

```
type g
update g +100
type g
```

You should see that the values of g have all been increased by 100. Then give the "run" command. You are doing a *counter-historical simulation*.

8. Now make graphs comparing the historical simulation, which is in the bank *histsim2*, with the counter-historical simulation, which is in the bank *highgov2*. Edit the file *comp.shw*. You can do "ed comp.shw" in the white command box, or you can click Editor on the main menu and the Open on the Editor's menu and open comp.shw Window's style. Make it begin with the commands

```
bank highgov2 b
bank histsim2 c
```

At the end of this file, you should put the commands to compute the multiplier and graph it. Here they are:

```
f mult = (b.gdp - c.gdp)/(b.g - c.g)
ti The Government Multiplier in Model 2
gname mult
gdates 1980.2 2013.3
gr mult
gdates 1980.1 2013.3
```

The "gdates" commands control the dates for the beginning and end of the graphs. The *mult* variable will be zero in the base year of the model run, for the model is not computed for that base period. Hence, in order to graph only the non-zero values of the multiplier, we begin the graphs in the second period of the run. Adapt the second date in these commands to the most recent data in your data bank. After this use, the second "gdates" command restores the dates to their normal values. Click the File | Save as ... on the main menu of the green editor window. Save the file as *comp2.shw*, that is, as a file adapted to making the comparisons for Model 2. You may then "run" the file by clicking "Run" on

the main menu of the green editor window. That will save the file and execute the commands in it in G. You should see the graphs go by as you click the Continue button on the left of control panel which appears along with the graphs.

9. Now we want to save the graphs created in step 7 so that they can be imported into a word processor. Look again at the file *comp2.shw* in the editor. You will see triplets of lines like this:

```
ti gdp - Gross Domestic Product
gname gdp
gr b.gdp c.gdp
```

The second line is what interests us here. It gives the filename of the file into which the graph will be saved if you click the “Save” button on the run control panel. To make the graphs again, you can either repeat the command “add comp2.shw” or, since you already have *comp2.shw* in the editor, you can click “Run” on the Editor menu. On several of the graphs, click the “Save” button on the run control panel. Let us say that you saved the gdp graph. Actually, you created a file named *gdp.wmf*, the “wmf” standing for Windows Metafile. Start your word processor and bring in this file. In WordPerfect the command is Insert | File. In Word and Open Office it is Insert | Picture. Drag the little square handles on the graph to size as you like.

Model 3. More Behavioral Ratios and Shortcutting Exogenous Variables

In the preface, I said that models are “serious business” because they are playthings. A good plaything should be simple enough to make playing with it fun. Model 2, while clearly a step in the right direction, still has an ungainly number of exogenous variables that have to be set whenever we want to run it. Furthermore, it still ignores a number of important behavioral relations. Model 3 helps with both those problems. It puts in several more important behavioral relations and eliminates a number of exogenous variables of secondary interest. In particular, there are no less than eleven exogenous variables between *gdp* and *pibg*. If we are prepared for the moment to ignore the distinctions among them, then we can make the entire transition in one step by introducing an exogenous behavioral relation *pibg* and *gdp*. Model 3 also adds a behavioral relation between Indirect taxes and GDP. It drops the distinction between “contributions for social security” and “personal taxes” and specifies the sum of the two as a behavioral ratio to a *pibg* less the Indirect taxes. At the same time, it adds important behavioral ratios between:

- Imports and private demand
- Personal savings and Personal disposable income
- Interest paid by consumers to business and Personal disposable income
- Personal transfers to foreigners and Personal disposable income.

The last three replace the single ratio between Personal disposable income and Personal consumption expenditure which was introduced in Model 3. Here is the new master file:

```

# Master 3
# 21 variables, 11 exogenous (not counting time)
# 7 behavioral ratios for m, pibg, nitpils, ptax, piipcb, piptt, pisav
# v, x, g, npctr remain exogenous
checkdup y
fex time = time
# Behavioral ratio for imports
fex mBR = m/(c+v+x)
f m = mBR*(c+v+x)

# Gross domestic product
f gdp = c + v + x - m + g

# Shortcut to Personal income before government action
fex pibgBR = (gdp + irrow - iprow - ncca - nbctpn - nsd - niprf +
pidiv - netintm + piint + pibtp)/gdp
f pibg = pibgBR*gdp

# Calculate personal tax base
fex nitpilsBR = (nitpi - nisub + nisurp)/gdp
f nitpils = nitpilsBR*gdp
# ptaxbase: Personal tax base
f ptaxbase = pibg - nitpils

# ptaxBR is personal tax rate including Social Security
fex ptaxBR = (piptax + nconsi)/ptaxbase
f ptax = ptaxBR*ptaxbase

# Personal disposable income
f pidis = pibg + pigsb - nitpils - ptax

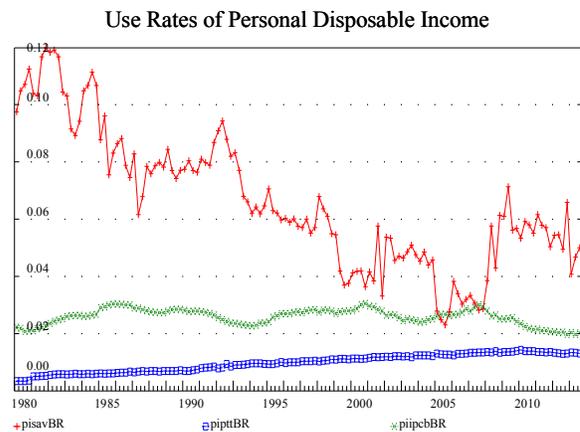
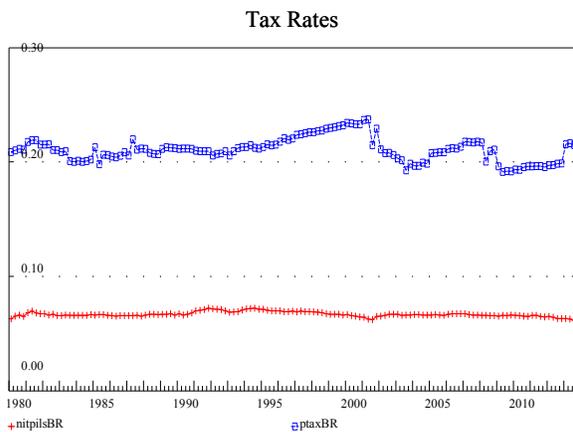
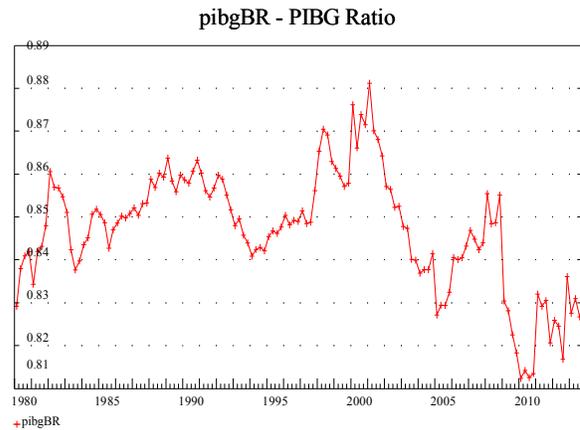
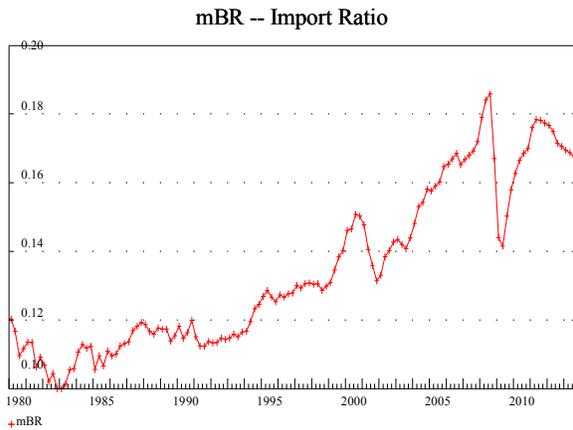
# Behavioral ratios for piipcb, piptt, and pisav
fex piipcbBR = piipcb/pidis
f piipcb = piipcbBR*pidis
fex pipttBR = piptt/pidis
f piptt = pipttBR*pidis
fex pisavBR = pisav/pidis
f pisav = pisavBR*pidis

# Implied consumption
f c = pidis - piipcb - piptt - pisav

check c .2
end

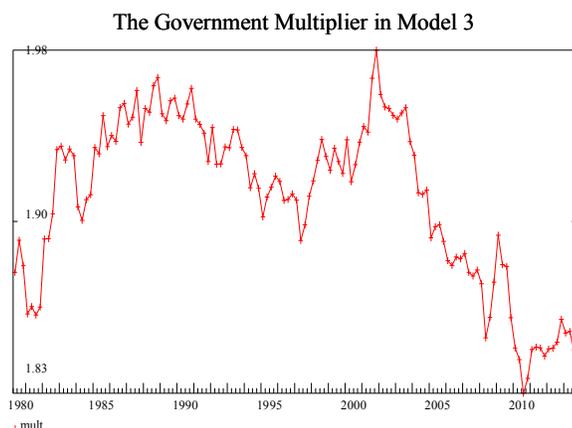
```

In Model 3, only 4 of the original 23 exogenous variables remain, namely v , x , g , and $pigsb$. Seven behavioral ratios, however, have been added, and one may well wonder how stable or how hard to forecast they are. On the next page are graphs of them that should help answer that question.



The scales have been chosen so that the variations in the variables can be seen. The PIBG ratio has remained surprisingly stable between .81 and .88; the import ratio is more volatile between .10 and .18 with a distinct tendency to rise, although it had a huge decline in 2009. The indirect tax rate (*nitpilsBR*) has remained stable throughout the period shown. The personal tax shows more variation. The Reagan tax cuts produced the slight dip in the personal tax rate between 1981 and 1984; after 1984, they began to crawl back up. The rise since 1995 reflect various “revenue enhancements” by the Clinton administration. Finally, the lower tax rate starting in 2001 reflects the Bush tax cuts, and a weakening economy. Clearly the most dramatic movement is in the Personal savings rate. The collapse and subsequent rise of personal savings is a matter into which we must look closely as we develop the model of the economy. With the exception of the savings rate, it would seem possible to make pretty good guesses about the future values of these ratios for a few years ahead.

The first game to play with a model, of course, is to calculate its GDP/G multiplier. The result is shown in the graph below.



The values range between 1.8 and 2, a value typical for a purely Keynesian model with no supply constraints, such as we have here. When we account for inflation and monetary effects, the value should come down further.

Exercise

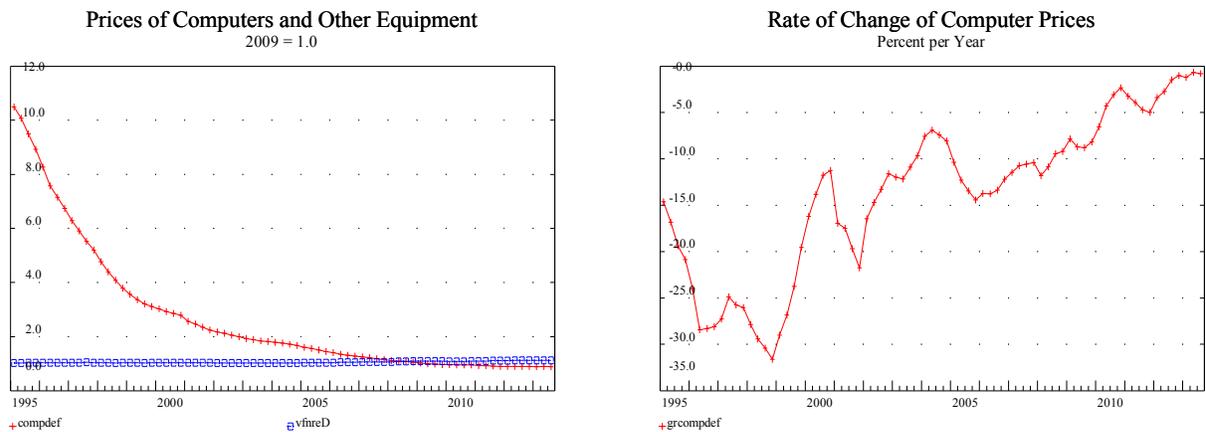
1.3. Bring your Master file up to the level of model 3. Build and run the model in both historical and counter-historical simulation. How does it respond to the increase of \$100 billion in government purchases? Proceed as in the previous exercise; be sure to include a graph of the multiplier. You do not have a file that automatically makes the graphs shown above, but with what you now know, you should have no trouble making them. Remember that, after building a model, all variables in the model are in the bank called *bws*.

Model 4. Seeing Past Inflation

So far, all of our variables expressed in dollars have been in *nominal* terms, that is, in terms of the dollars actually spent. We have made no attempt to account for inflation. But anyone old enough to remember can tell you that forty years ago (1967) a dollar would buy a meal that today costs six or more. Consequently, these nominal or *current dollar* measures greatly overstate the growth that has occurred in the economy. The makers of the NIPA have, therefore, produced another set of figures for the product side of the accounts by expressing each item -- consumption, investment, exports, imports, and government purchases -- in the prices of one particular, base year. Presently (2014), that base year is 2009, but the base year is shifted forward every five years. These accounts are said to be in *constant dollars*, or *constant prices*, or *real* terms. These are all equivalent expressions.

As long as one is dealing with products which are in common use over the period studied, there is no great conceptual problem in how to measure the change in an item's price. For example, there is no great problem in putting the 1982 consumption of milk or jeans into 2009 prices. But what about the 2009 price of a 1982 computer which was no longer made by 2009?

In fact, this very question of the pricing of computers has far-reaching effects in the U.S. NIPA. In the mid 1980's, BEA adopted a method known as “Hedonic” price indexes for computers, which it has extended to some other high-tech items. It looked at a variety of computers in, say, both 2012 and 2013 and recorded price and physical characteristics such as speed, RAM size, and hard disk size. By doing a regression (by methods we shall study closely in chapter 2) of price on the physical characteristics of the 2012 machines, an equation for the 2012 price as a function of physical characteristics could be determined. Coming then to the 2013 models, this function could be applied to say what their 2012 price would have been. By comparison with the actual 2013 prices, a price index between the two years can be deduced. The figure on the left below shows the resulting price index for computers and that for all other equipment. The figure on the right shows the annual percent change in computer prices -- note that the numbers are all negative.



The contrast between the treatment of high-tech items and everything else is stunning and, I believe, problematic. This method of evaluating computers is not the only possibility. For example, one might evaluate the marginal product of a computer by assuming that various factors of production are paid their marginal product. When I first began programming computers in 1959, the computer cost \$60 per hour and my wage was about \$2.00 per hour, so the marginal product of that computer was about 30 times my marginal product. The machine on which I am typing these words is at least a thousand times more powerful than that machine, but it costs less than one percent of what I cost per hour. If I generously suppose that my marginal product has not declined over the intervening years, that *Univac-1* in 1959 must have had a marginal product about 3000 times that of the machine in front of me now — not a thousandth of it! Thus, there seems to me to be no compelling reason to use the BEA method of deflation; and, indeed, considerable reason to doubt that it is even of the right order of magnitude. Consider another example. In the 1970's, many universities and companies were equipped with usually one mainframe computer. It usually took up the space of an entire room, and surely its power is dwarfed by today's personal machines. However, these mainframes still computed an enormous amount of data and information for whomever would line up. The BEA deflator takes these extremely important machines and writes off them and their contributions to GDP as nearly nothing.

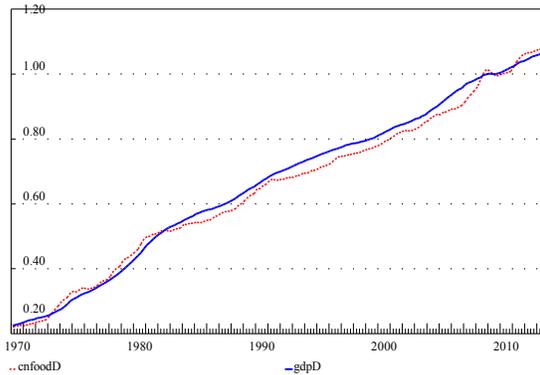
The use of the BEA computer and other high-tech deflators affect substantially the rate of growth of real GDP. To prevent these effects from turning the accounts into nonsense, the BEA has offset the effects of the Hedonic indexes by also adopting a non-linear procedure known as “chain weighting.” For each pair of adjacent years, say 2012 and 2013, the real growth ratio of each component of GDP is computed in two ways. In the first, all items are priced with the prices of the first year; in the second, all items are priced with the prices of the second. We may call the first index A and the second B. The geometric mean of the two, \sqrt{AB} is then taken as the real growth ratio of that component between those two adjacent years. If the growth between non-adjacent years is needed, it is taken as the product of all the year-to-year growth ratios between the two years, that is, the year-to-year ratios are *chained*. To get GDP or a component in prices of a base year, say 2009, one just moves the 2009 nominal GDP forward or backward by these year-to-year growth rates.

The official constant-price series are found in the QUIP data bank with names derived from the name of the current dollar series by adding a \$. Thus, the official series for real GDP is gdp\$.

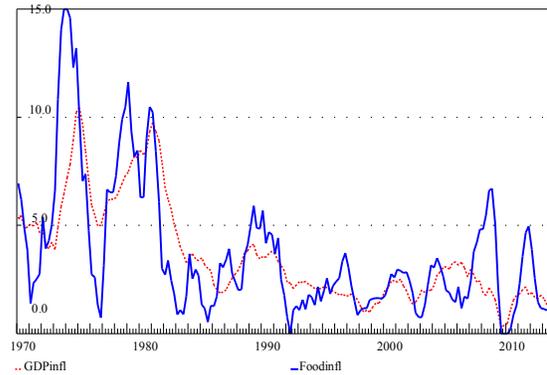
The use of chaining means that it is never necessary to evaluate the output of computers (or other goods) in prices more than one year distant from the year in which they were made. That is certainly an advantage, but chaining also has a major disadvantage. Namely, total real GDP, say for 2012 in prices of 2009, is not the sum of the components of real GDP. The same is true with the components. For example, the official 1970 value of fixed investment (vf\$) in 2009 prices is 27 percent smaller than the sum of its three major components: residential construction (vfr\$), non-residential construction (vfnr\$), and non-residential equipment (vfne\$). It will clearly be a nuisance in modeling if a total is not the sum of its parts.

Because I have serious reservations, as already explained, about the computer and other high-tech deflators and do not wish to be plagued with totals that are not the sum of their parts, I am going to adopt a radical simplification for our modeling. I am going to use the deflator for personal consumption expenditures on food as the deflator for all series in the national accounts. I choose food because it is important and I tend to trust the methods of making indexes for food prices. I also notice that when I need a subjective price index to put some dollar amount from twenty or thirty years ago into current prices, I tend to try to recall food prices. Remarkably, the trend of food prices and that of the GDP deflator computed with Hedonic indexes and chain weighting are very similar, as seen below. The food prices have, however, been somewhat more volatile, as shown in the right graph. In both graphs, the heavier, smoother line is the official GDP deflator. GDP and GDP product-side components deflated by the GDP deflator have names in the QUIP databank derived by adding R (for “Real”) to the variables name in current prices.

The Official GDP and Food Deflators
2009 = 1.0



Inflation in the Official GDP and Food Deflators
Annual Growth Rate



With the question of the computer deflator settled, we can turn to the proper subject of Model 4, the accounts in real terms. There are many reasons for wanting accounts in real terms. If we want to use GDP or Personal consumption expenditure as an indicator of welfare, it is certainly the measure in real terms that is relevant. If we want to relate employment to the NIPA, we clearly need measures in real terms. If we want to tie investment to growth in output, it is only real growth which is relevant. Also, forecasting exogenous variables in real terms is generally easier than in nominal terms. The model should make the conversion from real to nominal for us.

In the official accounts, each item on the real side has its own deflator, and there are no deflators on the income side except for an addendum item which gives Personal disposable income deflated by the Personal consumption expenditure deflator. Since we are using accounts in constant “GDP deflator dollars,” all items have the same deflator. We will make the GDP deflator an exogenous behavioral ratio, and set the consumption deflator relative to it. All the exogenous dollar amounts will be exogenous in real terms. Here is the master file of Model 4.

```
# Master 4: Product side in real terms
# 13 exogenous (not counting time)
# 7 behavioral ratios for m, pibg, nitpils, ptax, piipcb, piptt, and pisav.
# gdpD exogenous
# vR, xR, gR, npctrR exogenous in constant dollars

checkdup y
fex time = time
# Behavioral ratio for imports
fex mBR = mR / (cR + vR + xR)
f mR = mBR * (cR + vR + xR)

# Real Gross domestic product
f gdpR = cR + vR + xR - mR + gR

# Current price values
f m = mR * gdpD
f v = vR * gdpD

# Nominal GDP
f gdp = gdpR * gdpD

# Shortcut to Personal income before government action
fex pibgBR = (gdp + irrow - iprow - ncca - nbctpn - nsd - niprf +
             pidiv - netintm + piint + pibtp) / gdp
```

```

f pibg = pibgBR*gdp

# Calculate personal tax base
fex nitpilsBR = (nitpi - nisub + nisurp)/gdp
f nitpils = nitpilsBR*gdp
# ptaxbase: Personal tax base
f ptaxbase = pibg - nitpils

# ptaxBR is personal tax rate including Social Security
fex ptaxBR = (piptax + nconsi)/ptaxbase
f ptax = ptaxBR*ptaxbase

# Government transfer payments to persons, exogenous in real terms
fex npctrR = npctr/gdpD
f npctr = npctrR*gdpD

# Personal disposable income
f pidis = pibg + pigsb - nitpils - ptax

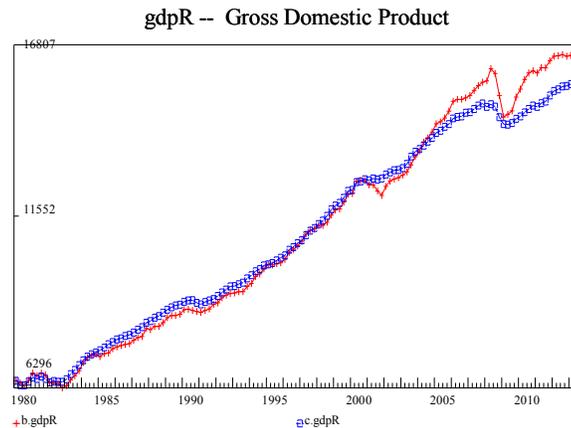
# Behavioral ratios for piipcb, piptt, and pisav
fex piipcbBR = piipcb/pidis
f piipcb = piipcbBR*pidis
fex pipttBR = piptt/pidis
f piptt = pipttBR*pidis
fex pisavBR = pisav/pidis
f pisav = pisavBR*pidis

# Implied consumption
f c = pidis - piipcb - piptt - pisav
# convert to real terms
f cR = c/gdpD
check cR .2
end

```

It is natural to be asking at this point, “Very well, as long as we know all the exogenous variables and behavioral ratios exactly, these models enable us to calculate GDP exactly. But if we are forecasting, we don’t know them. How good are the models if we must make rough guesses for the exogenous variables?” A partial answer may be had by seeing how well the models would have done in the past had we used constant values for all the behavioral ratios. I looked over the values of each behavioral ratio and picked the following typical values: $pibgBR = .847$, $mBR = .134$, $nitpilsBR = .067$; $ptaxBR = .211$, $piipcbBR = .025$; $pipttBR = .010$, $pisavBR = .066$. When the model was run with these values from 1980.1 to 2013.3, it produced the simulation shown by the line marked with a + in the graph below.

The model (red line, +'s) did quite well up through 2000 but by 2006 it went racing off beyond the economy’s actual course (blue line, squares). In fact, exports were rising considerably, but the rising exports went with a rising share of imports. The neglect of this rise (we kept mBR constant) is responsible for the runaway growth in the simulation. My fundamental impression from this graph, however, is that it is possible to model the real growth of the economy surprisingly well given just investment, exports, government purchases and transfers. Of course, explaining the movements in the major behavioral ratios would improve the simulation. A method for doing so is introduced in Model 5.



Exercise

1.4. Bring your model up to the level of Model 4. Run the counter-historical simulation with gR increased by 100. Using *comp.shw*, make graphs of the effects on the new, real-terms variables such as $gdpR$, cR , and so on. Graph the multiplier.

Model 5. Modeling Behavior with Regression

So far, we have been able to replace one exogenous variable with another which is easier to comprehend or forecast. But we have not reduced the number of exogenous variables. With Model 5, we will strike investment from the list of exogenous variables but keep it in the model, now as an endogenous variable. We will do so by fitting two equations by the method of least squares, one for vfR , fixed investment, and one for viR , inventory investment. We will also replace the rather volatile behavioral ratio for imports, mR , with an estimated equation. We will, in the process, reduce the number of exogenous variables by two.

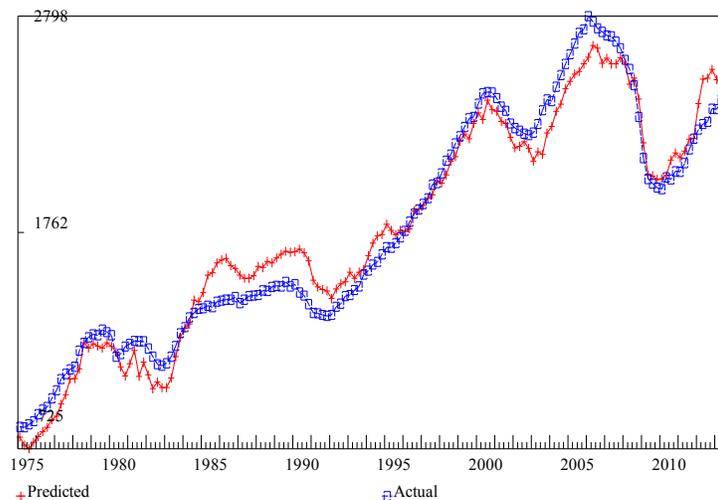
The fixed investment equation explains vfR as a linear function of estimated replacement and lagged values of changes in real GDP. The commands to the computer, using the *G7* software, are put into a file, called *vfR.reg* in this case, and this file is then executed by *G7*. The contents of *vfR.reg* are just

```
# vfR.reg - Regression to explain fixed investment
catch vfR.cat
save vfR.sav
lim 1975.1 2013.3 2013.3
ti Gross Private Domestic Fixed Investment
f gppR = gdpR - gR
f d = gppR - gppR[1]
f replace = .05*@cum(stockf,vfR[4],.05)
con 100000 1.0 = a2
r vfR = replace, d[1], d[2],d[3],d[4],d[5],
    d[6],d[7],d[8],d[9],d[10],d[11],d[12]
save off
gr *
```

catch off

The results of the regression are a graph showing how well the equation fits and two files with the results. One of these files, created because of the command in the first line, is called the *catch* file and is intended to be looked at by humans and included in books and papers. The other, created because of the command in the second line and called the *save* file, is intended to be read by the computer in making a model. The first two commands in the *vfR.reg* file just name the catch and save files. The third, with the *lim* command, specifies the period over which the equation is to be fit; the fourth supplies a title for the graph and the catch file. Next we define gross private product *gppR* as *gdpR* less government *gR*. Then comes the definition of the variable *d* as the difference between one period's value of *gppR* and the previous period's value. (The *[]* notation in *G7* is to indicate lags of variables.) The next line defines the stock of fixed capital, *stockf*, by cumulating investment with a depreciation rate of 5 percent per quarter and the sets replacement (called *replace*) equal to 5 percent of the stock. We will explain the *@cum* function further in a later chapter; suffice it to say here that it does just what an accountant using declining balance depreciation does. Next is a "con" command, which will constrain the first coefficient in the regression. (The topic of constraints will be covered more fully as we go into the investment equation in more detail later.) Finally, we come to the command that we have been leading up to, the one that fits a linear function of the variables listed on the right side of the = sign to the variable on the left. This fitting is often called *linear regression*; hence the *r* as the name of the command. We will look carefully at how this fitting is done in the next chapter. The remaining commands just draw the graph of the fit and close the save and catch files. Here is the graph for our equation.

Gross Private Domestic Fixed Investment
Fitted vs. Actual



The fit is clearly quite close, almost uncomfortably close if you consider how simple the equation is. Notice that it does not use the interest rate or any other financial or monetary variable. In producing a fit this close without such a variable, the computer seems not to have realized that the whole of macroeconomics pivots on the interest rate in the IS curve. Such a heretical computer should probably be burned at the stake.

In the top panel:

Statistic Name	Description
SEE	Standard error of estimate, or the square root of the average of the squares of the misses or "residuals" of the equation. (Not adjusted for degrees of freedom.)
RSQ	(pronounced r-square) = Coefficient of multiple determination.
RHO	(Greek rho) = Autocorrelation coefficient of the residuals.
Obser	Number of observations.
SEE+1	The SEE for forecasts one period ahead using rho adjustment.
RBSQ	(pronounced r-bar-square) = Coefficient of multiple determination adjusted for degrees of freedom.
DW	Durbin-Watson statistic; contains same information as does RHO.
DoFree	Degrees of freedom = Obser minus number of independent variables.
from	Starting date of the regression
to	Ending date of the regression.
MAPE	Mean absolute percentage error.

Across the lower columns, for each variable:

Statistic Name	Description
Reg-coef	Regression coefficients for the variable.
Mexval	Marginal explanatory value: the percentage increase in SEE if the variable were left out.
Elas	Elasticity at mean.
NorRes	The "normalized" sum of squared residuals (SSR) using this and all preceding variables, that is the SSR using this and preceding variables divided by the SSR using all variables.
Mean	Mean of the variable.
Beta	What the regression coefficient would be if both dependent and independent variables were in units which made their standard deviations equal to 1.0.

It is always important to ask whether the regression coefficients are of the correct sign and reasonable magnitude. In this equation, the coefficient on *replace*, 0.83073, seems reasonably close to its theoretical value of 1.0; the coefficients on *d* are all positive. One might expect them to fade away to zero more rapidly than they do.

The *save* file for this regression is meant for reading by the computer, but is legible by humans. In case you are curious, here is what it looks like for this regression.

```

ti Gross Private Domestic Fixed Investment
f gppR = gdpR - gR
f d = gppR - gppR[1]
f replace = .05*@cum(stockf,vfR[4],.05)
r vfR = 145.145148*intercept +
      0.895139*replace +

```

```

0.467734*d[1] +
0.428344*d[2] +
0.381594*d[3] +
0.562940*d[4] +
0.489958*d[5] +
0.370981*d[6] +
0.382080*d[7] +
0.342839*d[8] +
0.431552*d[9] +
0.351016*d[10] +
0.301090*d[11] +
0.514760*d[12]

```

d

As you see, it contains the commands to form the variables and the coefficients of the equation found by the regression. It omits other data intended to help in thinking about the results of the regression.

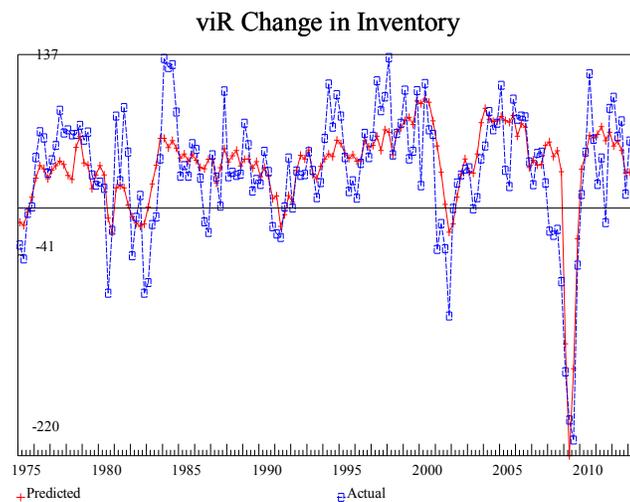
The equation for inventory investment is estimated by executing the following file, called viR.reg:

```

# viR.reg - Regression file for inventory investment
catch viR.cat
save viR.sav
lim 1975.1 2013.3
title viR Change in Inventory
# fs stands for "final sales"
f fsR = cR + vfR + xR + gR
f dfsR = fsR - fsR[1]
r viR = ! dfsR[1], dfsR[2], dfsR[3],dfsR[4]
save off
gname viR
gr *
catch off

```

The ! after the = sign in the *r* command means to leave out the intercept or constant term. Here are the results:



```

save viR.sav
fdates 1972.1 2013.3
lim 1975.1 2013.3
title viR Change in Inventory
# fs stands for "final sales"
f fsR = cR + vfR + xR + gR
f dfsR = fsR - fsR[1]
r viR = ! dfsR[1], dfsR[2], dfsR[3],dfsR[4]
:
      viR Change in Inventory
SEE   =      35.77 RSQ   = 0.5674 RHO =   0.46 Obser = 155 from 1975.100
SEE+1 =      32.03 RBSQ = 0.5588 DW   =   1.07 DoFree = 151 to 2013.300
MAPE  =      193.98
Variable name      Reg-Coeff  Mexval  Elas  NorRes      Mean  Beta
0 viR              - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
1 dfsR[1]          0.17059    8.9    0.39   1.49    80.01
2 dfsR[2]          0.20293    8.2    0.46   1.06    79.35  0.394
3 dfsR[3]          0.04296    0.4    0.10   1.02    79.71  0.083
4 dfsR[4]          0.05006    0.8    0.11   1.00    79.38  0.097

save off
gname viR
gr *
catch off

```

The fit here leaves room for improvement; but given the extreme volatility and small size of this variable, this equation should be usable.

Finally, the import equation is estimated by the commands in the mR.reg file

```

# mR.reg - Regression file for imports
lim 1975.1 2013.3
catch mR.cat
save mR.sav
ti mR Imports
subti Model 5
con 100000 0.5 = a2
con 100000 0.36 = a3
r mR = xR, vR, cR
save off
gname mRreg
gr *
save off
catch off

```

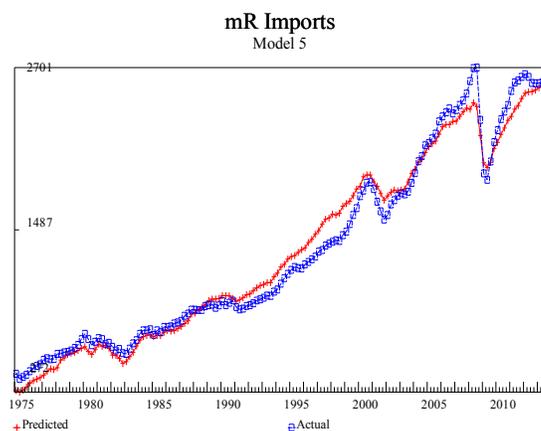
The results are:

```

save mR.sav
ti mR Imports
subti Model 5
con 100000 0.5 = a2
con 100000 0.36 = a3
r mR = xR, vR, cR
:
                                mR Imports
SEE   =      102.43 RSQ   = 0.9790 RHO =   0.95 Obser  = 155 from 1975.100
SEE+1 =       34.70 RBSQ = 0.9786 DW  =   0.11 DoFree = 151 to 2013.300
MAPE  =        7.92
Variable name      Reg-Coeff  Mexval  Elas  NorRes  Mean  Beta
0 mR              - - - - -
1 intercept       -590.68985  99.7   -0.44  48.91  1331.43 - - -
2 xR              0.54276  102.7  0.43   8.50  1051.09 0.379
3 vR              0.30206  48.8   0.41   2.47  1805.65 0.234
4 cR              0.11893  57.3   0.61   1.00  6779.25 0.403

save off
gname mRreg
gr *
catch off

```



The fit is close enough, but the high coefficient on exports is suspicious. It can hardly mean that 54 percent of the value of exports is imported. Rather, it would seem to be a sort of proxy variable for the openness of the economy. I am not happy with this equation, but at this introductory stage, let us accept it.

The master file for Model 5 follows.

```

# Master 5: Regression equations for investment and imports
# 11 exogenous (not counting time)
# 6 behavioral ratios for pibg, nitpils, ptax, piipcb, piptt, pisav
# 3 regression equations for mR, vfR, and viR
# xR, gR, npctrR exogenous in constant dollars

checkdup y
fex time = time

# Investment
add vfR.sav

```

```

add viR.sav

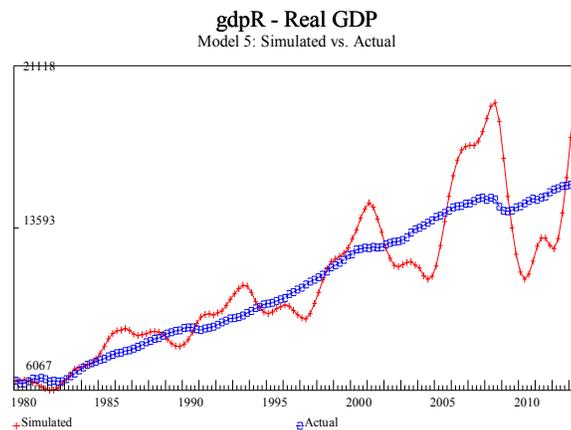
f vR = vfR+viR
# Imports
add mR.sav
# Gross domestic product
f gdpR = cR + vR + xR - mR + gR

```

... and so on exactly as in Model 4.

Note the “add” commands which bring into the model the results of the regressions recorded in the .sav files.

So, how well does our masterpiece simulate the economy if it is given the historical values of all the exogenous variables? The graph below shows the answer.



The model reproduced the general trend of the economy alright, but with a powerful cycle unlike anything in the real economy. In fact, it is reminiscent of the simple model of section 1 of this chapter. Thus we see that it is very easy to produce a model which has cycles. All we need is a multiplier and an investment function based on the “accelerator” theory of investment. Indeed, our graph should make it fairly clear why our investment equation is called an accelerator; whenever the economy turns up, it accelerates that upturn. In fact, this feature of Model 5 is so prominent that we will name Model 5 the Accelerator-Multiplier Interaction model, or *AMI* for short.

The problem we now face is to find out what stabilizes the economy. We shall return to that question after two intervening chapters to explain the theory of regression and some more of the mechanics of model building. You will then be able to estimate the equations, build, and run this model yourself and go on to work on improving it.

CHAPTER 2. BASIC CONCEPTS OF LEAST SQUARES REGRESSION

In this chapter, we will be concerned with how to sum up the lessons of the past about how one variable, say the sales of automobiles, depends upon others, such as price and income. We will see how to fit to historical data equations which express this dependence. Here we will concentrate on the mechanics of this curve-fitting process and say relatively little about the examination and interpretation of the results. That will come in due course in later chapters.

1. What is the Method of Least Squares and Why Use It?

Let us begin directly with a practical problem. Suppose that you need to make forecasts of automobile sales for the next four quarters. Your first question is naturally, "What factors affect the sales of automobiles and how strongly does each of them work?"

A good bet for an important factor is certainly disposable income. It also happens to be a convenient variable because a recent issue of a business magazine shows the forecasts of it made by a dozen different forecasting firms. Let us presume that you have ten years of quarterly data on both auto sales and disposable income. How do you find the relation between the two?

The simplest way to answer that question is to plot auto purchases against income as shown on Figure 2.1 and then draw a line down the middle of the points.

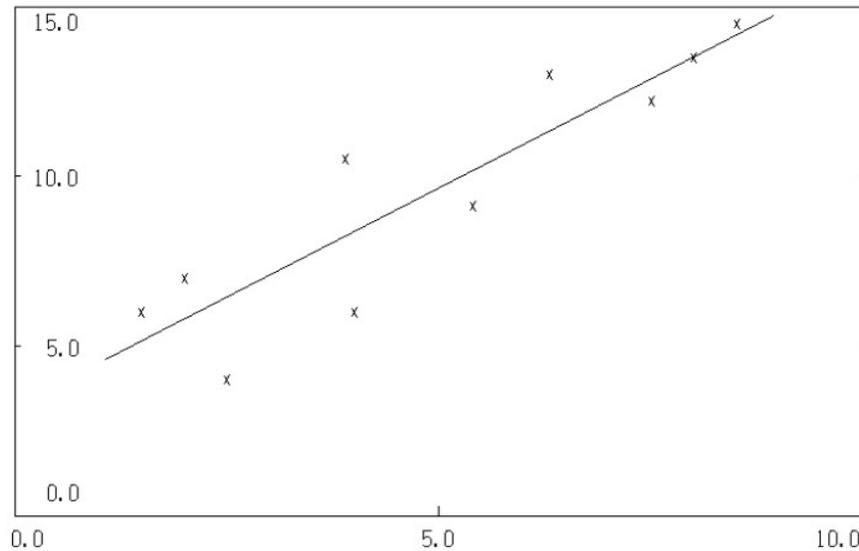
You may use the edge of a transparent ruler or a tightly stretched black string, to help you get a line which seems to fit. Then, given a forecast of income A , we see that automobile purchases of B would be right "on line" with that income. Mathematically, what we have done is to determine b_1 and b_2 of the following equation:

$$(1) \quad y_t = b_1 + b_2 x_t$$

where

- y_t = automobile purchases in quarter t
- x_t = disposable income in quarter t
- b_1, b_2 = constants, often called parameters, to be determined so that the line fits the past experience.

Fig. 2.1: Fitting Data with One Independent Variable



This method of "eyeing in" the line has important advantages over other methods. It allows us to see what we are doing. It is quick, easy, clear, and makes it easy to spot outliers in the data and to avoid putting a heavy weight on them. It is, however, seldom used. Theorists don't like it because no elegant theorems can be proved about it. Practitioners shun it because it is easy to see that it is subjective; more sophisticated methods allow them to cloak their subjectivity in a more opaque garment.

The real problem with the method, however, arises when we realize that automobile sales depend on more than just income. Suppose that we want to take into account the change in income, $dx_t = x_t - x_{t-1}$; then we should have to estimate

$$(2) \quad y = b_1 + b_2x + b_3dx$$

(The b_1 and b_2 of (2) may, of course, be different from those of (1).) To estimate these b 's by eye in this equation, we need a three-dimensional construction to suspend the points in space. Then perhaps we might use a plane of light to find the plane which fits best. But what was simple for one variable becomes something of an engineering feat for two. But now, of course, we realize that automobile purchases should also depend on existing stocks, s , of autos, thus:

$$(3) \quad y = b_1 + b_2x + b_3dx + b_4s$$

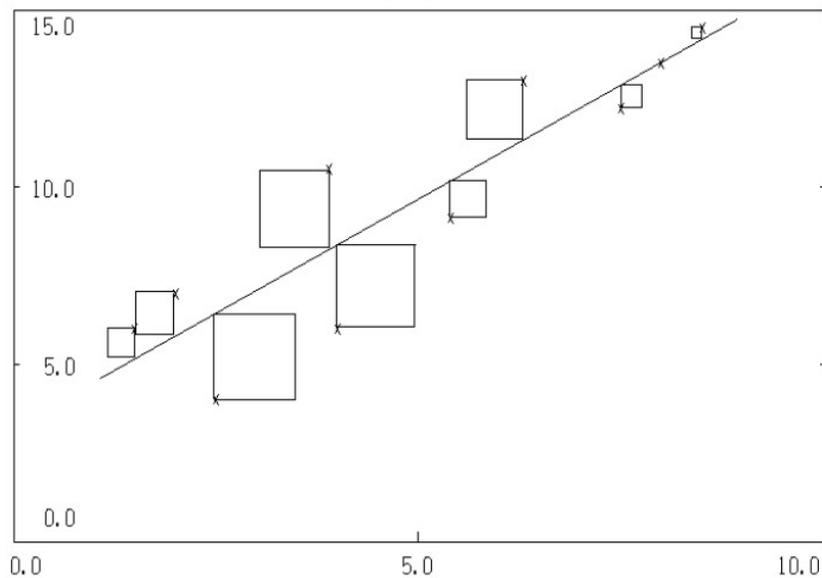
Now we need a four-dimensional structure of some sort; and when you have constructed it, be prepared to tackle a five-dimensional device, for automobile purchases also certainly depend on the price of automobiles, p , so that we should be estimating:

$$(4) \quad y = b_1 + b_2x + b_3dx + b_4s + b_5p$$

Here we seem to be pretty much past hope of making a physical picture to aid us in fitting the equation to the data. If we ever want to estimate an equation with that many parameters -- and for all practical purposes, with any more than two parameters -- we must find some way to replace vision by calculation. That switch can be dangerous, and we shall frequently find that we must check to be sure that the blind calculations have not lost their way.

To find a way to calculate the parameters, let us return to the simplest case, equation (1) and Figure 2.1, and let us agree to choose b_1 and b_2 so as to minimize the sum of the squares of the errors. Figure 2.2 shows these squares graphically. We twist and push the line through these points until the sum of the areas of the squares is the smallest. You may ask "Why the squares? Why not just get the smallest sum of absolute values of error?" Strangely, it is easy to compute the b 's to minimize the sum of squares, but quite difficult to minimize the sum of absolute errors, and it is really this ease that is responsible for our preference for the squares.

Fig. 2.2: Fitting by Least Squares



2. How to Calculate the Least Squares Regression Coefficients

To find the "least squares" values of the b 's, we just minimize with respect to b_1 and b_2 the following summation:

$$(5) \quad S = \sum_{t=1}^T (y_t - (b_1 x_{t1} + b_2 x_{t2}))^2$$

where T is the number of observations which we have, $x_{t1} = 1$ for all t and $x_{t2} = \text{income in period } t$. The sigma, Σ , of course, indicates summation, and the $t=1$ below it indicates that the summation begins with that value of t and extends to $t = T$, shown above the sigma. Now if S is to be minimal with respect to both b_1 and b_2 , then it must be minimal with respect to b_1 with b_2 held constant. The reader who has not studied calculus must now accept on faith that the solution to equations (8) below

will minimize this S . But anyone who knows that the derivative of x^2 is $2x$ can follow the derivation. First, the derivative of S with respect to b_1 (with b_2 held constant) must be zero:

$$(6) \quad \frac{\partial S}{\partial b_1} = \sum_{t=1}^T 2 (y_t - (b_1 x_{t1} + b_2 x_{t2}))(-x_{t1}) = 0$$

Likewise, S must be minimal with respect to b_2 , so

$$(7) \quad \frac{\partial S}{\partial b_2} = \sum_{t=1}^T 2 (y_t - (b_1 x_{t1} + b_2 x_{t2}))(-x_{t2}) = 0$$

We may see the implications of (6) and (7) more clearly if we will divide through by 2, move the terms not involving b_1 and b_2 to the right hand side, and factor out b_1 and b_2 from the sums involving them. Upon so doing, (6) and (7) become

$$(8) \quad \begin{aligned} b_1 \sum x_{t1}x_{t1} + b_2 \sum x_{t1}x_{t2} &= \sum x_{t1}y_t \\ b_1 \sum x_{t2}x_{t1} + b_2 \sum x_{t2}x_{t2} &= \sum x_{t2}y_t. \end{aligned}$$

(Here we have dropped the limits on the summation simply because they are always the same.) Now bear clearly in mind that the x 's and y 's are known, historically observed values. Hence, all the sums are known numbers. The unknowns are b_1 and b_2 . So we see that we have two linear equations in two unknowns. You solved equations like that in high school; if perhaps you have forgotten how, I shall remind you in a moment. Given that we know how to solve linear equations, we can consider our problem of how to find the b 's to be solved.

So far, you may say, we have only dealt with the two-variable case, handled satisfactorily graphically. What about more variables? Well, we might as well go for the general case and consider n independent variables x_1, \dots, x_n , and the equation

$$y_t = b_1 x_{t1} + \dots + b_n x_{tn}.$$

Then let

$$S = \sum_{t=1}^T (y_t - (b_1 x_{t1} + \dots + b_n x_{tn}))^2$$

Differentiating with respect to b_1, b_2, \dots, b_n gives

$$(9) \quad \begin{aligned} \frac{\partial S}{\partial b_1} &= \sum_{t=1}^T 2 (y_t - (b_1 x_{t1} + \dots + b_n x_{tn}))(-x_{t1}) = 0 \\ \frac{\partial S}{\partial b_n} &= \sum_{t=1}^T 2 (y_t - (b_1 x_{t1} + \dots + b_n x_{tn}))(-x_{tn}) = 0 \end{aligned}$$

These may be rewritten as

$$\begin{aligned} b_1 \sum x_{t1}x_{t1} + b_2 \sum x_{t1}x_{t2} + \dots + b_n \sum x_{t1}x_{tn} &= \sum x_{t1}y_t \\ b_1 \sum x_{t2}x_{t1} + b_2 \sum x_{t2}x_{t2} + \dots + b_n \sum x_{t2}x_{tn} &= \sum x_{t2}y_t \end{aligned}$$

$$(10) \quad \dots\dots\dots$$

$$b_1 \sum x_{t1}x_{t1} + b_2 \sum x_{t1}x_{t2} + \dots + b_n \sum x_{t1}x_{tn} = \sum x_{t1}y_t.$$

Do you see the system? In the first equation, the first factor in every product is x_{t1} , in the second equation, x_{t2} , and so on. In the i th column, the second factor in each sum is x_{ti} .

Let us take a simple example with $T = 5$.

	t	x_{t1}	x_{t2}	x_{t3}	y_t
	1	1	10	5	17
	2	1	5	1	10
(11)	3	1	0	6	12
	4	1	10	3	16
	5	1	0	10	20

You should now verify that the equations (10) are

$$5b_1 + 25b_2 + 25b_3 = 75$$

$$25b_1 + 225b_2 + 85b_3 = 380$$

$$25b_1 + 85b_2 + 171b_3 = 415$$

Table 2.1 shows how to solve them systematically.

Line	b_1	b_2	$b_3 = 1$	Derivation	
L1	<u>5.</u>	25.	25.	75.	
L2	25.	225.	85.	380.	Original Equations
L3	25.	85.	171.	415.	

L4#	1.	5.	5.	15.	L1/5
L5	0.	<u>100.</u>	-40.	5.	L2 -25*L4
L6	0.	-40.	46.	40.	L3 - 25*L4

L7	1.	0.	7.	14.75	L4 - 5*L8
L8#	0.	1.	-0.4	.05	L5/100
L9	0.	0.	<u>30.</u>	42.	L6 -(-40)*L8

L10	1.	0.	0.	4.95	L7 -7*L12
L11	0.	1.	0.	.61	L8-(-.4)*L12
L12#	0.	0.	1.	1.40	L9/30

A # after a line number marks the line computed first in each panel of three lines.

Table 2.1: Least Squares Computations

The first three lines (L1, L2, L3) show the original equations. (The line across the top shows what the numbers in each column would be multiplied by before adding together to give the equations.) The next three show the results of eliminating b_1 from the second and third equations. To obtain them, first get the coefficient of b_1 in the first equation to be 1.0 by dividing the equation by b_1 's coefficient, namely 5. The divisor is called the pivot element and is underlined in the table. The result is L4. Now to eliminate b_1 from equation 2, multiply L4 by b_1 's coefficient in L2 and subtract from L2 to get L5. L6 is similarly calculated. We have now completed one pivot operation and eliminated b_1 from all equations except the first. In the next three lines, L7, L8, and L9, we similarly get 0 for the coefficient on b_2 in all but the second equation. Finally, in L10 - L12, we get zero coefficients on b_3 in all but the third equation. From these last three lines, we read the answer

$$b_1 = 4.95 \quad b_2 = .61 \quad b_3 = 1.40$$

The process we have been engaged in is known as linear regression and the b's we have found are the regression coefficients. The particular method of solution of the equations is known as Gauss-Jordan reduction. The idea of fitting the equation by minimizing the sum of squared misses first appears in Adrien Marie Legendre's *Nouvelles méthodes pour la détermination des orbites des comètes* in 1805. In 1809, Carl Friedrich Gauss published a small volume in Latin on the orbits of planets in which he not only used the method but claimed to have been using it since 1795. Given Gauss's many achievements in mathematics, there is no particular reason to doubt his claim, but he might have forgotten to ever describe it had not Legendre done so first.

Exercise

2.1 Calculate the regression equation for the following data.

x1	x2	x3	y
1	5	2	14
1	4	3	13
1	6	3	17
1	7	5	20
1	6	7	19
1	8	6	21

3. Some Measures of How Well the Equation Fits

Now that we have computed the regression coefficients, we may well ask, How well does the equation fit the data? To answer, we first need to compute the values of the dependent variable "predicted" by the equation. These predicted values are denoted by \hat{y}_t thus

$$\hat{y}_t = \sum_{i=1}^n b_i x_{ti}$$

They are shown in the third column of Table 2.2, where the actual values are shown in the second column. The misses, or "residuals",

$$r_t = \hat{y}_t - y_t$$

are shown in the fourth column. Note that the sum of the residuals is zero; it will always be zero if there is a constant term in the equation. Since we were trying to minimize the sum of squares of these residuals, this quantity is naturally of interest:

$$S = \sum_{t=1}^T r_t^2$$

Actually, the Standard Error of Estimate,

$$SEE = \sqrt{S/T}$$

is easier to interpret for it has the same units as the dependent variable. Indeed, we could describe it as sort of average error. (Many texts define SEE with $T-n$ instead of T in the denominator. The *G7* program uses the above definition as a descriptive statistic in its own right, not an estimate -- under rather strong assumptions -- of some unobservable population parameter.)

t	y		r	r ²	d	d ²	f	f ²	%
1	17.	18.05	1.05	1.1025	2.0	4.	-1.65	2.7225	6.18
2	10.	9.40	-0.60	0.3600	-5.0	25.	1.95	3.8025	6.00
3	12.	13.35	1.35	1.8225	-3.0	9.	-2.10	4.4100	11.25
4	16.	15.25	-0.75	0.5625	1.0	1.	-0.30	0.0900	4.69
5	20.	18.95	-1.05	1.1025	5.0	25.	0.00	0.0000	5.25
			-----	-----	---	---		-----	----
			0.00	4.9500	0.0	64.		11.0250	33.37

$$SEE = \sqrt{4.95/5} = .995 \quad MAPE = 33.37/5 = 6.67$$

$$R^2 = 1 - (4.95/64) = .9227 \quad RBARSQ = 1 - (4.95/2) / (64/4) = .8453$$

$$DW = 11.025/4.95 = 2.2272 \quad RHO = (2 - 2.2272)/2 = -.1136$$

Table 2.2: The Fit of the Equation

Another measure of closeness of fit is the ratio of S , the sum of squared residuals, to D , the sum of the squared deviations from the mean of the dependent variable, \bar{y} . These deviations are shown in the column labeled "d" in Table 2.2. This D is, to be explicit,

$$D = \sum_{t=1}^T (y_t - \bar{y}_t)^2$$

The ratio S/D would be zero for a perfect fit and 1.0 in the worst possible case (provided there is a constant term in the equation.) Since it seems a bit strange to have zero as the perfect score, the S/D ratio is subtracted from 1.0 to form what is called the "coefficient of multiple determination" or "R square" or RSQ for short,

$$R^2 = 1 - S/D$$

A "corrected" R^2 , written with a bar over the R , and pronounced "R bar square" is often used to allow for the fact that as more and more independent variables are added to a regression, S goes to zero. In fact, if there are as many independent variables as there are observations, the fit will normally be perfect. For the corrected R^2 , therefore, S is divided by $T - n$, the number of observations less the number of independent variables. The formula is

$$\bar{R}^2 = 1 - \frac{S/(T-n)}{DI/(T-1)}$$

where n is the number of independent variables, counting the constant term. Where it is not convenient to use superscripts, we call this measure *RBARSQ*. The number $T - n$ is called the number of "degrees of freedom" in the regression.

An alternative measure of closeness of fit is the mean absolute percentage error, or *MAPE*, defined by

$$MAPE = \frac{100 * \sum_{t=1}^T \left| \frac{r_t}{y_t} \right|}{T}$$

Its calculation is illustrated in the last column of Table 2.2.

Although *RSQ*, *RBARSQ*, and *MAPE* are all dimensionless pure numbers, there is no absolute standard of "how good is good" for them. Generally, any equation whose dependent variable has a strong trend will have a high *RSQ* while equations with volatile, untrended dependent variables will have low *RSQ*. These measures are really useful only for comparing the fit of one equation for a particular variable with another equation for the same variable.

The columns of Table 2 labeled f and f^2 are explained in the next section.

Exercise

2.2 For the data in exercise 2.1, calculate also the *SEE*, *RSQ*, *RBARSQ*, and *MAPE*.

4. Measures of Autocorrelation and their Uses.

In addition to asking how well the equation fits, it is also usual to ask, To what extent does this equation tend to "go on making the same mistake"? That is, if it erred on the high side in one period is it likely to err on the high side in the next period? If so, the residuals are said to be autocorrelated. The Durbin-Watson statistic, *DW*, and *RHO* are both measures of autocorrelation of the residuals

$$r_t = \hat{y}_t - y_t$$

The Durbin-Watson statistic is

$$DW = \sum_{t=1}^{T-1} (r_{t+1} - r_t)^2 / S$$

and

$$RHO = (2 - DW)/2$$

is the autocorrelation coefficient, ρ . RHO is approximately the regression coefficient, ρ' , of the equation

$$r_{t+1} = \rho' r_t$$

The formula for this regression coefficient is

$$\rho' = \frac{\sum_{t=1}^{T-1} r_{t+1} r_t}{\sum_{t=1}^{T-1} r_t r_t}$$

and

$$DW = \sum r_{t+1}^2 + \sum r_t^2 - 2 \sum r_{t+1} r_t$$

where all of the summations go from 1 to $T - 1$. For large values of T , the first two summations on the right and the summation in the denominator of ρ' differ insignificantly from S , so that -- as a close approximation --

$$DW = 2 - 2\rho' \quad \text{or} \quad \rho' = (2 - DW)/2.$$

The computation of DW and RHO for the example is shown in Table 2.2 in the columns labeled f and f^2 and at the bottom of the table. Here $f_t = r_{t+1} - r_t$. The ideal values are 2.0 for DW and 0.0 for RHO ; but, as before, what is a good value of RHO depends on the variables in the equations. For some variables, it is easy to get RHO under .1, while for others a RHO of .5 may be quite an accomplishment. Some ailments caused by high values of RHO and some possible treatments for them are discussed in Chapter 9 in the section on the Hildreth-Lu technique. The definition of the Durbin-Watson statistic is included here because the statistic is reported by the $G7$ regression program. $G7$ includes it as a concession to the mores of the econometric profession. It is used for testing the hypothesis that there is no autocorrelation. The tests rest on assumptions that are rarely valid in building economic models and need not detain us here.

Knowing the value of RHO can be quite helpful when forecasting with an equation. Suppose, for instance, that we have data through the last quarter of 2006 and want to forecast the first quarter of 2007. We know, moreover, that RHO is .75 and that the equation predicted a value of 326 for 2006 fourth quarter, when the actual value was only 310. The residual was, therefore, 16. Now, for the first quarter of 2007, the equation is predicting a value of 340. You can accept the prediction at face value, or you can say to yourself, "I know this equation; if it has predicted too high in one period, it

is likely to predict too high in the next. In fact, the best guess for that error is *RHO* times the last previous error. So I am going to assume that it is now missing on the high side by $.75*16 = 12$, and my forecast is going to be $340 - 12 = 328$. And if I had to forecast the second quarter of 2007, I would subtract $.75*12 = 9$ from whatever the equation predicts, and so on into the future." Forecasts which incorporate such adjustments are called "RHO-adjusted" forecasts. We will denote the standard error of estimate of such forecasts for one period ahead by "SEE+1".

Exercise

2.3 For the data of exercise 2.1, calculate also *DW* and *RHO*. Suppose that you know the independent variables for two periods further into the future, namely:

x1	x2	x3
1	9	7
1	10	9

Calculate forecasts for these two periods with and without the RHO adjustment.

5. Statistics for Each Independent Variable

We have now covered the overall characteristics of a regression equation which are normally displayed by *G7*, the regression program which goes with this book. You have seen three of these displays in connection with Model 5 in Chapter 1.

For each individual variable, however, you will see that there are displayed six items labeled: Reg-Coeff, Mexval, Elas, NorRes, Mean, and Beta. The "Reg-coef" is, of course, just the regression coefficient, the *b* for each variable, which we have just learned to compute. The "Mean" is just the mean of the variable over the sample period. The "Elas" is the elasticity calculated at the means. This elasticity, the percentage by which the dependent variable will change if an independent variable changes by one percent, is often very useful for examining the economic content of the regression coefficients. In general, the elasticity of *y* with respect to *x* is

$$elas = (\delta y/y)/(\delta x/x) = (\delta y/\delta x)(x/y),$$

where δ indicates "change in". In a linear equation, such as

$$y = a + bx,$$

$(\delta y/\delta x) = b$, so $elas = b(x/y)$ depends upon the values of *x* and *y* at which it is evaluated. It is customary to evaluate at the means. In the numeric example we worked above, the elasticity of *y* with respect to x_2 is $.61*5/15 = .203$. What is the elasticity of *y* with respect to x_3 ?

The Mexval, or marginal explanatory value, of each variable is simply the percentage by which the SEE of the equation would increase if the variable were omitted from the equation and the coefficients of the remaining variables were adjusted to give the best possible fit without that variable. It helps us to see how important each variable is for the fit of the whole equation. Its calculation will be explained in section 8 of this chapter.

The Beta coefficient shows what the regression coefficient would be if the units of both the dependent and independent variables were changed so that the variance of each variable was 1.0. (The variance is the sum of squares of the deviations from the mean divided by the number of observations.) It is useful for judging the sensitivity of the dependent variable with respect to each independent variable. For example, in the equation for fixed investment in Chapter 1, the beta coefficient for the “replace” variable was .781, a coefficient which represents substantial sensitivity. If “replace” moves by one of its standard deviations, investment will move by about .8 of its standard deviations.

Many regression programs automatically give Student's t -statistics for each variable. It is intended for evaluating the “statistical significance” of the dependent variable for the equation. However, the estimate of the standard deviation involves several highly questionable assumptions, and the application of the Student t test to the variable requires even further dubious assumptions. For ordinary, everyday use, the simple, descriptive Mexval is, in my view, far preferable to the T-stat whose meaning depends on many dubious assumptions. If, however, you wish to see the t -values, give the command “showt yes”. To return *G7* to its normal output display, the command is “showt no”. Like Mexval, the calculation of T-Stat will be explained in the final section of this chapter. Activation of the “show t” option in this way also causes F statistics to be printed for each variable. These F values would be used for testing the combined statistical significance of that variable and all the following variables in the regression.

The NorRes column shows how the sum of squared residuals was gradually reduced as variables were added to the regression. The entry on row k shows the sum of squared residuals when the first k variables had been introduced divided by the corresponding sum when all variables had been introduced.

Exercise

2.4 . With the data of exercise 2.1, compute the elasticity of the dependent variable with respect to x_2 and x_3 at the means of all variables.

6. Matrix Notation for Regression

The calculations described in the previous section can all be expressed very neatly by a shorthand known as matrix notation. Without it, some of the things to be covered in this book just cannot be explained. Since it is easy to learn, and will smooth the way for the discussion in the third section of this chapter, we may as well take a moment to introduce it right here. Perhaps you are already acquainted with it; if so, just glance at equation (10m) and then skip to the last paragraph of the section.

To introduce matrices, let us consider again the equations we just solved. They were

$$\begin{aligned}5b_1 + 25b_2 + 25b_3 &= 75 \\25b_1 + 225b_2 + 85b_3 &= 380 \\25b_1 + 85b_2 + 171b_3 &= 415.\end{aligned}$$

We could economize on b 's and $+$ signs if we would write them instead as

$$\begin{pmatrix} 5 & 25 & 25 \\ 25 & 225 & 85 \\ 25 & 85 & 171 \end{pmatrix} \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix} = \begin{pmatrix} 75 \\ 380 \\ 415 \end{pmatrix}$$

We have changed nothing by this rewriting; we have just introduced a shorthand. Now let us think of the array of numbers on the left as a single entity and call it A . This A is a matrix. In fact, any rectangular array of numbers is a matrix. The column of numbers on the right side is then also a matrix, but since it has only one column it may also be called a "vector". Because it is a vector, we will denote it with a lower case letter and call it c . The column of the unknown b 's is also a vector, and we may as well call it b . Then the whole equation can be written as just

$$Ab = c.$$

In this equation, we say that the matrix A is "post multiplied" by the column b . What that means is fully explained by looking back at the original equations. Here are a couple of examples to check your understanding of multiplying a matrix by a vector:

$$\begin{pmatrix} 3 & 7 \\ 5 & 2 \\ 6 & 1 \end{pmatrix} \cdot \begin{pmatrix} 3 \\ 2 \end{pmatrix} = \begin{pmatrix} 23 \\ 19 \\ 20 \end{pmatrix} \qquad \begin{pmatrix} 3 & 1 & 2 \\ 2 & 0 & 8 \\ 3 & 6 & 4 \\ 4 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} 2 \\ 1 \\ 5 \end{pmatrix} = \begin{pmatrix} 17 \\ 44 \\ 32 \\ 9 \end{pmatrix}$$

Note that in order to post multiply the matrix A by the column vector b , b must have the same number of rows that A has columns. The result will be a column with as many rows as A has rows. The number of rows and columns of a matrix are called its dimensions. The matrix on the left in the first example above is said to be "3 by 2", that is it has 3 rows and 2 columns. In general, we write " A is (m,n)" when we mean that A has m rows and n columns.

Now suppose that we have two matrices, A and B . If they are of the same dimensions, we can define their sum, $A + B$, as the matrix composed of the sums of the corresponding elements in the two matrices. For example,

$$\begin{pmatrix} 3 & 5 \\ 4 & 3 \end{pmatrix} + \begin{pmatrix} 2 & 1 \\ 3 & 5 \end{pmatrix} = \begin{pmatrix} 5 & 6 \\ 7 & 8 \end{pmatrix}$$

When it comes to the product of two matrices AB , however, we do not define it as the matrix composed of the products of corresponding elements. Rather we define it as the matrix whose first column is the product of A post-multiplied by the first column of B , and whose second column is the product of A post-multiplied by the second column of B , and so on. Here are two examples:

$$\begin{pmatrix} 3 & 5 \\ 4 & 3 \end{pmatrix} \cdot \begin{pmatrix} 2 & 1 \\ 3 & 5 \end{pmatrix} = \begin{pmatrix} 21 & 28 \\ 17 & 19 \end{pmatrix}$$

$$\begin{pmatrix} 3 & 2 & 1 \\ 2 & 5 & 3 \end{pmatrix} \cdot \begin{pmatrix} 5 & 2 \\ 1 & 3 \\ 2 & 1 \end{pmatrix} = \begin{pmatrix} 19 & 13 \\ 21 & 22 \end{pmatrix}$$

In order for the product AB to be defined, B must have as many rows as A has columns. The product will have as many rows as does A and as many columns as does B . In general AB will not be the same as BA , although there are important special cases in which they are the same.

It is easy to verify that $(A + B) + C = A + (B + C)$ --the order in which we add matrices makes no difference -- and that $(AB)C = A(BC)$ -- the order in which we multiply makes no difference. Also, as with ordinary numbers, multiplication is distributive over addition: $A(B + C) = AB + AC$.

To express our least-squares problem in terms of matrices, we need just one more concept, the transpose of a matrix. The transpose of A , denoted by A' , is the matrix whose first column is the first row of A , whose second column is the second row of A , and so on. For example, if A is the matrix given by

$$A = \begin{pmatrix} 5 & 2 \\ 1 & 3 \\ 2 & 1 \end{pmatrix}$$

then the transpose is

$$A' = \begin{pmatrix} 5 & 1 & 2 \\ 2 & 3 & 1 \end{pmatrix}$$

If $A = A'$, then the matrix must be square and is said to be symmetric. You can, with a little thought,

see that $(AB)' = B'A'$.

Now let us use matrix notation to write in compact form the equations of section 2 of this chapter. We will denote by X the matrix of observations on the independent variables and by y the vector of observations on the dependent variable in our previous regression example. Thus,

$$X = \begin{pmatrix} 1 & 10 & 5 \\ 1 & 5 & 1 \\ 1 & 0 & 6 \\ 1 & 10 & 3 \\ 1 & 0 & 10 \end{pmatrix}, \quad y = \begin{pmatrix} 17 \\ 10 \\ 12 \\ 16 \\ 20 \end{pmatrix}$$

Finally let b denote the vector of regression coefficients. We want to choose b to minimize S , the sum of squared residuals. In matrix notation, S is

$$(9m) \quad S = (y - Xb)'(y - Xb).$$

This equation has been numbered (9m) because it is the matrix way of writing equation (9) of section 2. The minimizing b vector is given by the equation

$$(10m) \quad (X'X)b = X'y.$$

It may take a moment to see that this equation (10m) is the same as the previous equation 10. But just write out X' and X and start forming the product $X'X$, and you will soon realize that you are forming the matrix A with which we began this section. Note that $X'X$ is symmetric.

How can we show the process of solution of linear equations in matrix notation? To do so we first need a special notation for any square matrix that has 1's on the diagonal running from top left to bottom right and is otherwise zero. Such a matrix is called an "identity" matrix and is therefore denoted by I . Note that for any matrix M , $IM = M$ and $MI = M$, where I is an identity matrix of appropriate dimension. Suppose now that we have the matrix equations $Ax = c$, where A is a square matrix and x and c are vectors. If we can find some matrix B such that $AB = I$, then $A(Bc) = (AB)c = Ic = c$, so $x = Bc$ is the solution of the equations $Ax = c$.

But how do we find the matrix B ? Since $AB_1 = I_1$, $AB_2 = I_2$, and $AB_3 = I_3$, where the subscript denotes a column of B or I , we could just solve these equations one-by-one for the columns of B . There is, however, an easier way; we can solve them all at once. Table 2.3 shows how. In the first three rows and first four columns of each panel you will recognize the corresponding panel of Table 2.1. In the fifth column we have written the right-hand side of the equations $AB_1 = I_1$.

5.	25.	25.	75.	1.	0.	0.
25.	225.	85.	380.	0.	1.	0.
25.	85.	171.	415.	0.	0.	1.
75.	380.	415.	1189.	0.	0.	0.
1.	5.	5.	15.	0.2	0.	0.
0.	100.	-40.	5.	-5.	1.	0.
0.	-40.	46.	40.	-5.	0.	1.
0.	5.	40.	64.	-15.	0.	0.
1.	0.	7.	14.75	0.45	-0.05	0.
0.	1.	-0.4	0.05	-0.05	0.01	0.
0.	0.	30.	42.	-7.0	0.4	1.
0.	0.	42.	63.75	-14.75	-0.05	0.
1.	0.	0.	4.95	2.083	-0.143	-0.233
0.	1.	0.	0.61	-0.143	0.015	0.013
0.	0.	1.	1.4	-0.233	0.013	0.033
0.	0.	0.	4.95	-4.950	-0.610	-1.400

Table 2.3: Regression with Matrix Inversion

Now notice that if we carry the pivot operations through this column as well as the first four we will have in it in the fourth panel the solution of the equation $AB_1 = I_1$. Note that appending this extra column had absolutely no effect on the previous four columns. Nor did the fourth column, the original right-hand side of the equations, have any effect on what happened in the fifth column. We can, therefore, append the other columns of I as the last two columns of the first panel in Table 2.3, carry through the pivoting on them, and get in the top three rows of the last panel the matrix B . We have gotten the solution for three sets of equations for considerably less than three times the work required for one.

The matrix B which we have found in this way is called the "inverse" of A and is denoted by A^{-1} . From its construction, we know that $AA^{-1} = I$. It is also true that $A^{-1}A = I$, for if we start from the last panel of Table 2.3 and think of it as representing the equations $A^{-1}C = I$ where A^{-1} is known and C is unknown, we can solve these equations for C by retracing our steps one-by-one back up Table 2.3 and discover that $C = A$. So $A^{-1}A = I$. Thus, every right inverse is also a left inverse.

We will also often use the fact that the inverse of a symmetric matrix is itself symmetric. If $A = A'$, then

$$I = (AA^{-1})' = (A^{-1})'A' = (A^{-1})'A$$

and post multiplying both sides of this equation by A^{-1} gives

$$A^{-1} = (A^{-1})',$$

which is to say, A^{-1} is symmetric.

We can now summarize our discussion so far in one equation: the regression coefficients are given by the equation

$$(12) \quad \mathbf{b} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}.$$

For future reference, we need one more concept, the trace of a square matrix. The trace is simply the sum of the diagonal elements. Thus, if \mathbf{C} is (m,m) , then the trace of \mathbf{C} , $\text{tr } \mathbf{C}$, is just

$$\text{tr } \mathbf{C} = \sum_{i=1}^m c_{ii}$$

If \mathbf{A} is (m,n) and \mathbf{B} is (n,m) then \mathbf{AB} and \mathbf{BA} are both defined but are not generally equal to one another or even of the same dimension. But both are square so that $\text{tr}(\mathbf{AB})$ and $\text{tr}(\mathbf{BA})$ are both defined. Now it is a remarkable and useful theorem that $\text{tr}(\mathbf{AB}) = \text{tr}(\mathbf{BA})$. To see why this is true, just note that any element of \mathbf{A} , say a_{13} , will enter into the sum that forms $\text{tr}(\mathbf{AB})$ exactly once and that one time it will be multiplied by the symmetrically placed element of \mathbf{B} , b_{31} . Now notice that the element also enters $\text{tr}(\mathbf{BA})$ exactly once and is again multiplied by the same element of \mathbf{B} . Thus the sums that form $\text{tr}(\mathbf{AB})$ and $\text{tr}(\mathbf{BA})$ consist of exactly the same elements and must therefore be equal to one another. Working a small example will make this proof plain.

Exercise

2.5 For the data of exercise 2.1, compute $(\mathbf{X}'\mathbf{X})^{-1}$, then $\mathbf{X}'\mathbf{y}$, and then, by matrix multiplication, $\mathbf{b} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}$.

7. A Shortcut to the Sum of Squared Residuals

The last row of each panel of Table 2.3 provides a short-cut to the calculation of S , the sum of squared residuals. We have not yet explained this row. In the top panel, it contains the row vector $(\mathbf{y}'\mathbf{X}, \mathbf{y}'\mathbf{y})$, which is the transpose of the fourth column, so the first four rows and columns form a symmetric matrix. As we generate the successive panels, we carry through the pivot operations on the last row just as on the other rows.

Now let me point out a surprising "coincidence". In the fourth panel of Table 2.3 we find, in the position where $\mathbf{y}'\mathbf{y}$ was originally, the number 4.95. This is exactly the value that we got for S by a totally different procedure in Table 2.2. Can it be that we can find the sum of squares by pivoting instead of by calculating all of the residuals, squaring them and summing? Yes, that is true. And it is useful in many ways. Let us see why it is so.

By the time we reach the last panel in Table 2.3, we will have subtracted from the original last row

some combination of the rows above it and gotten, in the first three positions, zeroes. What combination of the rows above it did we subtract? Since we originally had in those positions $y'X$ and, after subtracting we have 0, we must have subtracted a combination, given by the row vector c , such that $c(X'X) = y'X$. In fact, this c is really just b' , the transpose of the vector of regression coefficients, for

$$(X'X)b = X'y$$

so

$$b'(X'X)' = y'X$$

and

$$b'(X'X) = y'X$$

since $(X'X)' = X'X$. Therefore what has been subtracted from the final position of this last row is $b'X'y$. What was originally in it was $y'y$, so what is left is $y'y - b'(X'y)$. The direct approach to calculating S first calculates

$$r = y - Xb = y - X(X'X)^{-1}X'y$$

and then forms

$$\begin{aligned} S &= r'r \\ &= y'y - y'X(X'X)^{-1}X'y - y'X(X'X)^{-1}X'y + \\ &\quad y'X(X'X)^{-1}X'X(X'X)^{-1}X'y \\ &= y'y - y'X(X'X)^{-1}X'y = y'y - b'X'y, \end{aligned}$$

which is exactly what the pivoting gave.

Now suppose for a moment that we had set out to regress x_2 on x_1 . We would have formed exactly the same 2-by-2 matrix that we see in the upper left corner of panel 1 of Table 2.3 and the final result would have been the 2-by-2 in the upper left corner of panel 2. The value of S for this problem would have been 100 and the regression coefficient would have been 5. Similarly, if x_3 had been regressed on x_1 , the value of S would have been 46 and regression coefficient 5. (Because x_1 is the constant 1, the regression coefficients are the means, and the values of S are the sum of squared deviations from the means.) *In general, we see that in panel $i+1$, after i pivot operations, we can see the regression coefficients and values of S for the regression on the first i variables of each of the remaining variables.* Thus, the regression panels show a great deal about the relations among the variables. Can you give an interpretation for the element in the third row and fourth column of the third panel (the number is 42)?

For computing the NorRes (Normalized sum of squared Residuals) coefficients shown in the standard $G7$ display of results, it is necessary only to store away the value of S , the value in the lower right corner of the array, at each iteration. These values are then all divided by the last of them to produce the NorRes display.

Please note that each pivot element was the S value for the variable about to be introduced when regressed on all the previous values. If a pivot element is zero, the regression cannot continue; but a zero pivot can occur only if the variable about to be introduced was perfectly explained by the previous variables. If this happens when, say the third variable is about to be introduced, the $G7$ program will give the message "Variable 3 is a linear combination of preceding variables," and will abort that particular regression.

A moment's study of Table 2.3 shows that in each panel three of the columns are the columns of the identity matrix. In working with a computer, these identity columns just waste space, and it is usual to store only the non-identity columns of the matrix, as shown below in Table 2.4. I will refer to this sort of table as regression with compact inversion.

Exercise

2.6 Extend your previous computation with the data of exercise 2.1 to include the "dependent variable" row in each computation. What is S when only x_1 is used as an explanatory variable? When only x_1 and x_2 are used? When all three are used? What are the regression coefficients for x_3 regressed on x_1 and x_2 ?

8. Mexvals and Derivatives -- Measures of the Importance of Each Variable

So far, we have not developed a way to say anything about how important any particular variable is to the whole equation. One measure designed to help in answering this question is the "mexval" of a variable. A variable's mexval, or marginal explanatory value, is defined as the percentage that *SEE* will increase if the variable is omitted from the regression and not replaced by any other, though the coefficients on the remaining variables are adjusted to do as well as possible without their departed comrade.

5	25	25	75
25	225	85	380
25	85	171	415
75	380	415	1189
0.2	5	5	15
-5.0	100	-40	5
-5.0	-40	46	40
-15.0	5	40	64
0.45	-0.05	7.0	14.75
-0.05	0.01	-0.4	0.05
-7.0	0.4	30.0	42.0
-14.75	-0.05	42.0	63.75
2.083	-0.143	-0.233	4.95
-0.143	0.015	0.013	0.61
-0.233	0.013	0.033	1.40
-4.950	-0.610	-1.400	4.95

Figure 2.4: Regression with Compact Inversion

Mexval is easily calculated in the process of regression with compact inversion. With n independent variables, this form of regression leads to a final panel like this:

$$\begin{array}{cccc}
 a_{11} & \dots & a_{1n} & a_{1m} \\
 \dots & \dots & \dots & \dots \\
 a_{n1} & \dots & a_{nn} & a_{nm} \\
 a_{m1} & \dots & a_{mn} & a_{mm}
 \end{array}$$

where $m = n + 1$. Remember that the lower right element is the sum of squared residuals. Note also that the values of the a 's do not depend upon the order in which the row reduction was done, that is, in regression terms, they do not depend on the order of introduction of the variables. Let us suppose that variable i was the last to be introduced and let us denote the elements of the panel before its introduction with a' . Then we have the following relation between the a and a' :

$$a_{ii} = 1/a'_{ii} \qquad \text{and} \qquad a'_{ii} = 1/a_{ii}$$

$$a_{im} = a'_{im}/a'_{ii} \quad \text{and} \quad a'_{im} = a_{im}/a_{ii}$$

$$a_{mm} = a'_{mm} - a'_{im}a'_{im}/a'_{ii} \quad \text{and} \quad a'_{mm} = a_{mm} + a_{im}a_{im}/a_{ii}$$

Thus, the drop in the sum of squares of the residuals as variable i was introduced -- and the increase in that sum of squares if it is now excluded from the equation -- is a_{im}^2/a_{ii} .

The standard error of estimate of the equation would therefore rise from

$$SEE = \sqrt{a_{mm}/T}$$

to

$$SEE' = \sqrt{(a_{mm} + a_{im}^2/a_{ii})/T}.$$

where T is the number of observations. Therefore,

$$\begin{aligned} \text{mexval}_i &= 100*((SEE'/SEE)-1) \\ &= 100*(\sqrt{1+(a_{im}^2/a_{ii}a_{mm})}-1). \end{aligned}$$

For the example of the text, we find

$$\begin{aligned} \text{mexval}_1 &= 100*(\sqrt{1 + (4.95^2/(2.083*4.95))}) - 1 = 83.74 \\ \text{mexval}_2 &= 100*(\sqrt{1 + (.61^2/(0.015*4.95))}) - 1 = 143.0 \\ \text{mexval}_3 &= 100*(\sqrt{1 + (1.40^2/(0.033*4.95))}) - 1 = 258.9 . \end{aligned}$$

(The numbers at the right are calculated from a more precise inverse than that shown above.)

The insights of this section and the last can be combined to calculate the derivatives of regression coefficients with respect to one another. Suppose that we were to decide that we did not trust the value provided by regression for the coefficient of some variable and that we wanted to fix the value of that coefficient. What effect would that fixing have on the other coefficients? More precisely, what would be the derivatives of the other coefficients with respect to that coefficient, if the others are determined to minimize the sum of squared errors, given the value of the fixed coefficient? Suppose the original equation was written $\mathbf{y} = \mathbf{X}\mathbf{b} + \mathbf{Z}\mathbf{c} + \mathbf{r}$, where \mathbf{Z} is a T -by-1 vector and c is a scalar. We are asking, What is the derivative of the least-squares value of \mathbf{b} with respect to \mathbf{c} ? The least squares estimate of \mathbf{b} , given \mathbf{c} , is $\mathbf{b} = (\mathbf{X}'\mathbf{X})^{-1}(\mathbf{X}'\mathbf{y} - \mathbf{X}'\mathbf{Z}\mathbf{c})$, from which it is clear that the derivative of the vector \mathbf{b} with respect to the scalar c is

$$d\mathbf{b}/dc = -(\mathbf{X}'\mathbf{X})^{-1}(\mathbf{X}'\mathbf{Z}),$$

which is just the negative of the regression coefficients of \mathbf{Z} on all the other variables. Now think of \mathbf{Z} as one more independent variable. This negative of the regression coefficients of \mathbf{Z} on the columns of \mathbf{X} is just the negative of what would have been in \mathbf{Z} 's column before it was pivoted on. To get it back, we just unpivot. In the notation developed in this section, if we want to think of \mathbf{Z} as column

i , we want a'_{ji} . From the pivot operation, we have $a_{ji} = 0 - a'_{ji}/a'_{ii}$ and $a_{ii} = 1/a'_{ii}$. These may be solved for $-a'_{ji} = a_{ji}/a_{ii}$. In other words, to get the derivatives of all of the regression coefficients with respect to coefficient i , we just divide the i th column of $(X'X)^{-1}$ by its diagonal element. These derivatives are very useful for seeing the sensitivity of one coefficient with respect to another. I have often seen it happen that an equation has several coefficients with nonsensical values, but by fixing one of the coefficients to a sensible value, the others also became sensible. The derivatives are useful for spotting such situations.

Exercise

2.7 Compute the mexvals for x_1 , x_2 , and x_3 with the data of exercise 2.1. Compute the derivative of b_1 and b_3 with respect to b_2 .

9. Leverage, a Measure of the Influence of Individual Observations

We mentioned in discussing the fitting of simple regressions by eye that one of the advantages of that method was that it made clear which observations were outliers -- observations which, if included, would have a lot of influence on the regression coefficients. Spotting outliers can be important because the data at the outlier point may be erroneous. One simple indicator of outlier observations is the "leverage" variable, defined, for observation t , simply as the derivative of the predicted value at t , \hat{y}_t , with respect to the observed value at t , y_t . If this derivative is very high for a particular observation, that observation must have a strong influence on one or more of the regression coefficients. Such observations should be checked for the accuracy of the data; if the data is accurate, however, the observation should not be discarded, for it is precisely the observation which, by standing out from the others, reveals what the equation really is.

The leverage variable is easily calculated. The vector of predicted values is

$$\hat{y}_t = Xb = X(X'X)^{-1}X'y$$

so the derivative of the predicted value at t , \hat{y}_t , with respect to the observed value at t , y_t , is simply the t^{th} diagonal element of the matrix $X(X'X)^{-1}X'$. Obviously, if we want only the diagonals of this matrix, we do not have to compute the whole matrix to find them. Rather, we could pre-multiply the first column of X' by $(X'X)^{-1}$ and then pre-multiply this column by the first row of X to get the first element of the leverage vector.

Note that leverage uses only values of X , not of y , and that observations other than the first entered the calculation of the leverage on the first observation only through the $(X'X)^{-1}$ matrix.

The leverage variable is automatically computed and placed in the workspace bank with the name *lever* after each regression in *G7*. It may be graphed or displayed like any other variable.

CHAPTER 3. INTRODUCTION TO G 7.3

From Chapter 1, you are already somewhat familiar with *G7*, the regression and modeling program we will be using throughout this book. It is part of a closely integrated package of programs that can both estimate a single equation or build a multisectoral model involving hundreds of equations. In Chapter 1, I tried to tell you as little about *G7* as I could; I wanted to concentrate on the logical succession of models. Here we will proceed more systematically through the basic functions of *G7*. Please pardon some occasional repetition of what you already know.

1. Tutorial of the Basics

Double click on the *G7* icon to start the program. You will be immediately given a choice of folder (formerly called directory) in which to start *G7*. This step is necessary because *G7* looks for a *g.cfg* file in the folder from which it starts, and this file determines the characteristics of the workspace which *G* creates and the initial assigned data bank. For this tutorial, you should select the *c:\ami* folder. To do so, you should click “Browse” on the screen that appears when you start the program until the *c:\ami* folder comes into view and then *double* click the *ami* folder or (b) single click the *ami* folder and tap the ‘Enter’ key, or (c) enter *c:\ami\g.cfg* in the edit box at the bottom of the window. Whatever route you choose, you should see “*c:\ami*” in this box. Then click the OK button, or tap Alt-O. The folder selection form disappears. Click OK again to proceed.

You then see the *G7* main form. It has a menu at the top, a few speed buttons, a white, one-line *command box*, and a large blue *results* area. Click on the Bank menu and select Look. You will see a list of the assigned banks. Select one of them and click OK. A window opens in the upper right of the screen. It has a list of the data series in the assigned bank. Run the cursor down the list until you find one that interests you. Tap ‘Enter’. Below you will see a graph and to the left, in the results area, the numerical values of the series. (The first number in each line is the date of the first observation in that line; then follow the data for four successive quarterly observations.) The cursor will have gone back to the command box, but if you wish to see another series, click on the “look” window again and continue looking. If you tap a letter key, the cursor will move to the next entry beginning with that letter. To find a word or phrase, like “Personal income”, click on Find in the menu and fill in the box. Note that you can search up or down from where you are in the file.

Now let us suppose that you would like to run a regression of real Gross private fixed investment on past changes in real GDP and an estimate of the investment required to replacement capital that is wearing out. If the cursor is not already in the command box, click there. We first set the limits for the regression, the starting and ending dates. To make the regression go from 1975 first quarter to 2013 third quarter, type

```
lim 1975.1 2013.3
```

and tap ‘Enter’. (If your data bank has data more recent than the third quarter of 2013, by all means use it.) You should see your command echoed in the results area. Then give a title to the regression with something like

```
ti Gross Private Domestic Fixed Investment
```

Next, you must form d , the first difference of real GDP by

```
f d = gdpR - gdpR[1]
```

The remaining variable we need is requirements for replacement, which you may calculate by

```
f replace = .05*@cum(stockf,vfR[4],.05)
```

It is not necessary for the regression, but you may be interested to see how this replacement compares with investment spending. To draw a graph with the two, type

```
gr replace vfR
```

Now you are ready for the regression itself. Just type

```
r vfR = replace, d[1], d[2], d[3], d[4], d[5], d[6], d[7], d[8], d[9], d[10], d[11]
```

There should now appear in the results area the numeric display similar to what you saw for Model 5 in Chapter 1. The font in the results area is set small so that you see most of the regression output. Many people, however, find that it is uncomfortably small. If you want a larger font click File | Auto Font Size, so as to remove the check mark by this item. Then click File | Font and set the size. Personally, I often pick the font called Fixedsys, which cannot be sized but the one available size works nicely. You should, however, stick to a monotype font, that is, one in which all letters are the same width. You may also set the font color and “script” or character set. If you are using titles and comments in a language such as Greek, Russian, Turkish, or Polish with characters that do not appear in the Western European languages, be sure to select the proper character set for the results window. It will be automatically carried over to the editor, the look command, and other displays. By clicking File | Background color, you can select the background color of the results window. Your choice of font (typeface, size, color, and character set) and your choice of background color will be remembered when you next start *G7*. (This generally pleasant feature can be confusing if you are using *G7* on a public machine in a computer lab; you will start off with the settings of the previous user.)

To obtain the graph of the regression just done, give the command

```
gr *
```

To see the graph better, it may be maximized by clicking the maximize button, the middle button in the upper right corner of the graph. Click again the button in the same place to get the graph back to its original size. If you would like to save the graph for comparison with later graphs, select Graph | Shift1 from the main menu. The graph shifts to a smaller window in the upper right of the screen. You can also use Graph | Shift2. Thus, you can view three different graphs at once.

In order to save your work, you should put the commands necessary to do regressions into files and then execute (or *add*) those files. To put them in a file, open the *G7* editor. Click Editor on the main menu and a simple editor opens in the upper right screen. Type into it

```

lim 1975.1 2013.3
ti Gross Private Domestic Fixed Investment
f d = gdpR - gdpR[1]
f replace = .05*@cum(stockf,vfR[4],.05)
r vfR = replace, d[1], d[2], d[3], d[4], d[5], d[6], d[7], d[8], d[9], d[10], d[11]
gr *

```

On the Editor's menu, select File | Save as, and save the file with the name vfR.reg. Then to execute it, simply click Run in the editor menu. (Clicking Run also saves the file to your disk.) Alternatively, if the file you want to run is not already in the editor, you can do File | Execute and in the window which then opens, you may either click the name of the file and then OK, or type in the name and tap 'Enter'. Either way, you should now see the results and be faced with a "dialog" box demanding to know what you want done with the graph. For the moment, click the Continue button.

A third way to execute your vfR.reg file is to enter the command

```
add vfR.reg
```

in the white command box. The results are exactly the same.

Your command file, vfR.reg, is fine so far, but it does not save the results in a way which can be put into a paper or included into a model. Click back on the editor window and modify the file there so that it is

```

addprint y
catch vfR.cat
save vfR.sav
gdates 1975.1 2013.3
lim 1975.1 2013.3
ti Gross Private Domestic Fixed Investment
f d = gdpR - gdpR[1]
f replace = .05*@cum(stockf,vfR[4],.05)
r vfR = replace, d[1], d[2],d[3],d[4],d[5],
  d[6],d[7],d[8],d[9],d[10],d[11]
gr *
save off
gname vfR
gr *
catch off

```

The new material is in bold. Save this file and "run" it again. (You will find that the command box remembers your previous commands. You can get them back either with the down and up arrow keys or by clicking on the arrow at the right end of the command box.) When the program stops and demands to know what to do with the graph, click the Save button.

You will have made three new files:

vfR.sav	The results to be used later by the computer in building a model. This file was made by the "save vfR.sav ... save off" pair of commands.
vfR.cat	The results to be brought into a word processor. This file was made by the "catch vfR.cat ... catch off" pair of commands.
vfR.wmf	The graph in a form to be brought into a word processor. This file was made when you clicked

“Save” on the “Continue add file” dialog box.

You can look at `vfR.sav` with *G7*'s editor. If the aqua green editor window is showing, click on the Open item on its menu and open `vfR.sav` file. Otherwise, first click Editor on the main menu to create an editor window. You will see that the `vfR.sav` file contains the commands to create the variables and the regression results written as an equation.

To appreciate the uses of the other two files, you should open your word processor. You may first minimize *G7* by clicking on the little square in the upper right with the `_` in it. Then open your word processor and import `vfR.cat` into it. In WordPerfect or Word, use the command Insert | File. In OpenOffice Writer, you need to open the file in one window, and then copy and paste it into your document. When you do so, you are apt to find that the columns look wavy. To correct that problem, put the imported text into a monotype font such as Courier or Line printer. To do so, select the text whose font you want to change, click on the button whose caption shows your present font, and select from the list. If you are using fonts that contain a number of different character sets, such as Cyrillic, Greek, West European, East European, you should be sure to select also the “script” (= character set) you desire.

When you clicked “Save” on the “Continue add file” dialog box, you caused the graph to be saved as a Windows metafile, a file which has the instructions for redrawing the file on any device supported by Windows. This file has the name `vfR.wmf`. It may be imported into a word processing program such as WordPerfect, Word or OpenOffice. In WordPerfect, the command is the same as to import text -- Insert | File. Word needs more help from you; the command is Insert | Picture. When it is first imported, the graph may be small; but it can be stretched by dragging with the mouse. Getting graphs to go where you want them when you have a number of graphs on a page is a problem with all present word processors. In WordPerfect, you have a choice of attaching the graph to a page, paragraph, or character. Probably character is your best choice.

You should now close the word processor and return to *G7*.

Let us explore the other options on the “Continue add file” dialog box. To do so, give the command “add `vfR.reg`” again. (By using the up and down arrow keys you can find this command among those already executed and just execute it again.) The program will stop, as before, with the “Continue add file” dialog box demanding to know what to do with the graph. The simplest answer is to click the “Cont.” button to continue the add file, throwing away the graph. Next to it on the right is the “Stop” button, which exits from the command file. Further to the right are the “Max” and “Normal” buttons which blow up the graph to fill the full screen and restore it to its normal size, respectively. Then comes the “Save” button, which, as we know, saves the graph as a Windows metafile with the filename specified by a previous `gname` command. The “Print” button sends the graph directly to the printer. The “Shift1” and “Shift2” buttons move the graph to storage areas 1 and 2 where the graph remains visible as the add file moves on to subsequent commands. A graph in these storage areas can also be resized.

In working with multisectoral models, it is common to have add-files which draw graphs for many sectors. Especially for use with them, “`gname`” can be started with a name and number, for example:

```
gname out 1
```

The name of the save file for the first graph drawn will then be out1.wmf, for the second out2.wmf, and so on. Note that the numbers increase with each graph drawn whether or not it is saved.

If you have a graph showing and would like to save it but are not in an add file and do not have an appropriate name already assigned for the file, use Graph | Save from the main menu. You will be prompted for a file name.

The next step is to use add-files within add-files. Look at the runall.reg with the editor. You will find

```
fdates 1960.1 2013.3
lim 1975.1 2013.3
add vfR.reg
add viR.reg
add mR.reg
```

The “fdates” command sets the range of dates over which the “f” commands work. (The default fdates are the same as the default regression limits in specified in the g.cfg file.) Adding or executing this one file will cause three other files to be executed. Try it. After the first graph has been drawn, you will find that the program stops and demands that you tell it what to do with the graph. The simplest answer is to click the “Cont.” button to continue the add file, throwing away the graph.

At this point, your first results on the blue screen will have scrolled out of sight. But you can use the scroll bars on the blue results window to bring them back into view. You can, by the way, use standard Windows cut-and-paste techniques to move material from this window to a word processor. The result may be somewhat disconcerting at first, for the color of the text (yellow) will move with the text but not the color of the background. If your word processor document has a white background, you will then have yellow text on a white background. In WordPerfect, the reveal codes key allows you to find the color change command and delete it, revealing the text in the normal text color. In Word or OpenOffice, commands in the Format menu control these colors.

As you work, the results window may become so full that tiny movements of the scroll bar produce big changes in what text is shown. You can clear the results window at any time by clicking File | Clear results or by giving the “clear” command in the command box or in an “add” file. The results area is large but not infinite. If you come close to filling it up, the oldest part at the top is automatically deleted.

You will notice also the item File | Auto font size. By clicking it, you can toggle a check mark next to it. If the check mark is visible, then widening or narrowing the results window by dragging its right border with the mouse will cause the size of the font to change so that the basic regression display fills the screen. If you do not want this automatic change in the size of the font, click this item so that the check mark disappears. This feature has no effect if using the FixedSys font, which gives type of pleasant size and clarity.

All of the commands under the Edit item on the main menu apply to the results window.

The Help menu explains the commands available from the command line in G. See especially the help reference section.

2. Digression on Graphs in G

While it is very simple to get a quick graph in *G7*, it is also possible to refine the graph. In order to produce nice looking reports, it is worth digressing from model building for a moment to learn what sorts of graphs you can make.

The dates over which series are graphed is controlled by the “gdates” command. This command may be followed by two or three dates, for example:

```
gdates 1980.1 2013.3
gdates 1980.1 2013.3 2017.4
```

Graphs drawn following the first of these commands will cover the period 1980.1 — 2013.3. Graphs drawn following the second cover the same period but have a vertical line following 2013.3. This vertical line is useful, for example, to divide history from forecast or the period over which a regression was fit from the period over which it was tested. Exception: graphs drawn with the “gr *” command use dates set by the “limits” command for the previous regression.

For greater variety in your graphs, choose Graph | Settings on the main menu. You will get a window for setting the characteristics of up to seven lines on the graphs. You can select the color, width (in pixels), style (solid, dotted, dashed, dash dot, and dash dot dot -- these work only for lines 1 pixel wide), mark, fill, and left and right parameters. The last three apply only if the type “bar” is chosen for the mark. The fill has to do with what sort of hatching appears in bars. The left and right parameters, numbers between 0 and 1, show where, within the space allocated for the bar, the left and right edges of the bar go. For example, with left = .1 and right = .9, the bar will be in the center of the allowed space, and the bars will be four times as wide (.8 units) as the space between them (.2 units). To eliminate the line connecting points marked by bars, set its width to zero. Experiment with what you can do with these options. If you wish to save your graph settings for a future run of G, click the “Save to file” button at the bottom of the screen. You will be asked for a filename. Any valid file name is acceptable, but let us suppose you say “graphs.set”. Then the next time you start G you can give the command “add graphs.set” and have your settings back. You can even add at the bottom of the g.cfg file the line

```
Initial command; add graphs.set
```

and you will automatically have back your settings.

Normally, *G7* automatically adjusts the vertical scale to include all points on any series being graphed, but only the top, bottom and middle of the vertical scale are labeled. You can, however, control the vertical range of the graph with the “vrange” or “vr” command. For example:

```
vr 0 500 1000 1500 2000
```

will cause graphs to be drawn with a range from 0 to 2000 and horizontal lines at 0, 500, 1000, 1500, and 2000. These lines will be labeled. The default is to put the labels inside the box of the graph, but you can put them outside by first giving the vertical axis label command, “vaxl”. It can be

```
vaxl out
```

to put them outside the frame of the graph or

```
vaxl in
```

to put them inside the frame. The “vr” command continues in force until another *vr* command is given. To simply turn it off and go back to the default, use

```
vr off
```

One particularly useful form of the “vr” command is

```
vr 0
```

which sets the lower boundary of the following graphs to be 0, while the top is picked by the program to fit in the highest point in any series. Try graphing *gdpR* with careful vertical axis labeling done with the “vr” and “vaxl” commands.

In addition to the “title” command, which we have used frequently, there is the “subtitle” command, which can be abbreviated to “subti”, which provides a subtitle for the graph. The title and subtitle remain until replaced by another title or subtitle. To remove either of them altogether, just give the command immediately followed by a carriage return. The “legend” command controls whether or not the legend normally at the bottom of the graph is included or not. The format is “legend yes” or “legend no”. Try using the “subtitle” and “legend” commands.

In working with multisectoral models, it is common to have add files which draw graphs for many sectors. Especially for use with them, “gname” can be started with a name and number, thus

```
gname out 1
```

The name of the save file for the first graph drawn will then be *out1.wmf*, for the second *out2.wmf*, and so on. Note that the numbers increase with each graph drawn whether or not it is saved.

Besides the ordinary “gr” command, there are three other variants of it. The “mgr” or “multi graph” command chooses a separate scale for each series graphed. For example, you may want to put the Treasury bill rate, *rtb*, and Residential construction, *vfrR*, on the same graph. But they have totally different scales. The answer is to make the graph with “mgr”, for example, “mgr *rtb vfrR*”. Try it.

Semi-logarithmic graphs are popular because series growing at a constant percent per year appear as straight lines. It is, however, often desirable to label the vertical axis in the original units. This is done by the “lgr” command. For example, to graph the Standard and Poor’s composite index of 500

stocks from 1980.1 to 2011.3, we could do

```
f lsp500 = @log(sp500)
vr 100 200 400 800 1600 3200 4800
gdates 1980.1 2011.3
lgr lsp500
```

Do it and note the straightness of the line and the labeling of the vertical axis in the original units.

Finally, for graphs in which the horizontal axis is not time but some other variable, use the “sgr” or “scatter graph” command. For details, see the help file.

3. Building a Model

From Chapter 1, we are already familiar with building and running models that do not include regression equations. The last model, Model 5, involved regression, but we did not build or run it. Here we add the inclusion of regression equations and review all the steps.

Four of the six steps in building a model appear under the Model menu item. The first step is to **create a master file** which specifies the regression result files — the .sav files — which will be used in the model and other identities necessary for the model. Selecting Model | Master will show the current contents of the master file in the editor. Note that each regression equation is brought into the model with a command of the form “add xxx.sav”, where xxx is the usually the name of the dependent variable in the regression. This must be the same name that appeared on the “save” line in the .reg file by which the equation was estimated. When you are adding an equation to the model, the first thing you should do is to modify the master file to include the new equation. Generally, it is not crucially important where the “add” command appears, but it is natural to place it after the determination of the explanatory (right-hand-side) variables which it uses.

At the bottom of the master file there should be one or more “check” commands. Model 5, for example, has the command

```
check cR .2
```

The check commands control the iterative solution of simultaneous models as explained relative to Model 5 in Chapter 1.

Note carefully that you conclude the editing of a Master file by clicking Save on the editor menu — **not** by clicking Run. You want to use the Master file as input into the Build program, not the G program. Clicking Run will accomplish nothing useful and may consume a good bit of time as you respond to various “bad command” messages.

The second step is to **estimate all the equations** and save the results in .sav files. This step is not mentioned under the Model menu items, because it involves all the other capabilities of G. We have already seen several examples and the following chapters have many more.

The third step is to **build the model**. This step is accomplished by selecting Model | Build. For it to work completely, the C compiler must have been installed. This step runs the Build program, which reads the master file and all the .sav files, writes a C++ program, hearta.cpp, to execute the model, and compiles and links this program to produce an executable file, run.exe, which is the model in the form of a program ready to run. This step also creates

bws	A data bank containing all the variables used in the model.
run.gr	A command file for G to graph all the variables in the model.
run.err	A record of all the diagnostic information provided by the Build program. Be sure to check it by looking at it with the editor. If you have included the command “addtype y” in your Master file, this file will also contain everything in the Master file and in the files which are “added.” If you include at the top of the Master file the command “checkdup y”, Build will check every “f” or “fex” command to see if it is calculating a variable which has been calculated before and, if so, whether the values are identical. This check is a powerful way to spot a source of frequent errors in building models. Check your run.err file often.
exogall.reg	A command file for G7 to create mechanical projections of all exogenous variables in the model by regressing them on time. Look at exogall.reg to see what it does. The projections will be put into files with the variable name as the name of the file and “xog” as the extension.
run.xog	A file to use all the .xog files made by running exogall.reg.
skipall.fix	A file to “skip” the calculation of all the variables calculated by regression equations. When all the regression equations are skipped, these variables become essentially exogenous and the model becomes tautological. Hence, its historical simulation should completely reproduce history. If it fails to do so, there is something wrong with the identities. Since it is <i>very</i> easy to make mistakes in the identities, it is always important to run a model with all regressions skipped. This file makes it easy to do so.
run.nam	A file with the names of all variables in the model and their numbers necessary to read the hearta.cpp program should you have to do so. With luck, you won’t need to ever look at hearta.cpp or run.nam.
heart.dat, heart.dim, run.lag, run.map	These files are internal to the system. You should not need to ever look at them.

The fourth step is to **run the model**. It is accomplished by selecting Model | Run. You will be asked to fill in a form with the starting and stopping dates, the name of the output bank, and the name of the “fix” file. For the quarterly demonstration model, use the most recent twenty years for which you have data. The first simulation can be a historical one with actual values for all of the exogenous variables. Hence we may name the output bank “histsim”. The “fix” files are where we specify various alternative scenarios or assumptions for forecasts. For the moment, we want a fix file with only rho adjustment values. This file is called rhoadj.fix, but we leave off the .fix in entering its name in the dialog box. When you click the OK button, you get a] prompt. Give a full title to your run, something like

```
ti Historical Simulation of Model 4
```

You get the] prompt again, and you type “run”, and the model runs.

The fifth step is to **look at the results graphically**. To do so for a historical simulation, we need to assign in one bank the actual data in the bws bank and in another the alternative for comparison — here histsim. We do so with the commands

```
bank histsim b
bank bws c
```

Then we have a number of “title”, “gname”, and “graph” commands to make the graphs. We can put all of these in an add file, such as hist.shw, when we execute the file by

```
add hist.shw
```

A simple hist.shw file is may look like this:

```
bank histsim b
bank bws c
gdates 1980.1 2013.3
title GDP -- REAL GROSS DOMESTIC PRODUCT
gname gdpR
gr b.gdpR c.gdpR

title cR -- CONSUMPTION
gname gcR
gr b.cR c.cR

title vfr -- GROSS PRIVATE FIXED INVESTMENT
gname vfr
gr b.vfr c.vfr
```

and so on for as many series as we want to graph. In the “gr” lines, the “b.” and “c.” tell *G7* to look for these series in and only in bank b or bank c, respectively. The effect of this command, therefore, is to graph the gdpR series from histsim as the first line, the red line, and then to graph the gdpR series from bws as the second, the blue line. The result is a graph with the actual historical course of gdpR shown in blue (true is blue) and the model’s calculated gdpR shown in red.

The sixth step is to **make tables**. Table-making is found on the Model | Tables menu item. Selecting this item brings up a form for preparing input data to the *Compare* program, which will actually make the tables. At the top of the form, you are asked for the name of the “stub” file. The stub file fundamentally gives the names of the series to be put into the table and the dates to be listed. It may contain, however, many commands to the *Compare* program which are described in the *G7* help files. An appropriate initial stub for the *AMI* model is in the file ami.stb, so put “ami” in the stub file edit window. We will come back to the contents of the stub file in a moment.

At the top right of the form is space for the name of the “output file,” the text file which will be produced as the output of the comparison. An acceptable entry would be “histsim.out”.

Next, you must specify how you want to show the alternative bank. The base bank, the first, will always be shown in actual values. The others, however, may be shown as deviations from the base, as percentage deviations from the base, or as actual values. Usually — but not always — the best choice is deviations, so that is the default value.

Then we must enter the types and names of the banks we want to compare. The types are easy: throughout Parts 1 and 2 of this book, the type is always *workspace*. Since we are making a comparison of a the histsim bank against the bws bank, put “bws” in the name field for the first bank and “histsim” in the name field for the second bank. Then click OK.

Compare is a console application program; it will run in a window. Normally, there is nothing to do except tap any key when the program has finished. However, if *Compare* finds fault with your commands, it will let you know in this window, so look for messages before blithely tapping a key.

When *Compare* has finished and the window has closed, look at the output file, histsim.out, in our example, with the editor. Be sure that the editor is using a monospaced font, such as Courier, Line Printer, or Fixedsys; otherwise, the columns of numbers may not line up vertically.

Now let us look at that ami.stb file. Here it is.

```

\dates 1990 1995 2000 2008 2009 2010 2011 2012 2013
\under =
\7 1 65 2 2 45
;
&
gdpR          ;Real Gross domestic product
cR            ; Personal consumption
vR           ; Gross private domestic investment
vfR          ; Fixed investment
viR          ; Inventory change
xR           ; Exports
mR           ; Imports
gR           ; Government purchaes
;
gdpD          ;GDP Deflator
gdp           ;Gross domestic product
pibg         ; Personal Income before gov
nitpils      ; - Taxes on production & imports
ptax         ; - Personal taxes
npctr        ; + Gov transfers to persons
pidis        ; = Personal disposable income
piipcb       ; - Interest paid by consumers
piptt        ; - Transfers to foreigners
c            ; - Personal consumption expenditure
pisav        ; = Personal saving

```

The commands beginning with a \ are commands to *Compare*. The first one, and the one you may well wish to change, sets the dates for the data to be shown. A date like 2000 is an annual date, and *Compare* will take an annual average of the quarterly data in our model. We could also give a quarterly date like 2010.2. If we include intervals in the dates, like 2005-2010, *Compare* will show the growth rate over that period. The growth rates will be calculated with continuous compounding. (Other compounding specifications are available, but are generally inferior to the default.) The \under command just sets the character which will be used for underling in the table. The \ command by itself, as in the third line, sets the following parameters:

fw field width of the numbers in the table

dp	decimal places to be shown in the table
pl	page length, in lines
tm	top margin, in lines
bm	bottom margin, in lines
tw	width of the titles on the left side of the page.

An `&` at the beginning of a line causes the dates to be written across the page and underlined with the underline character. A line with a `*` in the first position forces a new page. The other lines, that is to say, most lines, have variable or expression on the left followed by a `“ ; “` followed by a title. The title will appear at the left of the page and the value of the expression will be shown in the columns under the appropriate dates.

There are many other *Compare* commands. You may also wish to look at `Compare.pdf` which is in the `pdg` directory.

You are now in a position to begin trying to use the software by doing the exercise at the end of the chapter.

4. Using Help

As is usual in Windows programs, the last item on the menu is Help. When you click on it, you should get a window with two books, one labeled Tutorial and the other labeled Command reference. If you click on either of them, it will open and show you the topics in it. To go to one of these topics, double click on it or put the cursor on it and tap the ‘Enter’ key. The text for the topic will appear in a box at the right of the screen. You may need to enlarge the window to read it easily. The tutorial is similar to what you have just read in this chapter. You can move forward or backward through it by using the “browse” buttons at the top, marked by `>>` and `<<`. After the tutorial, you will find the command reference for *G7*. These commands are those which you give in the white command box or include in a command file to be “added.” If you click on “contents” at the top, you get back to the window with the books. If you click “index” you get references to keywords which the help author identified in the text. When you pick one of these, you get in the lower window a list of topics where the keyword occurred. If you still do not find what you want, you can do a full text search of the help material by clicking on the “Find” tab and waiting — at least the first time — while the computer makes a listing of every word that occurs at any point in the help text.

In the Command reference section, you should first read the first three topics. After that, each topic should be self-contained. It is my own experience in using help files for Windows programs that they are often clipped, circular, and unclear. They try to substitute hypertext links for old-fashioned clarity. In contrast, I hope that you will find the *G7* help text clear and with what you need to know all in one place. Examples are plentiful.

Exercises

- 3.1. Create and run .reg files for all of the regressions used in Model 5 of Chapter 1. Modify the Master file as shown in Chapter 1 to make the model include these regression equations. Build and run the model in historical simulation. Make graphs and tables. In doing this exercise, you should read over the section on Model 5 in Chapter 1 and then try to do the whole exercise without looking again at Chapter 1. Only if you get hopelessly stuck should you look back before finishing. Then you might want to compare the way you did the exercise with the way I did it.

- 3.2. Develop an equation relating employment (*emp*) to real GDP, *gdpR*, include it in your model, build and run the model in historical simulation. You should:
 - Modify the master file to use the new equation
 - Create a file with the editor to estimate the equation
 - Run that file so as to get the results in a .sav and a .cat file.
 - Choose Model | Build from the main menu
 - Choose Model | Run to run the new model
 - Edit your .shw file to put in a graph for employment
 - Execute this file and save the employment graph
 - Edit ami.stb to put in a line for employment
 - Choose Model | Tables to make the numerical listing of the results.

When all the computing work has been done, go into your word processor and bring in your new master file, your emp.reg file, the resulting .cat file, the graph of the fit of the equation, the graph of employment in the historical simulation of the model, and the table comparing the historical simulation with actual values. You will find that to show the text files correctly with numbers in nice straight columns, you need to use a monotype font such as Courier or Line Printer.

CHAPTER 4. FORECASTS AND EXPERIMENTS

Let us take stock of where we are, what we have done, and where we have to go. In Chapter 1, we saw what a model was, but we learned little about how to actually make one. In Chapter 2, we looked carefully at theoretical aspects of fitting equations to data. In Chapter 3, we first saw how this curve fitting was done in practice with the *G7* program and how the equations can be combined into a model, and how that model can be run in historical simulation, that is, with historical values of all the exogenous variables. In this chapter, we will see how to run counter-historical simulations or “experiments with the past” by changing the values of one or another of the model’s exogenous variables from what they actually were to what they might have been. The model will then spell out the effects of these changes on all the endogenous variables in the model. In this way, it gives us a sort of controlled experiment by means of the model. We will also see how to use the model to make forecasts. We have but to forecast the exogenous variables and then use the model to calculate the endogenous variables. Indeed, we can make not just one forecast but a number of them, with different projections of the exogenous variables. Usually a number of the exogenous variables represent policy variables such as government spending, tax rates, or money supply. Thus, comparison of the outcomes of the simulations with different projections for these variables enables us to evaluate the wisdom of pursuing various policies. This use of models is perhaps their most important application.

Unfortunately, I have found no way to prevent users of the *G7* software from occasionally making egregious errors that cause their models to explode or collapse or, more rarely, hang. There are, however, several tools for finding these errors, and they are described at the end of this chapter in the section on debugging of models.

Some of the deficiencies of *AMI* are already visible and more will become apparent in this chapter. In the next chapter, Chapter 5, we will look at some techniques for developing equations which can, to some extent, remove those deficiencies. That chapter ends with a list of problems which outline a research program for turning *AMI* into a fully respectable macroeconomic model. You can, at that point, set to work on making your own model. The final chapter, Chapter 6, reviews a number of qualitative factors in judging an equation. Fundamentally, an equation should make sense. It should, at a minimum, be conceivable that such a relation as you are estimating should exist in the economy over a number of years. It is surprisingly easy to write down equations that fit fairly well but make no sense. Chapter 6 develops a checklist of pitfalls to look out for.

Your work through the rest of this book will be an iterative application of the material in the last four chapters. To improve the simulation or forecasting properties of your model, as found using the techniques of this chapter, you will look for new relations using, perhaps, the techniques described in Chapter 5. Then you pass your new equation in review before the checklist of Chapter 6. When it passes, you return to the techniques of Chapter 3 to include it in the model, and build and run the model in historical simulation. Then you come back to the methods of this chapter to test the model in counter-historical simulation and forecasting. When you are satisfied with those results, you are ready to tell the world -- or at least your class -- about your model.

1. Counter-historical Simulation

You might well suppose that a good performance in historical simulation would be a quite adequate test of a model. Indeed, it is a necessary condition for a model to be useful. But it is by no means sufficient. For it is easy to build a model that does quite well in a historical simulation, especially over the period for which it was fit. The trick is simply to make some key variables exogenous or derived *by identity* from exogenous variables.

For example, the equation for fixed investment in *AMI*, (in the *vfR.reg* file) contains the lines

```
f d = gdpR - gdpR[1]
```

Now if we change from defining *d* in terms of *gdpR* to defining it in terms of *gnp\$*, thus

```
f d = gnp$ - gnp$[1]
```

we will see no appreciable change in the fit of the equation but a big improvement in the historical simulation of the model. Try it for yourself. It works like magic, stage magic, that is, for the improvement is only in the appearances, not in the essence. Indeed, the essence of the model is much damaged by the change. Why does the change work such wonders on the historical simulation? Just because *gnp\$* is not otherwise in the model. Formally, it is exogenous although logically it should be endogenous. But because it is exogenous, when the historical simulation is run, fixed investment will be calculated from actual, historical real GNP, not that calculated by the model itself. Thus, the simulated value of *vfR* will be almost exactly the predicted value found in fitting the equation. That fit, you will recall, was much closer than that of the historical simulation where the simulated value, not the historical value of *gdpR*, had to be used in calculating *d*.

We will call this sort of deceptive exogeneity *false exogeneity*. It usually occurs by accident, unthinkingly. But the possibility of this occurrence is enough to make us leery of judging a model by its historical simulation alone. And so we are led to try some counter historical simulations.

A counter-historical simulation is just a simulation over the historical period with one (or possibly more) of the exogenous variables set at values different from what they actually were historically. The simplest way to do this is to start the run of the model (with Model | Run) just as you would for a historical simulation except that in the bottom box where you have previously given “*histsim*” you give a name (of eight or less letters and no spaces) for the data bank to be created by this run. For example, if you are going to increase government purchases of goods and services by \$20 billion, you might call the bank “*gRup20*”. When you click OK and get the] prompt, give a title something like

```
ti Counter-historical simulation with gR increased by $20 billion.
```

You then get the] prompt again. Give the command

```
type gR
```

and you will get the historical values of *gR* displayed on the screen and the] prompt again. This time give the command

```
update gR +20
```

and then again

```
type gR
```

By comparing what you now get with what you got from the first “type gR”, you should see that each and every value of gR has been increased by 20. You can abbreviate “update” to “up”. To subtract \$20 billion, the command would be

```
up gR -20
```

or to increase each value by 20 percent,

```
up gR *1.20
```

To specify the complete course of gR to see, for example, the effects of a major reduction in gR beginning in 1995 and falling from 1684 at the beginning of the year down to 1500 by 1997.1 and remaining there for the year and then rising steadily to 2200 by 2013.3, you can use the following:

```
up gR
1995.1 1684 0 0 0 0 0 0 0 0
1997.1 1500 0 0 1500 0 0 0 0
1999.1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
2005.1 0 0 0 0 0 0 0 0 0 0 0 0 0 0
2010.1 0 0 0 0 0 0 0 0 0 0 0 2200
```

Here we have made use of the capacity of the program to replace a zero by a linearly interpolated value. In effect, the program draws a straight line from the 1684 in 1995.1 to the 1500 in 1997.1, and then another straight line from the 1500 at 1997.4 to the 2200 at 2013.3. Remember that the first number on each line is the date of the *first* observation on the line. There is no effective limit on how many numbers can be on the line. As you can see, you have the capacity to run a wide variety of counter historical simulations by altering the exogenous variables.

You can also modify the values produced by the equations of the model. You could ask, for example, how would the economy have behaved if investment had always been 20 percent above what the equation predicted. Such modifications of the predicted values are referred to as “fixes” There are four types of fixes available to you:

cta	Constant term adjustment (also called "adds"). Add a specified amount to the value calculated by the equation. The amount added can be different in different periods.
mul	Multiplicative fix. Multiply the value calculated from the equation by a specified factor, which may be different in different periods.
ovr	Over-ride the equation. Discard the value calculated and use the one specified.
rho	Add a rho-adjustment to the variable
skip	Ignores the value predicted by the equation and uses the historical value in its place.

The format for the first three is similar:

```
fixtype <variable_name>
data lines
```

For example

```
cta cpcR
1998.3 -20 -20 -25 -30 -30 -25 -15 -12 -10 -8 -4 -2 ;
```

The format for the rho-adjustment fix is

```
rho <variable_name> <rho_value>
```

for example

```
rho vFR .9
```

The format for skip is just

```
skip <variable_name>
```

for example

```
skip cpcR
```

These fixes work only when applied to variables which are the dependent variable in a regression. When applied to other variables, the forecasting program does not reject them; it just never uses them. Furthermore, *only one can be applied to any one variable*. If more than one is applied, only the last takes effect. For example, if somewhere in the fix file after the above "cta cpcR" there should appear "rho cpcR .9", the cta will be completely forgotten by the program without any warning.

Constant term adjustments are useful but easily abused. If, for example, construction of a major new oil pipeline had recently been announced, it might be fair to assume that the equation in the model doesn't know about the announcement and would forecast better with the help of a constant term adjustment. On the other hand, the worst abuse is to piece out the imperfections of the model in running a scenario. For example, if the scenario is an increase in money supply, and the model fails to show an increase in investment, it is an abuse of the cta to use one to pump up investment and to claim that the rise was a result of the increase in money supply, something "we know has to happen even though the model refuses to show it." This abuse, unfortunately, is not unknown in the commercial model building industry. In my own forecasting, I make extensive use of rho adjustments, which are a sort of automatic cta, but I seldom use a cta directly.

The values of *rho* in the various regressions can be conveniently captured for use in a .fix file by G7's "rhosave" command. The format is

```
rhosave <filename>
```

for example,

```
rhosave rho.fix
```

After giving this command, estimate all the equations by doing "add runall.reg" — a good reason for keeping your runall.reg file up to date as you add more regressions to the model. Then give the command

```
rhosave off
```

and the file in which the values of rho are being saved will be closed and available for importing into the rhoadj.fix file or any other .fix file.

When making tables from a counter-historical run, the base of comparison is nearly always the historical simulation and *never* the actual historical values. Why? If we compare with the historical simulation, we know that the differences we are looking at are the result of the changes we made in the exogenous variables or in the equations by the fixes. Whereas if we compared the counter-historical simulation with the actual historical values, the differences would be a combination of errors in model and the effects of the changes we made in the exogenous variables. The whole value of having the model is to isolate the effects of the changes you made, so be sure to use histsim as the base against which to compare your counter-historical simulation.

Exercise

- 4.1 Do the suggested counter historical simulation with gR increased by \$50. Then try \$100 and \$200. (The table maker will let you have up to eight alternatives.) Does there appear to be any limit on how big gdpR can be made by increasing government expenditure? Is anything bad happening as a result of increasing gR? Do you believe that what you have found to be properties of the *AMI* model are in fact properties of the U.S. economy? If not, you have used counter-historical simulation to reveal defects of a model. How do you think *AMI* might be improved?

2. Forecasting

A forecast is made, of course, by running the model over a future period. Like any other run, it is made by clicking Model | Run and filling in the blanks. But, unlike historical and counter historical simulations, a forecast will not have sensible values of exogenous variables unless you provide them. You do so via the .fix file.

The work of forecasting therefore centers on preparing this fix file. It must contain projections of all the exogenous variables and may contain also instructions for modifying the results of the regression equations by cta, mul, ovr, and rho fixes.

I suggest that you make a file called fore.fix to hold this information. To make it, start *G7*'s editor (File | Editor) with a blank screen. (If the editor already has something in it, use the Editor's File | New command to clear the screen.) Then use the Editor's File | Insert command to insert the file rhoadj.fix. The top line of the file gives a title which you should be sure to change for your forecast. With the cursor just below this line -- that is, at the beginning of the second line -- select File | Insert, and insert the file run.xog. This file was created when you did Model | Build. Here is what it contains for the basic *AMI* model.

```
add time.xog
add gdpD.xog
add xR.xog
add gR.xog
```

```

add pibgBR.xog
add nitpilsBR.xog
add ptaxBR.xog
add npctrR.xog
add piipcbBR.xog
add pipttBR.xog
add pisavBR.xog

```

As you see, it is a series of “add” commands to include in the data for your forecast the values of each exogenous variable in the model. You must create the .xog files with the future values of each exogenous variable. (In the *AMI* model, the first is for “time”; future values for it are already in the model's historical bank, and you should delete the line for it in the fore.fix file.) You should now save this file. You are perfectly free to save it with any legal file name provided that you remember what it is, but I will assume that you used fore.fix.

Now you must get to work creating the .xog files that have the future values of the exogenous variables. The format of these files is familiar. Future values are specified by "update" commands exactly like the update (or up) commands used in the counter historical simulation but with future dates. (Of course, update commands like “update xxx +20” will not work because there is nothing to which to add the 20.) When forecasting, you can use linear interpolation to automatically fill in zero values in the exogenous variables. (Actual values of zero for an exogenous variable should be replaced by very small values.)

For exports, xR, we could express its forecast in either of two possible ways in the xR.xog file. We could have

```

update xR
2011.4    1856.269    1852.852    1850.222    1848.333
2012.4    1847.142    1846.607    1846.690    1847.354
2013.4    1848.566    1850.293    1852.505    1855.173
2014.4    1858.271    1861.774    1865.657    1869.899
2015.4    1874.478

```

or

```

update xR
2011.4    1856.269    1852.852    1850.222    1848.333
2012.4    0 0 0 0    0 0 0 0    0 0 0 0
2015.4    1874.478

```

The second makes use of the interpolation abilities of the update command, whereas the first provides a value for every period. Because this projection is very close to a straight line, there is little practical difference between the two.

The first of these updates was prepared in a very mechanical way by regressing xR on time and using the predicted value of the forecast using the rho adjustment. The historical values followed by this forecast are in the variable called *depvar* created by *G7* when the "mode f" has been given. (The f in "mode f" is for "forecast".) The *G7* commands to create this file were

```

bank bws b
mode f
tdates 2013.4 2017.4
limits 1980.1 2013.3 2017.4
ti xR
r b.xR = time
gr *
f xR = depvar
save xR.xog
sty xR
save off

```

This command file first assigns the data bank “bws”. This bank is the one created when the model was built. It has all the variables in the model and no others. Most of the variables in the model are in the Quip data bank which is automatically assigned when *G7* is started, but some, such as *taxrate* and the various behavioral ratios, are made up in the process of building the model. By assigning the bws bank as bank b by

```
bank bws b
```

these variables become available by putting a “b.” in front of their name. In this command file, the *limits* command caused the equation to be fit from 1980.1 to 2013.3 and then forecasted to 2017.4. The forecast worked because the only independent variable is time, one of only two variables with future values in the Quip bank. After the forecast is completed, the *depvar* variable (*depvar* = dependent variable) contains the actual values of the dependent variable for the historical period and the forecasted values with rho adjustment for the future period. The command

```
f xR = depvar
```

gives xR not only the correct historical values but also the forecast. After opening the *xR.xog* file for saving, the command

```
sty xR
```

causes the past and forecasted values of xR to be written to *xR.xog*. (*sty* is short for “silent type” — “silent” because the values are not written to the screen.) This command wrote into the *xR.xog* file the first of the “update xR” displays shown above. In the regression, we could have used, besides

```
r xR = time
```

also

```
r xR = xR[1]
```

or even

```
r xR = time, xR[1], xR[2]
```

for lagged values of the dependent variable are available for forecasting. The resulting forecasts are, of course, somewhat different.

The *fore.fix* file may contain either the command “add *xR.xog*” or the contents of the *xR.xog* file can be brought into *fore.fix* and the line “add *xR.xog*” deleted. I myself generally prefer this second way of working, because it puts all of the projections in one place where they are easily seen by the user.

The sort of mechanical projection illustrated here for exports makes sense for a few variables like population and possibly exports. It does not make sense to use it for tax rates, defense expenditures, and other policy variables. Projections of such variables should be thought about carefully in the light not only of their recent trends but also of current political developments. Mechanical forecasts can also be made for these variables, but *only as a starting point for thinking about them*.

If you discover any surprise exogenous variables in `run.xog`, ask where they came from. Did you really mean for them to be exogenous or are they just sort of accidental exogenous variables? If the latter, you have false exogeneity and should modify the model to eliminate it. If you find yourself making exogenous projections for variables that are clearly not independent of the endogenous variables in the model, revise the model to eliminate them. Typical examples are to find that `gdp` (current dollar GDP) is exogenous while `gdpR` is endogenous. Or that the consumer price index (`cpi`) is exogenous while the `gdp` deflator, `gdpD`, is endogenous. If you find yourself making exogenous projections of variables that are *not* largely external to the economy in the short run (such as population and labor force) or that do not represent policy (such as money supply, tax rates, or government spending), **stop!** Your model has in it one of those tricks that both make the historical simulation look good and render the model useless. Get rid of the false exogeneity by either adding an equation for the variable or use a variable that is already endogenous in place of the offender.

After this exhortation to think about your projections of the exogenous variables, I must now confess that the *Build* program automatically made a file for making these mechanical projections of all of the exogenous variables. It is called *exogall.reg*. To use it, you must first edit the second and third lines to specify the dates over which you want the regression run and the projections printed. (If your forecasts fail to go as far into the future as you intend, you probably need to add an “`fdates`” command to specify the range over which “`f`” commands work, for example:

```
fdates 1960.1 2020.4
```

That done, just execute the file, and you will get `.xog` files for all the exogenous variables. Look at one or two of them such as `xR.xog` or `gR.xog`. The `fore.fix` file can then be made to use all these exogenous projections inserting into it (File | Insert) the `run.xog` file.

I repeat, use these mechanical projections *only* as starting points for thinking about economically justifiable projections. When you have thought about a particular exogenous projection and edited the `.xog` file that contains it, I suggest that you save your edited version with the extension `.xg` instead of `.xog`. Edit *fore.fix* to use the `.xg` file. The reason for this suggestion is to avoid accidental destruction of your carefully edited files by doing “*add exogall.reg*” again, perhaps because changes in the model have added some new exogenous variables.

Inclusion of the rho adjustment factors is particularly important for a good short-term forecast, for they avoid abrupt changes in variables as the model “jumps onto” the equation. Use the *rhosave* command described above to get all the rho fixes together in a convenient way.

You can, of course, use the `cta`, `mul`, and `ovr` fix types as well. If you have used a “`cta`”, “`mul`”, or “`ovr`” adjustment on any variable, say `x`, you can check the values of that adjustment by doing “`ty x:f`” at the `]` prompt.

Experiments with the model can be very useful both in testing it and, once you are satisfied with it, in learning about the economy.

Exercise

- 4.2 Make a forecast with *AMI* over the next five years. You will have to make independent forecasts of all exogenous variables. Put them, as described above into a file that you may call "fore.fix". When your "fore.fix" file is ready, make the forecast with something like the following on the form obtained by Model | Run

```
Start date    2013.4
Stop date     2017.4
Fix file      fore
Output bank   my4cast
```

You should, of course replace the dates by dates appropriate to the time at which you are working. The Start date must be a period for which full data are available. The name "my4cast" is just an example. Be sure that the fore.fix file has in it rho fixes for all the regression equations.

Use *G7* to graph the forecasts. You will probably want to create a "fore.shw" file which will make graphs with nice titles. Make tables showing your forecast. To do so, you will need to edit the \dates line at the top of the ami.stb file to show the periods you are interested in. To get growth rates, put the dates between which you want the growth rate with a hyphen between them in place of a date in this line. You may find it convenient to keep a copy of the stub file with these dates as fore.stb.

- 4.3. Make other forecasts with *AMI* with different values of the exogenous variables. Compare the results both graphically (with the gr command) and numerically (with Model | Tables).

3. Debugging Models

You may be so lucky as to never have a model that behaves so badly that you know it must have a serious error in it. Chances are, however, that sooner than you would like you will encounter a serious error. The model may not converge, it may produce values of NAN (Not A Number) for some variable, may take the logarithm of a negative number, divide by zero, produce utterly ridiculous values for one or more series, or otherwise complain that it is sick and needs your care. The *G7* software gives you various diagnostic tools for finding the problem.

Even if the model shows no evident problem, you should check the correctness of its identities by running a historical simulation and, at the] prompt, give the command

```
add skipall.fix
```

This skipall.fix is a file created by Build which applies a "skip" command to every regression equation. Consequently, this historical simulation should precisely reproduce the data, that is the bank it creates should be identical to the bws bank. To find out if they are identical, assign one as bank b, the other as bank c, and add run.gr, thus

```
bank histsim b
bank bws c
add run.gr
```

All the graphs should show the two lines on top of one another so that they look like one line. If you

see two lines on any graph, you know you have an error in the identities.

The most common problem is that you have given two different variables the same name. This error is virtually certain to cause trouble if lagged values of either variable are used. *G7* has a special check for this problem. Be sure that the master file begins with the command “checkdup y”; this command will cause Build to check for duplicate names. If it finds such a problem, it will record it in the file run.err. If you also have “addprint y” somewhere before the spot where the trouble occurs, you can pinpoint exactly which statement is changing the value. You should look at the run.err file with the *G7* editor and search for the word “Error”. If you find such a change of definition, you should correct it.

Another debugging tool is the use of the “maxit” command when running the model. You will recall that in each period the *AMI* model goes through a cycle of computations to go from the assumed value of disposable income to the implied value, and then repeats this cycle with the implied value of the first round as the assumed value of the second round. An error in the model will normally make the implied value from the first round nonsense and then everything in the second round nonsense. To spot where the error is, it can be helpful to limit the number of iterations to one. Then the variables calculated in the first period are sensible down to the point where the error occurs; but from that point on, one or more of them is nonsense. Here are the steps in this process. We shall suppose that the historical simulation which is giving trouble starts in 1980.1.

1. Run a historical simulation from 1990.1 to, say, 2000.4, but when you get the] prompt where you would normally type “run”, type first

```
maxit 1
```

and then run. The program will then perform one iteration, report that it has not converged, and ask what you want to do. Answer g for go to next period. Let us say that the bank you produced in this way was called histsim.

2 In *G7*, do

```
bank bws c
bank histsim b
gdates 1990.1 2000.4 2013.3
add run.gr
```

The red and the blue lines should be identical before (to the left of) the vertical line and should be close to one another to the right of it. If you find a difference before the vertical line, there is an error in an identity or “f” statement. If there is a big difference to the right (but no difference to the left), there is an error in the regression equation. With this sort of pinpointing of the error, you should be able to spot it.

If neither of these methods shows up the error, take the model back to its simplest form by removing all the regression equations. To do so, just put a # in front of each line of the form “add xxx.sav” in the master file. If there are no errors in the identities, that should certainly produce a model which runs. Then put the equations back in one by one (by removing the #'s), each time running the model. When the equation you have just put back causes the model to malfunction, you know that the trouble is in the interaction between that .sav file and what is already in the model. That process

should certainly identify which .sav file is making trouble and enable you to see the problem.

If you already know how to use the Borland C++ debugger and wish to use that knowledge to debug your model, you need to know that the model itself is in the file `hearta.cpp` and `run.nam` gives the correspondence between the names of the variables and their position in the `vp` array in the program. It is beyond the scope of this book, however, to explain the use of the debugger. I very seldom find it necessary or helpful for these models.

Exercise 4.4. Check the identities of the *AMI* model.

CHAPTER 5. TECHNIQUES FOR IMPROVING A MODEL

The exercise at the end of the previous chapter pointed to a major problem with *AMI*. As you increased government spending, real GDP and just about everything good got better and better. It would seem that all problems could be solved by the government's just buying more goods and services. Many of the early econometric models from the 1950's and 1960's had this sort of simplistic Keynesian character. Today almost no one would argue that they were adequate representations of economic reality. Making a model show the adverse effects of liberal fiscal (and monetary) policy will take us through some of the major problems of macroeconomics of the last thirty years. It will also require us to introduce some new tools for working with regression.

What happens if the government tries to achieve very high levels of employment by increasing spending? Any student of elementary macroeconomics knows, or believes he knows, that inflation will accelerate, perhaps in a never ending spiral. (If your macro is a bit rusty, look up in your macro book the Phillips curve with the acceleration modification.) According to this theory, it is not the absolute level of inflation which depends on unemployment, but the deviation of inflation from its expected value. This expected value, in turn, is usually thought to be a weighted average of past values of inflation. While the theory says nothing about the distribution of weights in the weighted average, a common assumption is that the weights decline exponentially. Such a variable can be easily created with *G7*'s `@cum()` function.

1. Working with the `@cum` Function

The "cum" in `@cum` is short for "cumulate". The function creates a variable, y , from a given variable, x , by the following formula:

$$y_t = (1 - s)y_{t-1} + x_t$$

where s is a number between 0 and 1. In *G7*, we would create y from x by this equation with the formula

$$f \ y = @cum(y, x, s) \ .$$

(The `@cum` function is slightly strange in that the series which is the result of the cumulation -- which I shall call the *cumulate* and is here indicated by y -- must be given to the function as an argument. That is because, when the model is running, the `@cum` function must know the previous value of y as well as the current value of x in order to calculate the current value of y .) We can write the formula for y_t as a function of the preceding values of x as

$$y_t = \sum_{\tau=0}^{\infty} (1 - s)^\tau x_{t-\tau}$$

For example, if $s = .1$, then by the time we get to $t = 100$, we would have

$$y_{100} = 1.0x_{100} + .9x_{99} + .81x_{98} + .72x_{97} + .656x_{96} + \dots$$

The weights form a geometric series with sum equal $1/s$ if we take infinitely many terms -- and very, very close to $1/s$ if we stop when the individual terms become less than, say, .01 . In many applications, such as the present one, we would like to have the sum of weights equal to 1.0. We can achieve that in either of two, not quite equivalent, ways. The first is to multiply s , since that is equivalent to dividing by $1/s$. For example, if s is .1, we can define y by

$$f \ y = 0.1 * @cum(cx, x, 0.1)$$

This y will have the same dimensions as x . Note that we had to introduce a different variable, cx , which is the pure, unmultiplied cumulate of x as the first argument in the `@cum` function.

The `@cum()` command has an approximate physical analogy in the filling of a leaky bucket from a water pipe. The rate of inflow is x , the level of the water is y ; the outflow through the leak is proportional to the level of water in the bucket. Thus the outflow equals sy and s is called the spill rate.

The problem with this way of making the weights sum to 1.0 is that it does not work exactly, especially not near the beginning of the data bank when we have only a few observations. The second way solves this problem by dividing by exactly the sum of weights of the terms so far included. To get this sum, we just compute the cumulate of the series which is 1.0 from the beginning of the data bank. Following the analogy mentioned above to a bucket with a leak in it, these cumulates of 1.0 are called “unit buckets.” Here is the formula for calculating the ten-percent unit bucket, which we may as well call `ub10`:

$$f \ ub10 = @cum(ub10, 1.0, 0.10) \ .$$

Let us now apply these ideas to computing expected inflation.

2. Rates of Change with Continuous Compounding

The measure of the price level which we shall use is the GDP deflator, for that is the one we shall need to compute GDP in current prices, which we shall soon need. This variable appears in the Quip data bank with the name `gdpD`. It measures the level of prices in any quarter relative to the average in the base year, currently 2009. It is thus not directly a measure of inflation. Rather, the corresponding measure of inflation is the percentage rate of change of `gdpD`. How do we compute this percentage rate of change? One possibility would be

$$f \ infl = 400. * (gdpD - gdpD[1]) / gdpD[1]$$

Why the 400 on the right instead of the 100 you perhaps expected? Because we are working with quarterly data so the change observed is only over a quarter of a year, but inflation — like interest rates with which they are often compared — are conventionally stated in percent per year, not per

quarter. Please compute this variable and graph it. You will find a very jagged series. To get a smoother series which we have a better chance of explaining, we will take inflation over the last year. It can be calculated in *G7* by

```
f infl = 100.*(gdpD - gdpD[4])/gdpD[4]
```

An alternative definition is to take the natural logarithm of *gdpD* and then to take the difference of the logarithm, like this:

```
f lgdpD = 100.*@log(gdpD)
f infl = lgdpD - lgdpD[4]
```

The first definition corresponds to the idea of annual compounding of growth rates, while the logarithmic definition corresponds to continuous compounding. For natural growth processes, continuous compounding is the only natural way to measure the rate of growth. All others introduce an extraneous element — the period of compounding — into describing the process. Further, continuous compounding is the only way to get the same growth rates both up and down. If, for example, a series goes from 100 in year 1 to 120 in year 2 and back to 100 in year 3, annual compounding says it went up by 20 percent and then fell by 16.67 percent to get back to where it had been. Thus the rise seems to have been bigger than the fall! The continuous, logarithmic definition says it rose by 18.23 percent in the first year and fell by 18.23 percent in the second, so that the two changes are recognized as being of the same magnitude but in opposite directions.

3. Inflation I — The Wolf in Sheep's Clothing

Now with inflation defined, we can undertake to define expected inflation with the `@cum` function. We would have

```
f lgdpD = 100.*@log(gdpD)
f infl = lgdpD - lgdpD[4]
f ub10 = @cum(ub10, 1.0, .10)
# call expected inflation "inflex"
f inflex = @cum(cinfl, infl[1], .10)/ub10
```

Put these commands into *G7* and graph the series you have created. Since the price deflator data in the Quip data bank begins in 1969, you might want to begin some of the graphs back near that date to watch how the `@cum` function works.

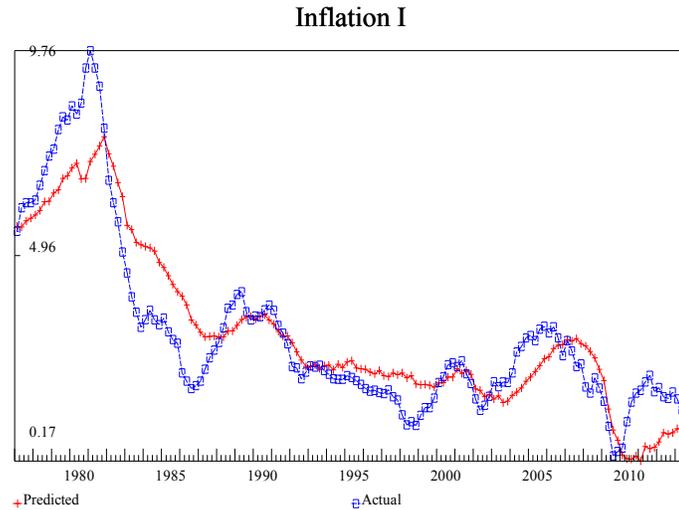
Before we can try an equation for inflation, we need to define unemployment in terms of the employment variable for which you already have an equation. Besides *emp*, the Quip data bank also contains a variable for the civilian labor force, *lfc*. To compute the unemployment percentage we need

```
f u = 100.*(lfc - emp) / lfc
```

You can now put all the preceding ideas together and formulate a regression as follows:

```
r infl = inflex, u, u[1], u[2], u[3]
```

The fit, shown below, over the period 1977.1 to 2013.3 is surprisingly good considering that only the unemployment rate has been used as an explanatory variable. However, it does predict extremely low inflation in the last several quarters, due to the high unemployment rate.



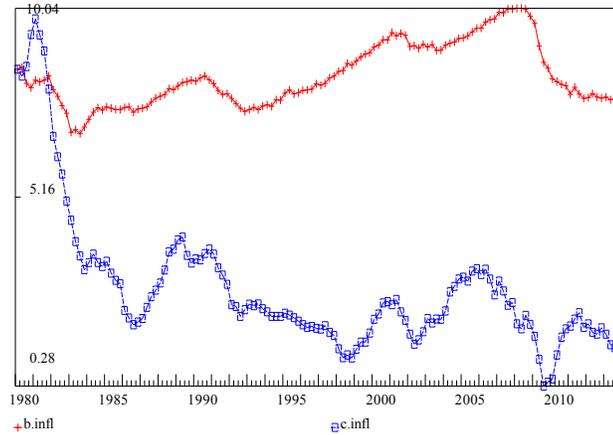
Now put the equation in the model and run it. Because we are interested here in testing how well this equation simulates when it has to use (in the definition of *inflex*) the prior values of *infl* which it itself generated, I have isolated this equation from the rest of the model. Namely, I have used actual unemployment, not that generated by the model. This change required simply replacing the line defining *u*, given previously by

```
f    u = 100.*(lfc - emp) / lfc
by
fex  u = 100.*(lfc - emp) / lfc
```

You will recall that a “fex” is used for forming exogenous variables. It puts the variable on the left into the model, but not the equation. When we go to actually use the model, we certainly want *u* to be endogenous. For the moment, however, so that we can concentrate on the interaction of the inflation equation with itself, the “fex” definition is desirable. Subsequent simulations of the inflation equation in this chapter are all done with the equation isolated in this way.

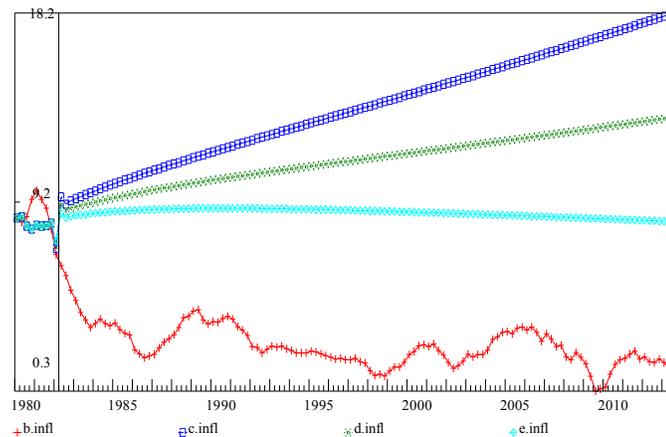
Once the model has been run in historical simulation, plot *infl* from the *bws* and *histsim* banks. The result, shown below diverges quickly, and is much too high in the rest of the simulation. Remember that the fitted equation (see above) was not bad. Although the forecast roughly follows the pattern of actual inflation. It is well above the actual level throughout the rest of the forecast period.

infl - Inflation
Model 7: Simulated vs. Actual



The beginning of that divergence warns us that we should look carefully at the equation for explosive tendencies. One way to check for them is to run the model with constant values of unemployment. Since we have left unemployment exogenous, we can easily set its values when we run the model. The lines marked by + signs in the graph below shows the results of running it from 1982.1 to 2013.3 with three different constant values of unemployment, namely 4, 5 and 6 percent, to produce the top, middle, and bottom lines, respectively. Here, the dangerously explosive nature of this equation becomes apparent. It is hard to believe that at an unemployment rate of 4 percent, prices would go into an upward spiral reaching 18 percent per year after 30 years! Indeed, this equation turns out to be a wolf ready to eat alive an unsuspecting modeler.

Inflation 1 Equation in Historical Simulation
Unemployment Rate Set to 4, 5, 6



This example shows the value of a historical simulation, for while we have seen that a good historical simulation is no guarantee of a good model, a bad one, such as we have here, is surely an indication of an unusable model.

Since the equation is misbehaving badly, we had best have a close look at it. Its numerical output is shown below.

```

:
Inflation I
SEE = 0.90 RSQ = 0.8128 RHO = 0.95 Obser = 147 from 1977.100
SEE+1 = 0.28 RBSQ = 0.8062 DW = 0.10 DoFree = 141 to 2013.300
MAPE = 29.45
Variable name      Reg-Coeff  Mexval  Elas  NorRes  Mean  Beta
0 infl            - - - - -  - - - - -  - - - - -  - - - - -  3.17 - - -
1 intercept       1.59921   7.7     0.51  5.34    1.00
2 inflex          1.04040  126.8   1.14  1.28    3.48  0.963
3 u               0.15231   0.1     0.31  1.02    6.45  0.117
4 u[1]           -0.57609  0.3     -1.17 1.00    6.45 -0.441
5 u[2]           0.12615  0.0     0.26  1.00    6.45  0.097
6 u[3]           -0.02141  0.0    -0.04  1.00    6.45 -0.016

```

Look at the coefficient on *inflex*: 1.04. That means that if expected inflation rises by one percentage point, inflation itself will rise by 1.04 percentage points. Notice in the graph of the fit that there were positive errors in 1997 for several consecutive quarters. When these quarters were hit in the simulation, expected inflation rose well above actual inflation and the 1.04 “perpetuation factor” took hold and generated this runaway inflation. It could very well happen, by the way, that a few consecutive quarters of misses on the low side could start a plunge of inflation down into negative values.

A more reasonable value for the coefficient of *inflex* would be 1.0 or perhaps something a bit below 1.0. Values above 1 for this coefficient are almost certain to make for an unstable model. Of course, *G7* did not realize that our *inflex* was expected inflation. It had no idea of what was a reasonable value for its coefficient. It just picked one to minimize the sum of squared errors. How can we tell *G7* what would be a reasonable value? That question brings us to the matter of soft constraints.

4. Soft Constraints

Suppose that we want to suggest to *G7* that a value for b_2 of .9 would be appropriate. How can we do so? We can simply manufacture some new, artificial observations. In these observations, the value of x_2 , *inflex* in the present case, is 1.0, the value of *infl*, the dependent variable, is .9, and the values of other variables are 0 (including x_1 , the variable which is multiplied by the intercept). Thus, we have an observation that looks something like this

x_1	x_2	x_3	...	x_n	y
0	1	0	...	0	.9

The residual from this observation will be $.9 - 1b_2$. Pretty clearly the value of b_2 which minimizes the square of this one residual is .9. Of course, there are other residuals whose squares go into the sum

of squared residuals which is being minimized, so including one such observation won't result in a b_2 which is exactly .9. But as more and more such observations are added, the value of b_2 will certainly move towards .9. Thus, we can say that by the addition of such observations we are *softly constraining* the estimated regression coefficients.

The command in *G7* for adding one such artificial observation for each natural observation is just

```
con 1 .9 = a2
```

and to add 10 such observations per natural observation is

```
con 10 .9 = a2
```

The general form of the constraint or con command is

```
con <count> <left> = <right>
```

where

count is the number of artificial observations per natural observation.

left is a number, the left side of the equation which expresses the constraint.

right is any linear expression in the coefficients of the equation.

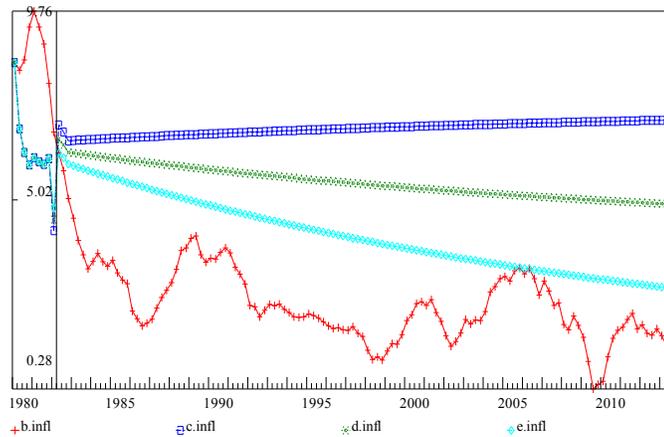
Some examples of con commands illustrating the “right” element are

```
con 3 .9 = a2
con 200 0 = a2 - 2a3 + a4
con 1000 1 = a3 + a4 + a5 + a6 .
```

(The letter a is used here to denote the regression coefficients instead of the b which we have used in the text because *G7* is able to estimate several regression equations at once with constraints across the equations. In such tasks, a is used for the coefficients of the first equation, b for coefficients of the second, and so on. We will come to such problems in Part 2.)

The natural question at this point is: “How do I know how many of the artificial observations to use?” Is it really conceivable that, as the third example seems to imply, I would ever want to use 1000 artificial observations per natural observation? The only general rule about how many to use is “As many as it takes to accomplish what you want done.” And yes, it is quite common to take large numbers of artificial observations, because their impact depends on the dimensions of the variables. Suppose, as in the third example above, that we are trying to constraint the sum of four coefficients, all of them between 0 and 1, to be 1.0. The discrepancy is likely to be .2 or .4 at worst. Now if the dimension of the dependent variable is such that the *SEE* for the equation is 1200, then it is going to take a lot of observations with an error of .2 to make the least-squares algorithm “listen” to the constraint. Indeed, it is going to take a thousand, or maybe ten thousand. Experiment to find out what it takes for your particular equation.

Inflation 2 Equation in Historical Simulation
 Unemployment Rate Set to 4, 5, 6



It is often difficult to get newcomers to model building to apply soft constraints because they tend to have some sort of feeling that “the computer knows best” or that it isn’t “scientific” to provide information and understanding that is not in the database. The considerable improvement in the simulation in this case should provide some assurance that thinking about equations and imposing requirements based on those thoughts can make our models more capable of imitating economic reality.

6. Inflation III — Accounting for External Shocks

If we look closely at the fit of this last inflation equation, we see that the biggest problems arise in the 1976 to 1980 period, the time of the second “oil shock.” At the time, journalists and “the man in the street” were blaming the whole of inflation on the oil shock. We have seen that we can explain the broad outlines of inflation without mention of oil prices, but it is at least worth an experiment to see if we “imported” the high inflation of those years by a faster rise in the price of imported goods than in the price of comparable domestic goods. For “comparable” domestic goods, we will use U.S. exports. While we obviously do not export the same goods that we import, both aggregates involve tradable goods and services. The measure of the relative prices we will use is calculated by the line

```
fex relpri = 100.*@log((m/m$)/(x/x$))
```

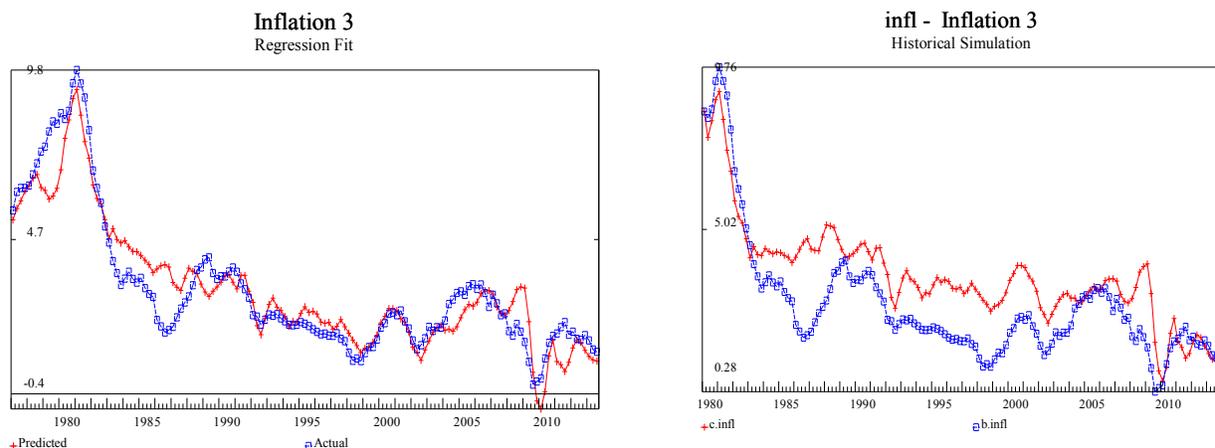
and its rate of inflation, which we can call *imflimp* for “inflation imported” is

```
f inflimp = relpri - relpri[4]
```

We have used the official deflated values of imports and exports, because their deflators reflect the specific prices of the goods traded, whereas our *xR* and *mR* are both deflated by the same deflator. We will define *relpri* by a “fex” rather than by an “f” because we want to think of this variable as exogenous. A littler experimentation with the lags led to the equation

```
r infl = inflex, u, u[1], u[2], u[3], inflimp[2], inflimp[3]
```

The graph of the fit, on the left, looks very good; there are only two or three periods of noticeable errors. The *inflimp* variable helped explain the drop in the inflation rate in 1997, and the rise in inflation in 2001. Though on the low side, the historical simulation on the right shows a further improvement in the last several years.



Inflation 3							
SEE =	0.74	RSQ =	0.8731	RHO =	0.90	Obser =	147 from 1977.100
SEE+1 =	0.33	RBSQ =	0.8667	DW =	0.20	DoFree =	139 to 2013.300
MAPE =	24.57						
Variable name	Reg-Coeff	Mexval	Elas	NorRes	Mean	Beta	
0 infl	- - - -	- - - -	- - - -	- - - -	3.17	- - -	
1 intercept	1.19644	6.2	0.38	37.11	1.00		
2 inflex	0.90842	493.0	1.00	1.90	3.48	0.841	
3 u	0.09651	0.0	0.20	1.60	6.45	0.074	
4 u[1]	-0.55774	0.4	-1.14	1.57	6.45	-0.427	
5 u[2]	0.22366	0.1	0.46	1.57	6.45	0.171	
6 u[3]	0.03875	0.0	0.08	1.57	6.45	0.030	
7 inflimp[2]	0.06240	1.6	0.01	1.09	0.52	0.109	
8 inflimp[3]	0.10449	4.4	0.02	1.00	0.52	0.182	

There is, however, an anomaly in the numerical results of all of the equations we have used so far. We turn now to correcting it.

7. Inflation IV: Using Distributed Lags

Look at the coefficients for the *u* variables in the last regression. The first is positive, the opposite one would expect; the second is *negative*; the third is positive again; and the fourth is again negative. That is certainly not what I had in mind when I put in several lagged values of unemployment. I just thought it might take some time for prices to respond to unemployment. I had no expectation that the direction of causality should reverse in the middle and must confess that I would be very uncomfortable forecasting or simulating the economy with such an equation. As with the coefficient on *inflex*, I expect that the plausibility of the results of our model would be much enhanced by smoothing out those three coefficients. By “smoothing out” I mean making them lie

more nearly on a straight line. If they lay exactly on a straight line then we would have

$$a_3 - a_4 = a_4 - a_5$$

or

$$0 = a_3 - 2a_4 + a_5$$

If a_6 is also on that line, then

$$0 = a_4 - 2a_5 + a_6.$$

And if that line passes through zero one period further out, then

$$a_6 = .5a_5$$

or

$$0 = a_5 - 2a_6$$

We can impose these constraints with the “con” command previously introduced, thus

$$\text{con } 2 \ 0 = a_3 - 2a_4 + a_5$$

$$\text{con } 2 \ 0 = a_4 - 2a_5 + a_6$$

$$\text{con } 2 \ 0 = a_5 - 2a_6$$

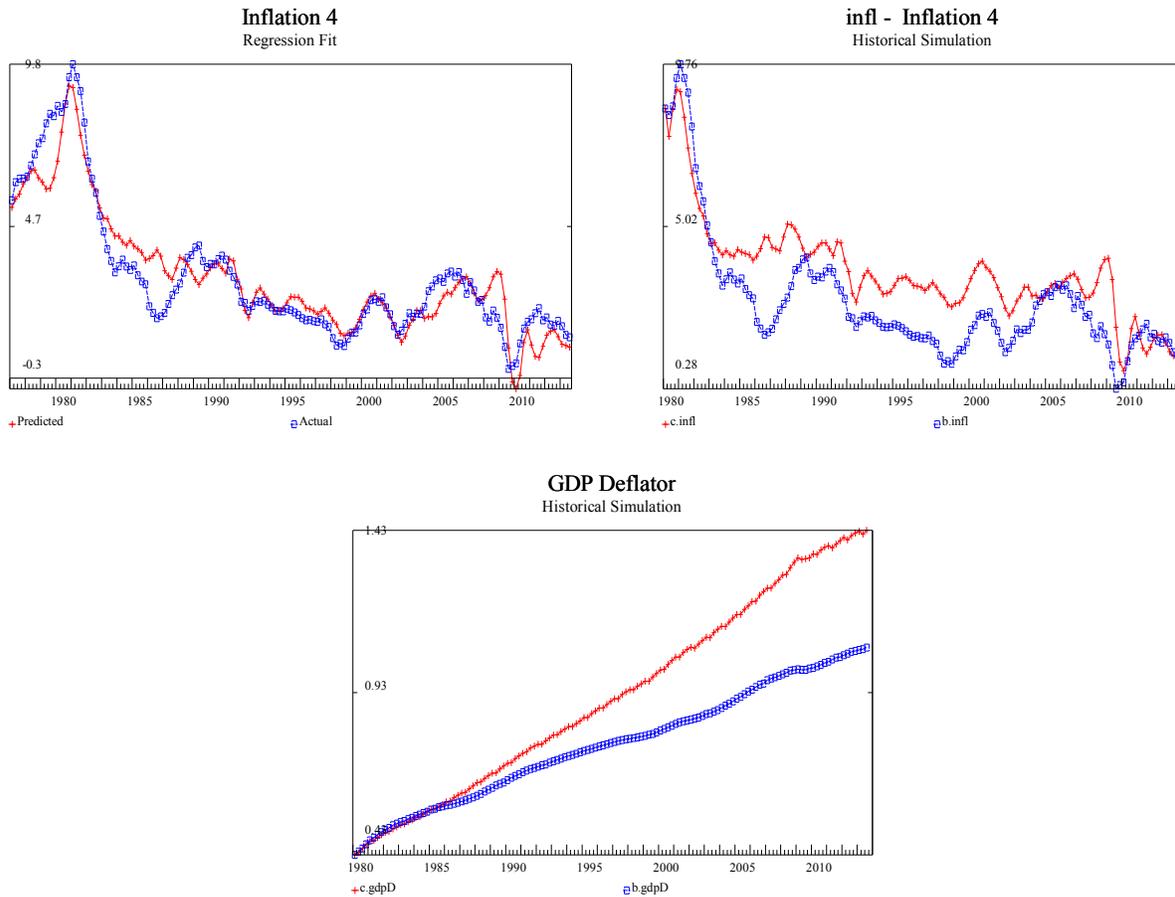
This use of the “con” command in this way to impose a smooth pattern on coefficients is quite common. But it frequently happens that we have a number of lagged values of some variable, perhaps five or ten, and we want to impose smoothness over all their estimated coefficients. For example, in Chapter 1, the first regression we looked at (with vfr as dependent variable) had 12 lagged values of d , with coefficients a_3 through a_{14} . If you look closely at these coefficients, you will see some implausible jumps from one to another. If we wish to smooth them out, we could impose constraints like the following:

```
con 100 0 = a3 - 2a4 + a5
con 100 0 = a4 - 2a5 + a6
con 100 0 = a5 - 2a6 + a7
con 100 0 = a6 - 2a7 + a8
con 100 0 = a7 - 2a8 + a9
con 100 0 = a8 - 2a9 + a10
con 100 0 = a9 - 2a10 + a11
con 100 0 = a10 - 2a11 + a12
con 100 0 = a11 - 2a12 + a13
con 100 0 = a12 - 2a13 + a14
con 100 0 = a13 - 2a14
```

where it is assumed that a_{15} is zero.

Now it is a bit tiresome to type all of these lines, so *G7* provides a shorthand, the “sma” command:

```
sma 100 a3 a14 1
```

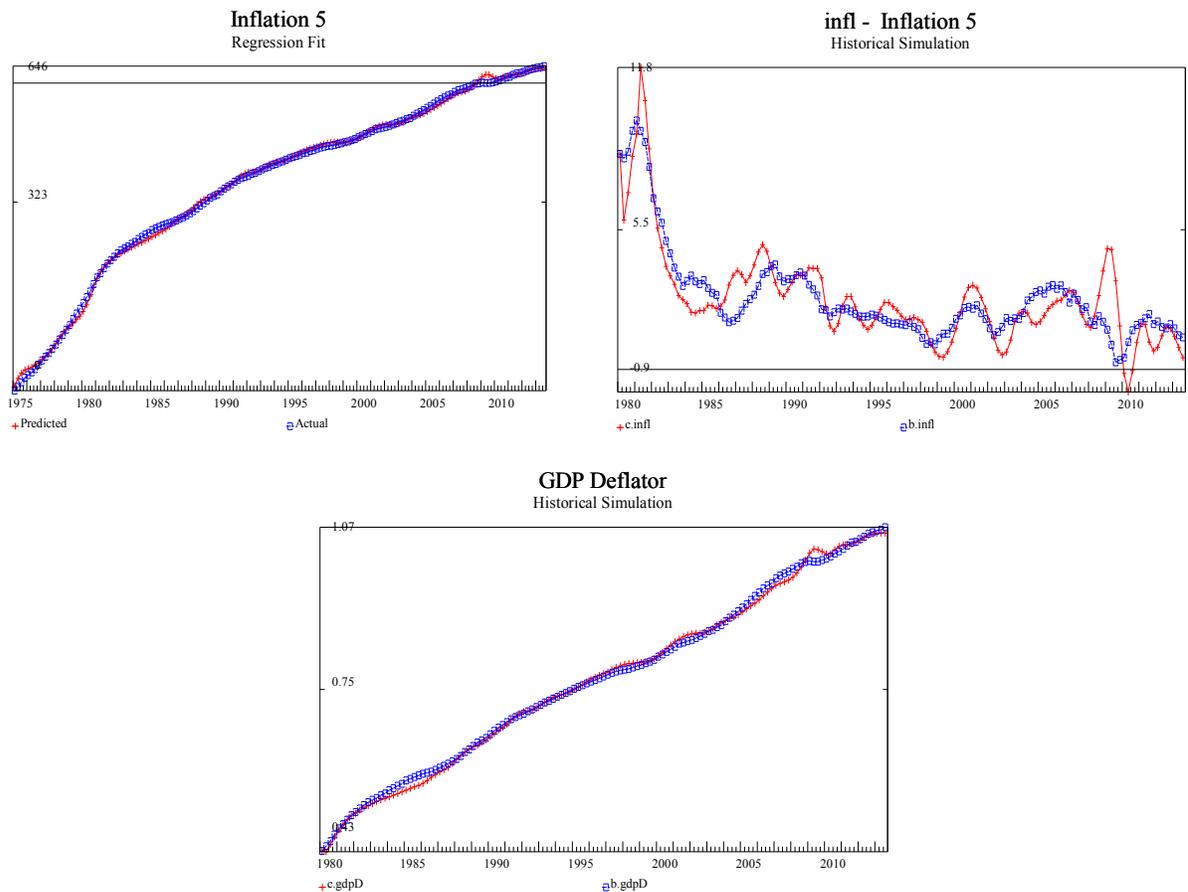



The model needs at various points the price level, $gdpD$, as well as the inflation rate. We work back from the inflation rate to the price level by the equations

$$\begin{aligned}
 f \text{ lgdpD} &= \text{lgdpD}[4] + \text{infl} \\
 f \text{ gdpD} &= @\exp(.01 * \text{lgdpD})
 \end{aligned}$$

The historical simulation for $gdpD$ is shown in the third graph of the group just above. In the present case, this simulation is acceptable only until about 1987, after which point inflation is too high, and $gdpD$ drifts upward above the actual deflator value. It should be obvious that the procedure we have used thus far can produce results for the historically simulated $gdpD$ which deviate steadily more and more from the historical value.

This deflator is as important to us as is the rate of inflation, indeed, perhaps more so. For it is what we shall use to convert $gdpR$ to gdp . And it is the ratio of gdp (in current prices) to some measures of the money supply which will indicate credit conditions. Thus, a systematic error in $gdpD$ is a cause of considerable concern.



The historical simulation results are shown above. The historical simulation of the inflation variable *infl* is a bit too cyclical, but manages to track the level of inflation correctly. The historical simulation of the deflator, *gdpD* is now quite good and this forecast is clearly superior to the previous ones. This last lesson is an important one. It matters whether we work in levels or differences. In general, my experience is that it is best to work at the level of the most “integrated” concept that we want to use, in our case, the levels of prices, not the rate of inflation. The trouble with working at the “differenced” level, inflation in our case, is that if we miss on, say, the low side a few times, then the cumulated function gets on the low side and many periods may go by before it catches up.

Exercise

- 5.1. Include in your current version of the *AMI* model an equation for the price level. First test the equation by itself by defining unemployment with a “fex” command. Then replace the “fex” with just “f” so that unemployment becomes endogenous. Then rerun your experiments with increasing *gR* and watch what happens to inflation.

9. Ways to Improve *AMI*

Addition of the inflation equation to *AMI* began a process of making the model more comprehensive. You can now carry that process further in many directions. Here are some possibilities.

Improve the Investment Equations.

The graphs of the historical simulations of *AMI* showed large errors in the simulation of total fixed investment. The effects of these errors on the rest of the model can be seen by running a historical simulation with the statement “skip νfR ” in the .fix file. This statement will cause the model to use the actual value of νfR rather than the equation’s value. The improvement in the model’s performance is striking. The first step in improving the fixed investment equation is certainly to make use of the distinction which the national accounts offer among three different types of fixed investment: (1) Residential investment (2) Nonresidential structures and (3) Producers’ durable equipment. Residential investment should depend heavily on personal disposable income, interest rates, and the stock of housing created by a bucket. Do not be surprised if this stock of housing has a negative sign. The negative coefficient would simply mean that if we have housing, we don’t need to build it. Equipment investment can follow very much the pattern of the equation we used in *AMI* for all investment, but with the stock formed from only equipment investment and with a faster spill rate than was used for total investment.. Nonresidential structures are apt to be quite a problem, probably because of their great diversity, from electric power plants and oil pipelines to churches and swimming pools.

Improve the Savings Function.

AMI uses exogenous behavioral ratios for Personal savings, Interest paid by persons to business, and Personal transfers to foreigners. That approach implies that these ratios are unaffected by, say, the interest rate, a most unlikely result. Now is a good time to replace the exogenous ratios by economically meaningful relations.

The identity relating income, consumption, interest payments, and savings points out that an increase in interest payments relative to income can come out of either consumption or savings. Thus, the best place to begin work may be on these interest payments. Interest rates, automobile and other durable purchases would be good candidates for explanatory variables, but clearly you do not expect an additive relationship nor one involving just current values of interest rates and durable purchases. With interest payments explained, the savings ratio would be the next in line. Here also interest rates might be expected to play a role, along with the ratio of $piipcb$ to $pidis$. It has also been suggested that the ratio of the contributions to social insurance (Social Security) to disposable income might play a role. You may also find that automobile purchases in the current quarter help explain savings, for the accounts consider that you totally consume the car the moment you buy it. Thus, a quarter of unusually high automobile expenditures might well prove to be one of low savings. But if you use automobile purchases ($cdmvR$), you will need to add an equation to explain it.

The remaining variable in the income-consumption-savings identity, personal transfers to foreigners

(*pipttf*), is small and could be left as an exogenous ratio.

Expand the financial sector.

So far, money supply has played no role in the model and interest rates, if used, have remained exogenous. The monetary policy variable, however, is not interest rates — which are strongly influenced by inflation — but some measure of the money supply. The Quip data bank has several measures of money supply including M1 and M2. They are natural exogenous variables for the model, though exact modeling of them from the variable the Federal Reserve's open market operations control directly, the unborrowed reserves, is itself a tricky matter. Begin your study here by graphing the ratios of each of these measures of money supply to GDP. With the results of those graphs clearly in mind, develop an equation for the rate on 90-day Treasury bills, *rtb*. Remember that expected inflation plays a big role in determining interest rates.

Expand the Government Sector

The government sector in *AMI* is both highly aggregated and incomplete. Section 3 of the NIPA give the accounts of Federal and of State and local governments separately. Include these accounts in the model so that it can calculate the deficit of the Federal government and the surplus or deficit of the State and local governments. You will probably want to leave many of the expenditures exogenous in constant prices; interest payments will certainly depend upon accumulated deficits and interest rates. Likewise, unemployment insurance costs depend on the level of economic activity. On the revenue side, you may want to make tax rates exogenous while the tax bases are endogenous. *AMI's* one *taxrate* variable is an example, but you should have a number of rates for different taxes.

Improve the Foreign Sector

You can add an equation for exports as a function of foreign demand, *fgndem* in Quip. This variable is constructed as a weighted average of total real imports of our trading partners. By looking at our exports relative to it, we can judge whether we are gaining or losing shares in world markets. You will find that the change has been considerable. Can you explain it by changes in relative prices? You have in Quip exports in current and constant prices (*x* and *xR*) and the same for imports (*m* and *mR*). You may wish to exclude petroleum from the price index for imports, and you have the necessary material for doing so in Quip. You may also wish to revisit the import equation and put in variables for relative prices. And, of course, if you find these relative price variables valuable, you may want to develop equations for them also.

CHAPTER 6. GOOD ADVICE AND BAD EXAMPLES

1. Good Advice

How can you tell a good equation from a bad one? If there were any single rule, it would not be necessary for *G7* to print out its many statistics and graphs. It could just report the coefficients and give some numerical grade to the equation. Students who themselves are going to receive a silly numerical or alphabetical grade on their equation want to know what is expected. Unfortunately, there are absolutely no absolute answers to that question. Judging an equation requires thought. What *G7* offers is food for thought but is no substitute for thought itself. In this chapter, we shall discuss some ways of thinking about equations.

Since R^2 is used to measure the closeness of fit, a beginner often assumes that R^2 is also the measure of his success in developing an equation. "How high should my R^2 be?" is frequently asked, but there can be no general answer. For some items, say inventory change, .50 may be quite satisfactory; for others, say consumption of food, anything below .98 would be embarrassing. If R^2 alone, then, is inadequate as a guide, what else should be considered in judging the quality of an equation? Here are some things a good equation should do.

1. **Account for important influences.** If the equation is for food consumption, it should certainly account for the influences of income and the price of food relative to other commodities. On the other hand, if it fails to include the percentage of women in the labor force, or the stock of home appliances, or the stock of housing, or the price of electricity and gas, it can hardly be seriously faulted. In fact, if one of these other factors turned out to be crucial, one should be suspicious.

2. **Economize on variables.** In forecasting, every independent variable must either be produced in the model or must be exogenously forecasted. Be sparing, therefore, in the use of special-purpose variables. Especially avoid variables whose forecasting is tantamount to forecasting the dependent variable. For example, the Federal Reserve discount rate would prove a very good variable for explaining the Treasury bill rate in the *QUEST* model. Since the model does not, however, otherwise explain the discount rate, an exogenous forecast of it would be required, and making an exogenous forecast of the discount is just about the same thing as forecasting the T-bill rate. Do not clutter up the equation with numerous variables of small marginal explanatory value (*mexval*), for each unnecessary variable reduces the efficiency with which the coefficients on the others are estimated. Remember Occam's razor, and prefer a simple equation to a complicated one. On the other hand, if there are strong reasons for believing that a variable should enter the equation and if in fact it does so and with a plausible coefficient, do not throw it out just because its *mexval* is low or its *t*-statistic is less than 2.0 in absolute value.

3. **Avoid false exogeneity.** You can very easily eliminate the excessive cycles in *AMI* by making investment depend on changes in real National income instead of changes in real Gross private product. Why? Because real National income is not otherwise in the model, so it is technically an exogenous variable in the model. Consequently, when the model is run in historical simulation it will always have exactly the historical values of National income, and the predicted values for investment will be those found when the equation was fit. But National income is closely related to

GDP and cannot rightfully be considered exogenous. If you run an experiment by increasing, say, exports, you will get no impact on investment because the exogenous National income is not affected although GDP is affected. In the logic of the national accounts, National income should also be affected, but because you have left it exogenous, it is not. When it comes to forecasting, you will have to produce an exogenous forecast of National income. If this forecast turns out to be out of line with the GDP forecast, you will have a problem. All these problems arise because of what I call the false exogeneity of National income. Avoid all such false exogeneity. Limit the exogenous variables to policy variables, such as tax rates or government expenditures, or variables truly unaffected by short-run economic fluctuations, such as population.

4. **Use appropriate dimensions.** If the dependent variable is, say, food expenditure per capita, then it would be inappropriate to use total disposable income to explain it. Total income might grow because total population was growing but without any change in income per capita to increase food purchases per capita. Similarly, one would not use the total population aged 16-25 in this equation, though the fraction of the population in that age range might be quite appropriate. Or again, the index of food prices would be inappropriate -- because food is not the only thing whose price has gone up -- but the ratio of food prices to an overall price index could be used. Failure to get appropriate dimensions is a common problem with student regressions.

5. **Mix trend and stationary variables only with utmost caution.** Roughly speaking, a trend variable is one which, you know without plotting, would, if plotted, go from the lower left to the upper right of the screen. GDP, population, the consumer price index, and the quantity of money are familiar examples. A stationary series, again roughly speaking, is one where you cannot count on such a trend. The growth rate of GDP or quantity of money, an interest rate, the price of imports relative to domestic goods, the price of gasoline relative to the price of shoes, or rainfall or mean temperature for the year are typical examples. If a stationary variable, say the interest rate, is used to explain a trended variable, say the level of aggregate investment, then one has built into the very form of the equation a declining relative importance for the interest rate as the economy -- and with it, aggregate investment -- grows. Conversely, if one tries to explain a stationary variable, say an interest rate, with a trended variable, say the quantity of money, one is clearly looking for a relation which cannot exist. This simple fact is the valid point behind the recent vogue for studying "cointegration".

6. **Avoid "umbrella" variables.** Once I was criticizing the equation of a student who had used residential construction to explain interest rates. High levels of residential construction seemed to cause interest rates to fall. When the student finally understood my objections, he exclaimed, "Oh, I see. It is as if I had used the number of people carrying umbrellas to explain rainfall." Exactly. The independent variables should "cause" the dependent variable, not vice-versa. While the question of what "cause" means in economics is not always clear, a little common sense applied to eliminating such "umbrella" variables will pay good dividends.

7. **Allow for lags as necessary.** Any relevant variable may act with a lag or a distributed lag. Those lags should be looked for, and if found, the lag distribution should be encouraged, via buckets or soft constraints, to take on a sensible shape. On the other hand, lagged values of the dependent variable should not be used to express lags, since their use will generally distort the parameters on other

variables and lead to excessively long lags.

8. **Have plausible parameter values.** Try your best to interpret every parameter in intuitive terms. Of every regression coefficient (except perhaps the intercept) one should ask "Does that value for this coefficient make sense?" If the answer is "no," there may be a case for using soft constraints. For some equations, it is the elasticities which should make sense. To continue the food example, one would expect the income elasticity to be positive but definitely less than 1.0, while the price elasticity should be negative and probably less than 1.0 in absolute value. (In other equations -- and especially for variables with means close to zero -- the elasticities have no intuitive value.)

9. **Fit adequately.** Look at the plot of the regression. Has the equation explained the trend and the major variations from the trend? An equation that misses by a wide margin on the same side for years at a time is missing something important. Try to think what might be the cause and to find a variable to express that cause. Note that this point comes near the end of the list, not at the top.

10. **Show stability of the coefficients when the sample period is changed.** One way to check for this stability is to try the equation with a test period of two or three years at the end of the fit period. Another is through the use of the "recur" command in *G7*. If the coefficients change significantly when the period of fit is changed, then they probably do not reflect a true structure but just an accident of the period you are looking at. Both of these techniques are illustrated in the next section. The *lever* variable should also be checked mainly to spot spurious observations.

Note that the first six points can be and should be taken care of before you ever try to estimate the equation. They are the most important for the functioning of the model.

Until these criteria become second nature, you should use them as a checklist for each equation in your model.

2. Bad Examples

Since the *QUEST* model contains numerous examples of more or less good equations, we will take the opposite approach here and show some examples of equations that are definitely bad. Except for the last, we should not immortalize these bad examples with beautiful graphs, so I will just show the setup of the equations and the numerical results.

Consumption depends on population, so we could do the following:

```
title Consumption and Population
r c = pop
```

with these results:

consumption go up exactly one percent when population goes up one percent and per capita income is unchanged. We can impose that requirement by working with *per capita* income. Does this equation do the trick?

```

title Consumption and Income per Capita
f ypc$ = pidis$/pop
r c$ = ypc$
:
SEE = 151.92 RSQ = 0.9951 RHO = 0.81 Obser = 135 from 1980.100
SEE+1 = 93.15 RBSQ = 0.9951 DW = 0.39 DoFree = 133 to 2013.300
MAPE = 1.98
Variable name Reg-Coeff Mexval Elas NorRes Mean Beta
0 c$ - - - - - - - - - - - - - - - - 7213.87 - - -
1 intercept -4325.19359 434.6 -0.60 204.20 1.00
2 ypc$ 0.39887 1329.0 1.60 1.00 28929.12 0.998

```

No, it does not. It has violated the "appropriate dimensions" maxim again because it is relating *total* consumption to *per capita* income. Here is the equation we should have estimated.

```

title Consumption per Capita and Income per Capita
f cpc$ = c$/pop
r cpc$ = ypc$
:
SEE = 306.22 RSQ = 0.9967 RHO = 0.69 Obser = 135 from 1980.100
SEE+1 = 223.43 RBSQ = 0.9967 DW = 0.63 DoFree = 133 to 2013.300
MAPE = 0.91
Variable name Reg-Coeff Mexval Elas NorRes Mean Beta
0 cpc$ - - - - - - - - - - - - - - - - 26081.09 - - -
1 intercept -2486.40441 80.1 -0.10 307.52 1.00
2 ypc$ 0.98750 1653.6 1.10 1.00 28929.12 0.998

```

The elasticity of consumption with respect to population -- with per capita income constant -- is 1.0 by definition with this equation. The elasticity of consumption per capita with respect to income per capita turns out to be somewhat greater than 1.0. That is because the equation mistakenly attributes the secular decline in the savings rate in the United States to increasing income. The decline had other causes, which we will discuss in connection with the *QUEST* model in Part 2.

Although this last equation passes the maxims of this chapter fairly well, it probably suffers from more subtle problems. If, for some reason, consumption is higher in a particular quarter than it "should" have been according to the equation, then -- because consumption influences income -- income is probably unusually high in that quarter. Thus, the error term in this equation is probably correlated with the independent variable. This problem is referred to as "simultaneous equation bias" or sometimes (less accurately) as "least squares bias." What to do about it is dealt with in Part 2.

Here is an example adapted only slightly from a recent major work of a major economist. It begins from the idea that consumption expenditures per capita on automobiles depends on income per capita and the price of automobiles. The price of automobiles is expressed by taking the ratio of automobile purchases in current prices to the same thing in constant prices. The example goes as follows:

```

f pauto = cdmv/cdmv$ # price of automobiles
f cdmvpc$ = cdmv$/pop
f ypc$ = pidis$/pop
r cdmvpc$ = ypc$, pauto

:
Consumption per Capita of Automobiles
SEE = 120.29 RSQ = 0.7030 RHO = 0.90 Obser = 135 from 1980.100
SEE+1 = 53.52 RBSQ = 0.6985 DW = 0.21 DoFree = 132 to 2013.300
MAPE = 10.28
Variable name Reg-Coeff Mexval Elas NorRes Mean Beta
0 cdmvpc$ - - - - - 1029.50 - - -
1 intercept -95.80447 0.6 -0.09 3.37 1.00
2 ypc$ 0.02334 12.6 0.66 1.07 28929.12 0.574
3 pauto 478.52469 3.3 0.44 1.00 0.94 0.289

```

Both variables turn out to be highly significant and the income elasticity is 0.66, a reasonable value. However, the elasticity on *pauto* comes out to positive 0.44, which is the wrong sign. How can that be? Do you see what the problem is without reading further?

The problem lies, of course, with the *pauto* variable. As calculated, it is an index of automobile prices, not the price of automobiles relative to other competing goods. The variable which should have been used was

```
f pauto = (cdmv/cdmv$)/(c/c$)
```

Try the regression with this variable. You should be quite unhappy with the results, for they actually show an even larger positive price elasticity. There are many other things to consider with regard to automobile consumption. I'll not pursue this equation further here since a full treatment is given later with the *QUEST* model in Part 2. Suffice it here to emphasize the importance of thinking about the whole structure of the equation and considering the plausibility of the magnitudes of each coefficient, rather than just looking at the correctness of the signs on the coefficients.

There are a number of ways of fooling yourself into thinking that you have a good equation when, in fact, what you have is trash. Here is one way. We need an equation for construction of non-residential structures, *vfhrsR*, and we suppose that it might be proportional to the difference between need for such structures, which we take to be proportional to real GDP, *gdpR*, and the stock of such structures. So we make up an equation like this:

```

ti vfhrsR - Non-residential Structures
lim 1980.1 2013.3
f ub01 = @cum(ub01,1.,.01)
# Make up stock of structures
f StockSt = @cum(cumSt,vfhrsR,.01)/ub01
r vfhrsR = gdpR,StockSt

```

It doesn't work very well, so let's try lagging the stock of structures one quarter with this regression command:

```
r vfhrsR = gdpR,StockSt,StockSt[1]
```

Now we get the following wonderful results:

The fit is undeniably excellent. The negative sign on M1 lagged and the positive signs on the interest rate should give one pause. But I have often been more or less told by the ingenious and proud creators of such equations, “How do you know what those signs ought to be? Look at this fit!”

Well, one way that often shows that warnings of theoretically wrong signs should have been heeded is to fit the equation over only part of the period and to use the end of the data to test how the equation performs outside the period over which it was fit. Let’s try this with the above equation. We change only the “limits” line to

```
lim 1980.1 1997.4 2013.3
```

The fit over the shorter period is actually slightly better:

```

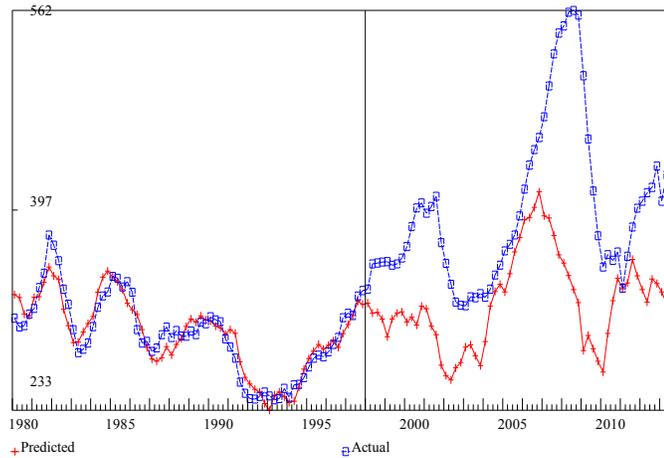
:
          vfnrsR - Non-residential Structures
SEE   =    11.60  RSQ   = 0.8680  RHO =  0.73  Obser =  72  from 1980.100
SEE+1 =     8.16  RBSQ  = 0.8232  DW  =  0.54  DoFree =  53  to  1997.400
MAPE  =     3.35  Test period:  SEE   95.09  MAPE   17.92  end  2013.300
Variable name      Reg-Coeff  Mexval  Elas  NorRes  Mean  Beta
0 vfnrsR          - - - - -  - - - - -  - - - - -  295.92 - - -
1 intercept       120.87993  2.2    0.41  7.58    1.00
2 gdpR[1]         -0.00864  0.8    -0.25  5.73    8414.33 -0.374
3 d[1]            -0.05298  1.8    -0.01  5.73    64.37 -0.100
4 d[2]            -0.00328  0.0    -0.00  5.63    63.06 -0.006
5 d[3]            0.04009  1.3    0.01  5.35    61.22  0.074
6 d[4]            0.06749  3.7    0.01  4.99    60.03  0.125
7 d[5]            0.08636  5.2    0.02  4.65    59.58  0.159
8 d[6]            0.01830  0.3    0.00  4.53    59.28  0.034
9 d[7]            0.00447  0.0    0.00  4.51    59.95  0.008
10 d[8]           0.03291  1.0    0.01  4.48    59.11  0.063
11 d[9]           0.03533  1.2    0.01  4.45    58.49  0.067
12 d[10]          0.01090  0.1    0.00  4.44    58.54  0.021
13 StockSt[1]    0.03075  5.5    0.67  3.74    6475.56  0.416
14 m1R            0.11393  0.4    0.46  1.50    1200.24  0.799
15 m1R[1]        -0.21881  1.8    -0.88  1.41    1193.91 -1.546
16 rtb            1.71196  0.6    0.04  1.29     7.07  0.159
17 rtb[1]         6.50303  7.2    0.16  1.20     7.17  0.611
18 niprfR         0.04511  0.4    0.10  1.02    681.03  0.254
19 niprfR[1]     0.10336  1.2    0.24  1.00    673.56  0.558

```

but the forecast into the test period is not as good, as you can see from the SEE and MAPE in the test period in the table above or even more dramatically in the graph below.

Another way to check the reliability of a regression is through the use of *recursive regression* by G7’s “recur” command. This amounts to running the same regression a number of times with progressively longer periods of fit and observing what happens to the values of the estimated regression coefficients. In G7, we need make only two minor changes in the commands to do recursive regression. Just change

vfnrR - Non-residential Structures



```

lim 1980.1 2013.3
...
r vfnrR = gdpR[1],d[1],d[2],d[3],d[4],d[5],d[6],d[7],
to
lim 1980.1 1997.4 2013.3
...
recur vfnrR = gdpR[1],d[1],d[2],d[3],d[4],d[5],d[6],d[7],

```

The first regression will be over the period 1997.4 to 2013.3, the next over 1997.3 to 2013.3, and so on until the last is over 1980.1 2013.3. The results will pour out on the blue screen, but to make it easier to see what is happening, *G7* enters the values of each coefficient as time series in its workspace bank. These series are named *b1*, *b2*, *b3*, etc., up to the number of variables in the regression. The value of a particular *b* in a particular period is the value of that regression coefficient in the regression beginning in that period. Thus, *b1* for 1997.4 is the value of the intercept in the regression from 1997.4 to 2013.3. Also entered into the workspace are the standard deviations of the coefficients, *s1*, *s2*, *s3*, etc., as computed by the standard formulas.

The results are best viewed by graphing each coefficient and two standard deviations on either side. Here is a *G7* command file which will do the job for the present example.

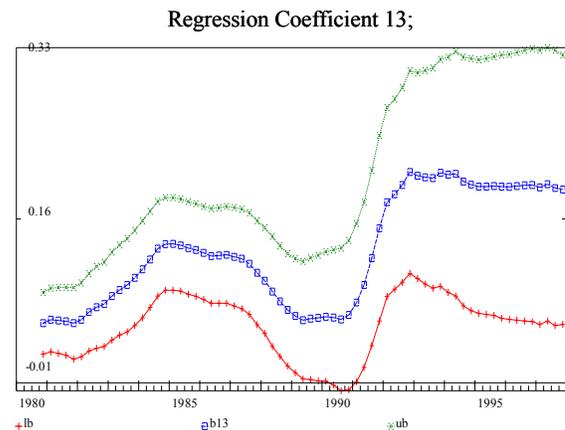
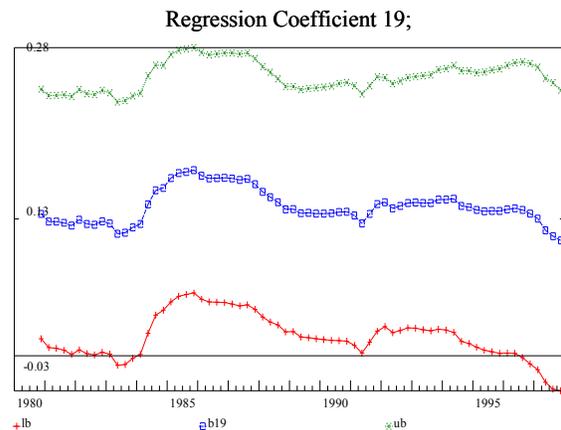
```

do {f lb = b%1 -2*s%1;
f ub = b%1 +2*s%1;
ti Regression Coefficient %1;
gname coef%1;
gr lb b%1 ub} (1-19)

```

This command file uses the *do* command of *G7* to repeat the same commands with substitution of text; it is fully described in the *G7* help files and need not detain us here.

What one would hope to find in these graphs is something not worse than the graph on the left below for coefficient 19. Nearly all of the estimated values, the middle line, are within two standard deviations of any other value on that line, by standard deviations calculated for any period. If all of the coefficients had graphs like that of number 19 (the coefficient for *niprfR[1]*), we could say that the recursive regression examination had been passed satisfactorily.



Unfortunately, that is by no means the case. A much more typical result is that for Coefficient 13 for the stock of structures. Estimation beginning in 1981.1 gives a very low coefficient; estimation beginning a few quarters sooner or later gives a much larger coefficient. “How can that be?” you may well ask. The answer can only be that there was no stable structure of the kind posited by this regression. Thus, this equation fails examination by recursive regression and thereby illustrates how the method can perhaps save us from using equations which would make embarrassing forecasts.

Next, let us look at a problem of mixing trended and stationary variables. We can see it in the problem of relating imports to domestic demand and exchange rates. Here is one approach using a "terms of trade" (*tot*) variable defined as the ratio of the prices of imported goods relative to the prices of domestically produced goods.

```
ti m$ -- Imports
" Define Domestic Demand
f dd$ = c$ + v$ + g$ + x$
" define terms of trade as ratio of foreign to domestic prices.
f tot = (m/m$) / (gnp/gnp$)
r m$ = dd$, tot
```

Over a long period, Domestic Demand (*dd\$*) and Imports (*m\$*) both grow substantially. The terms of trade variable (*tot*), however, is essentially a relative price. It is unlikely to have a long-term, secular trend. Consider a future year when *dd\$* will have doubled, and, let us say, *m\$* has tripled, but *tot* is about the same then as now. Will a change of, say .1, change *m\$* by the same dollar amount then as now? The form of the equation requires an answer of "yes", but I think you will agree that the correct answer should be "No, the effect will be larger." The problem is that we have mixed stationary and trend variables.

One approach to the problem just posed is to regress the logarithm of *m\$* on those of *dd\$* and *tot*. Then the elasticity of imports with respect to terms of trade will be constant. But we will have only shifted the problem. If the elasticity of imports with respect to domestic demand is greater than 1.0 -- which is likely -- then with sufficient growth in domestic demand, imports will exceed domestic

demand and domestic production will have become negative. Here we have a case of what I call *asymptotic idiocy*. One detects it by asking what happens to the equation as the variables move far along the trends they are likely to follow. Actually, trend-stationary problems are just a special case of it.

My long involvement in long-term forecasting has perhaps made me overly sensitive to the asymptotic idiocy problem. It is certainly defensible to use for a short forecast an equation which is asymptotically idiotic, but it does seem to me that one should be aware of the problem.

Unfortunately, avoiding asymptotically idiotic equations often requires non-linear equations with coefficients which are laborious to estimate. We will see a method for doing so in Part 2.

If you are working on the improvement of *AMI* along the lines suggested at the end of Chapter 4, you already have a number of equations to which to apply the ten-point check list of this chapter.

3. The Worst Example

A lagged value of the dependent variable is frequently used as an independent variable not just by greenhorn students of modeling but also by serious professionals who should know better. Such equations are, for reasons I will explain, exercises in self deception. They are, however, frequently excused with a comment to the effect that they allow for a distributed lag. Now allowing for a distributed lag is, as we have seen, frequently a good thing to do. We therefore need to see first why the lagged value of the dependent variable might possibly be a way to include a distributed lag and then why it usually does something quite different.

Let us suppose, then, that y depends upon infinitely many prior values of x with exponentially decreasing weights. The true equation relating y and x is then supposed to be

$$(1) \quad y_t = a + b \sum_{i=0}^{\infty} \lambda^i x_{t-i} + \epsilon_t$$

where $0 < \lambda < 1$. At first glance, this equation seems to call for a regression of y on an infinity of prior values of x . But the assumption of exponentially declining weights allows us to perform a trick with the equation. First we lag the equation once and then multiply both sides by λ to get

$$(2) \quad \lambda y_{t-1} = \lambda a + b \sum_{i=1}^{\infty} \lambda^i x_{t-i} + \lambda \epsilon_t$$

Notice that the infinite summation on the right side of the two equations is the same except that the summation in equation (2) lacks the first term of the summation in equation (1). Subtracting (2) from (1) therefore gives

$$(3) \quad y_t - \lambda y_{t-1} = a(1 - \lambda) + b x_t + \epsilon_t - \lambda \epsilon_{t-1}$$

which we rewrite in a form to be estimated as

$$(4) \quad y_t = b_1 + b_2 x_t + b_3 y_{t-1} + \epsilon_t - \lambda \epsilon_{t-1}$$

We can now estimate equation (4) without the slightest concern about an infinite number of independent variables. The estimate of b_3 is the estimate of λ . More flexible forms of lags can be obtained by including more than one lagged value of the dependent variable. One should note, of course, that if the error terms in equation (1) are independent, then the ones in equation (4) are autocorrelated and correlated with one of the independent variables, namely y_{t-1} .

Now this transformation is indeed a clever trick and has mesmerized many model builders into a rather indiscriminate use of lagged values of the dependent variable to account for lags. The problem is that the equation with the lagged dependent variable may fit very well for reasons totally unrelated to the supposed distributed lag on one particular variable. Namely, the dependent variable may well depend upon many factors which change slowly relative to the period of observation. The lagged value of the dependent variable sums up perfectly the effects of all these factors for one period earlier. But, since the factors change slowly, that summary is still an excellent representation of their values in the present period. Thus, the lagged value of the dependent variable will explain most of the variation in the dependent variable and leave only a little noise to identify the coefficients of the other variables in the regression.

Now equations with lagged values of the dependent variable offer an alternative way of calculating the predicted value. Namely, instead of using the actual lagged value of the dependent variable in computing the predicted value, we can use the value predicted for the first period to calculate the prediction for the second period, and then use the predicted value for the second period to calculate the prediction for the third, and so on. We will call the variable computed in this way *BasePred*, because it uses the actual lagged value only in the base period. If G detects the lagged value of the dependent variable, it automatically computes this *BasePred* and graphs it as a third line when the “*gr*” command is given. If the above analysis is applicable to your equation, the fit of the ordinary predicted value should be very close, while that of *BasePred* may be anywhere from so-so to dreadful. Let us look at two examples.

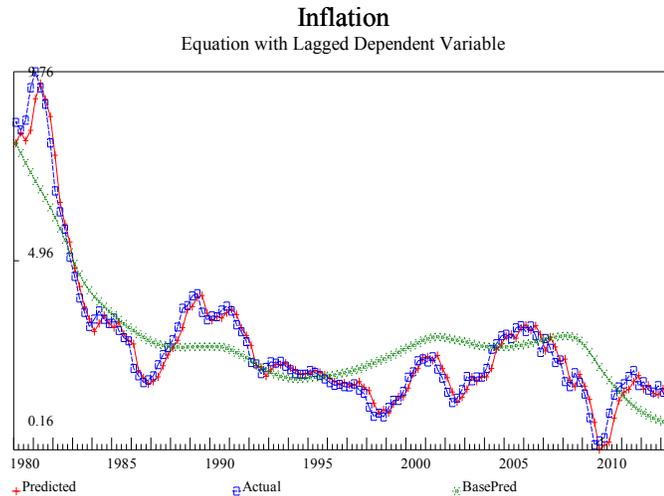
In the first, we regress inflation on its own lagged value and the unemployment rate. The results are shown below.²

2 If the lagged value of the dependent variable is included among the independent variables, the Durbin-Watson statistic is invalid in the best of circumstances. Durbin devised another statistic for this situation known as the Durbin H. If the error terms are uncorrelated, the distribution of H approaches the normal with mean zero and unit standard deviation as the sample size grows. When G detects the presence of the lagged value of the dependent variable among the independent variables, it automatically replaces the Durbin Watson with the Durbin H.

```

:
                                Inflation
SEE   =      0.29  RSQ   = 0.9754  RHO =   0.48  Obser = 135 from 1980.100
SEE+1 =      0.25  RBSQ  = 0.9750  DurH =   5.61  DoFree = 132 to 2013.300
MAPE  =      11.03
Variable name      Reg-Coeff  Mexval  Elas  NorRes  Mean  Beta
0 infl            - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
1 intercept       0.31715   3.6    0.11  40.68   1.00
2 u               -0.04320  2.9   -0.10  39.16   6.46 -0.039
3 infl[1]         0.96914  525.7  0.99   1.00   2.88  0.996

```



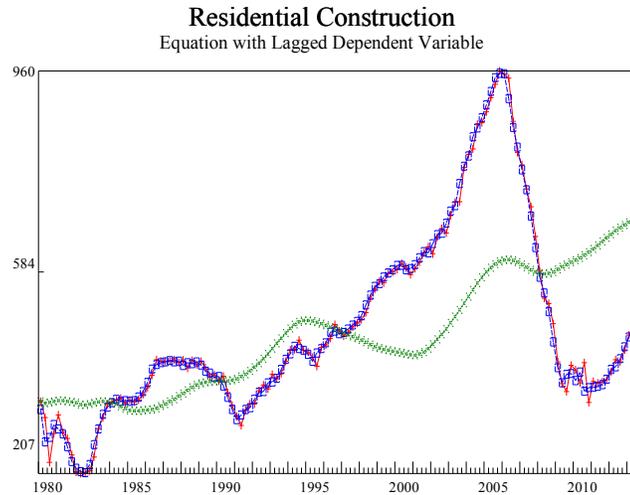
As you can see, the fit of the equation as normally calculated is fantastic; the fit of the *BasePred* line, however, shows that the equation is worthless for forecasting beyond a few periods. It has failed to identify the connection between unemployment and changes in inflation which our earlier equations identified fairly well.

Another example is given by regressing Residential Construction on the real interest rate and two lagged values of the dependent variable. The R^2 is even better than before, and the performance on the *BasePred* test is even more miserable. It doesn't even do well right at the beginning.

```

:
Residential Construction
SEE = 13.71 RSQ = 0.9939 RHO = -0.13 Obser = 135 from 1980.100
SEE+1 = 13.58 RBSQ = 0.9938 DurH = -2.05 DoFree = 131 to 2013.300
MAPE = 2.38
Variable name      Reg-Coeff  Mexval  Elas  NorRes  Mean  Beta
0 vfrR            - - - - - - - - - - - - - - - - - - 478.46 - - -
1 intercept       11.55952   2.7    0.02  164.60  1.00
2 rtbR           -1.12629   1.5   -0.00  138.15  2.03 -0.015
3 vfrR[1]         1.72175  178.1  1.72   2.25  477.37 1.725
4 vfrR[2]        -0.74026  50.0  -0.74   1.00  476.49 -0.743

```



Just because you get a close fit when using a lagged value of the dependent variable, do not suppose that you have arrived at any kind of structural relation between the dependent and independent variables. Possibly you have found such a relation, but much more probably you have simply found that the determinants of the dependent variable change slowly from period to period. Those determinants may or may not have anything to do with the independent variable you have specified. If the dependent variable has a trend, *BasePred* will generally follow that trend, but that following is no evidence of having found a structural relation.

When distributed lags are estimated by direct regression by putting a variable into a “bucket” (as we did in creating *inflex* in the inflation examples of Chapter 5), and the independent variables fail to explain the dependent, we immediately recognize that fact by poor fit. When the lagged value of the dependent variable is slipped into the equation, we no longer have that warning. We can get excellent fits without having found any causal relation between the independent and dependent variables. I, therefore, find the lagged value of the dependent variable deceptive and avoid it. Self-deception, however, is always popular, and the use of the lagged value of the dependent variable has reached epidemic proportions. This use, I believe, is responsible for much poor forecasting and bad policy analysis which has done nothing to engender respect for the economic forecasting profession on the part of the general public.

CHAPTER 7. THE THEORY OF LINEAR SYSTEMS

1. The General Solution of a System of Linear Difference Equations

In Chapter 1, we found that the *AMI* model had a pronounced tendency to cycle. Observers of the economy have often thought they discerned cycles. Indeed, the term “business cycle” has become part of the language. In modeling, our descriptions of the economy do not seem to have anything to do with cycles. Where, then, do the cycles come from? In this short chapter, we shall try to answer that question.

In this first section, we will seek the general solution of a system of difference equations such as can, with some approximation, represent many of our models. It will be an abstract and analytical section. In the following sections, that general solution will enable us see some interesting features of models and, perhaps, of economies. I must emphasize, however, that this chapter is aimed at *understanding qualitatively* the results of models, not at calculations you would ever be likely to want to make. We will find, for example, the *form* of the general solution of a linear model. You generally want, however, a particular solution, and it, as you already know, is calculated directly, without any reference to that general form. We will show how one could, in principal, calculate analytically a frequency response function for a linear model. In practice, in models of any size, it is easier to calculate it from simulations of the model, as is suggested by the exercise at the end of the chapter.

Many of the equations of our models take a form somewhat like the following:

$$(1) \quad y_{t+1} = Ay_t + f_t$$

where y_t is a vector of endogenous variables, f_t is a vector of exogenous variables, and A is a matrix. If some variable, say u , appears on the right with values lagged further back than t , we can easily introduce v defined by

$$(2) \quad v_{t+1} = u_t$$

so that

$$(3) \quad v_t = u_{t-1}$$

and use v_t in place of u_{t-1} while including (2) among the equations. Among the exogenous values, a lagged variable, say $x[4]$, can be simply regarded as the value in period t of x four periods earlier. Thus, the presence of variables with more than one lag does not prevent us from *thinking* of the model as though it were described by the equation (1).

Nonlinearities, however, such as $gdp = gdpR * gdpD$, cannot be described exactly in system (1). They can, however, be represented approximately by the first terms of the Taylor series approximation. Thus, equation (1) comes close enough to describing many models to be worth studying.

Equation (1) is called a system of *linear, first order difference equations*. The term *first order* refers to the fact that at most only one lagged value of any variable appears. In the form (1), the differences are not evident, but if we subtract y_t from both sides, it becomes

$$(4) \quad y_{t+1} - y_t = (A - I)y_t + f_t$$

and the differences on the left side are apparent. (Systems described in continuous time would have the form $\frac{dy(t)}{dt} = Ay(t) + f(t)$, which is a system of linear *differential* equations. The mathematical theories of differential and difference equations are closely parallel, except that for difference equations there is no problem of the existence of a solution, since the equations tell us precisely how to calculate it.)

We will now show that any one of the variables in a system of n first-order equations, such as equation (1), follows an n^{th} - order difference equation not involving the other variables in the system. That is, the value for period $t+n$ can be expressed as a linear function of the values of the same variable in n previous periods and the exogenous variables. This fact will allow us to apply some well-known results about the roots of polynomials to characterize the solutions of the difference equations.

We will at first concentrate on the *homogeneous* part of the system, that is to say, the part of the system obtained by dropping the exogenous variables. Let us begin by writing this homogeneous part of the system (1) out in more explicit form:

$$(5) \quad \begin{pmatrix} y_{1,t+1} \\ y_{2,t+1} \\ y_{3,t+1} \\ \dots \\ y_{n,t+1} \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \dots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \dots & a_{3n} \\ \dots & \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & a_{n3} & \dots & a_{nn} \end{pmatrix} \begin{pmatrix} y_{1t} \\ y_{2t} \\ y_{3t} \\ \dots \\ y_{nt} \end{pmatrix}$$

From this system, we want to derive a single equation for y_t of the form

$$(6) \quad y_{1,t+1} + c_{n-1}y_{1,t+n-1} + \dots + c_0y_{1t} = 0 \quad .$$

First, we now solve the second of these equations for y_{2t} and substitute in all the equations, thus:

$$(7) \quad \begin{pmatrix} y_{1,t+1} \\ y_{2,t+1} \\ y_{3,t+1} \\ \dots \\ y_{n,t+1} \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & \dots & 0 \\ a_{21} - \frac{a_{22}a_{11}}{a_{12}} & \frac{a_{22}}{a_{12}} & a_{23} - \frac{a_{22}a_{13}}{a_{12}} & \dots & a_{2n} - \frac{a_{22}a_{1n}}{a_{12}} \\ a_{31} - \frac{a_{32}a_{11}}{a_{12}} & \frac{a_{32}}{a_{12}} & a_{33} - \frac{a_{32}a_{13}}{a_{12}} & \dots & a_{3n} - \frac{a_{32}a_{1n}}{a_{12}} \\ \dots & \dots & \dots & \dots & \dots \\ a_{n1} - \frac{a_{n2}a_{11}}{a_{12}} & \frac{a_{n2}}{a_{12}} & a_{n3} - \frac{a_{n2}a_{13}}{a_{12}} & \dots & a_{nn} - \frac{a_{n2}a_{1n}}{a_{12}} \end{pmatrix} \begin{pmatrix} y_{1t} \\ y_{1,t+1} \\ y_{3t} \\ \dots \\ y_{nt} \end{pmatrix}$$

The matrix on the right of (7) we shall call B_2 . It may be obtained from the matrix A on the right of (5) by post-multiplication of A by

$$(8) \quad M_2 = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 \\ -a_{11} & 1 & -a_{11} & \dots & -a_{11} \\ a_{12} & a_{12} & a_{12} & \dots & a_{12} \\ 0 & 0 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 1 \end{pmatrix}$$

Thus, .

$$B_2 = AM_2$$

Notice that while we have now eliminated y_2 on the right of (7), it still appears on the left. To continue the process of eliminating y_2 , we use the first equation of (5) advanced by one period (so that it is an equation for $y_{1,t+2}$) to write $y_{1,t+2}$ as a linear combination of the equations (7). The resulting equation for $y_{1,t+2}$ does not contain any reference to y_2 . If we put this equation in place of the second equation in (7), the system will have the form

$$(9) \quad \begin{pmatrix} y_{1,t+1} \\ y_{2,t+1} \\ y_{3,t+1} \\ \dots \\ y_{n,t+1} \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & \dots & 0 \\ d_{21} & d_{22} & d_{23} & \dots & d_{2n} \\ d_{31} & d_{32} & d_{33} & \dots & d_{3n} \\ \dots & \dots & \dots & \dots & \dots \\ d_{n1} & d_{n2} & d_{n3} & \dots & d_{nn} \end{pmatrix} \begin{pmatrix} y_{1t} \\ y_{1,t+1} \\ y_{3t} \\ \dots \\ y_{nt} \end{pmatrix}$$

where the matrix D_2 on the right is derived by premultiplying B_2 by N_2 given by

$$(10) \quad N_2 = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 \\ a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ 0 & 0 & 1 & \dots & 0 \\ 0 & 0 & 0 & \dots & 1 \end{pmatrix}$$

thus,

$$(11) \quad D_2 = N_2 A M_2 \quad .$$

Note that

$$(12) \quad N_2 M_2 = I \quad ;$$

in other words, N_2 and M_2 are inverses of one another.

In (9) we have succeeded in eliminating y_2 from the system. We can now repeat this process to eliminate, one by one, all the variables other than y_1 from the system. When finished, we will have a system of the form

$$(13) \quad \begin{pmatrix} y_{1,t+1} \\ y_{1,t+2} \\ y_{1,t+3} \\ \dots \\ y_{1,t+n} \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 1 \\ -c_0 & -c_1 & -c_2 & \dots & -c_{n-1} \end{pmatrix} \begin{pmatrix} y_{1,t} \\ y_{1,t+1} \\ y_{1,t+2} \\ \dots \\ y_{1,t+n-1} \end{pmatrix}$$

A matrix of the form shown on the right of (13) is called a *Frobenius* matrix, and the method we have just explained for deriving a Frobenius matrix from a given matrix A is called the Danilevsky process. The Frobenius matrix is also called the *companion* matrix.

From the last equation of (13), we read, after bringing all the right-hand side to the left side of the equation,

$$(14) \quad y_{1,t+n} + c_{n-1} y_{1,t+n-1} + c_{n-2} y_{1,t+n-2} + \dots + c_0 y_{1,t} = 0$$

If we now try a solution of (14) of the form

$$(15) \quad y_{1t} = \lambda^t$$

where λ is just a (possibly complex) number and the superscript t denotes raising to a power, then we

see (after dividing both sides of the resulting equation by λ^t) that there will be such a solution if and only if λ is a solution (or root) of the *characteristic polynomial*

$$(16) \quad \lambda^n + c_{n-1}\lambda^{n-1} + c_{n-2}\lambda^{n-2} + \dots + c_0 = 0$$

One may reasonably ask at this point: Would we have gotten a different polynomial in (16) if we had chosen to convert (5) into a single equation involving only values of y_2 instead of y_1 ? The answer is No, but to demonstrate that fact, an alternative interpretation of the transformation process is useful. Let us return to the homogeneous part of equation (5),

$$(17) \quad y_{t+1} = A y_t$$

and ask, under what condition would

$$(18) \quad y_t = v \lambda^t$$

be a solution, where v is a constant vector? Direct substitution in (17) and cancellation of λ^t from both sides shows that the condition is that

$$(19) \quad v \lambda = A v \quad .$$

Values of λ for which (19) holds are called *characteristic values* of the matrix A and the corresponding vectors are called *characteristic vectors*. (The half-German words *eigenvalues* and *eigenvectors* are also in common use in English. One occasionally sees the older *proper* or *secular* values and vectors.) Well, how does one go about finding the characteristic polynomial? You may encounter books that tell you that for (19) to have a non-zero solution, $A - \lambda I$ must be singular, so its determinant must be zero, and its determinant is a polynomial in λ . That is true but useless information, since computing the determinant as a function of λ is an enormous and complicated job by the usual definitions of determinants when n is more than 2 or 3. Is there a simpler way?

There is, and you already know it; but to bring that knowledge into consciousness we need a simple definition and theorem. A matrix A is said to be *similar* to a matrix B if there exists a nonsingular matrix M such that $A = MBM^{-1}$. You can easily show that if A is similar to B , then B is similar to A , so we can say simply that they are similar without bothering over which is similar to which. Moreover, if A and B are similar and B and C are similar, then A and C are similar. The theorem we need is simply: *Similar matrices have the same characteristic roots*. Proof: suppose that $Bv = \lambda v$ and $A = MBM^{-1}$. Then $A(Mv) = MBM^{-1}(Mv) = MBv = M(\lambda v) = \lambda(Mv)$, so that λ is also a characteristic value of A with Mv as the associated characteristic vector.

Now recall that in the Danilevsky method, N_2 and M_2 are inverses of one another, so that D_2 in (11) is similar to A . At each further step in the Danilevsky method, similarity is maintained, so that the Frobenius matrix in (13) is similar to A . Furthermore, λ is a characteristic root of the Frobenius matrix if and only if it satisfies the polynomial equation (16), for in that case

$$(20) \quad \lambda \begin{pmatrix} 1 \\ \lambda \\ \lambda^2 \\ \dots \\ \lambda^{n-1} \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 1 \\ -c_0 & -c_1 & -c_2 & \dots & -c_{n-1} \end{pmatrix} \begin{pmatrix} 1 \\ \lambda \\ \lambda^2 \\ \dots \\ \lambda^{n-1} \end{pmatrix}$$

No two *different* Frobenius matrices are similar to one another, for if they were similar, they would have the same roots and therefore have the same bottom row, and therefore be identical. If we had wanted an equation in y_2 instead of y_1 , we would have begun the Danilevsky process by interchanging the first and second rows and then the first and second columns. You can easily verify that the matrices which accomplish these operations by pre- and post-multiplication are inverses of one another, so that the resulting matrix is similar to the original A . Thus, the final Frobenius matrix is similar to A and thus is the same as the one obtained without interchange of rows and corresponding columns.

We can now bring to bear on the solution of a system of difference equations a few well-known facts about the roots of polynomials. These are:

- (a) An n^{th} degree polynomial $P(\lambda)$ – such as (16) – has n roots $\lambda_1, \lambda_2, \dots, \lambda_n$, and can be factored and written $P(\lambda) = (\lambda - \lambda_1)(\lambda - \lambda_2) \dots (\lambda - \lambda_n)$. If the roots are not all distinct, tiny changes in the coefficients of the polynomial will make them distinct.
- (b) Some or all of the roots may be complex, that is, they may be of the form $a + bi$, where a and b are real numbers and $i^2 = -1$. Any complex roots occur in conjugate pairs: if $a + bi$ is a root, so is $a - bi$.

We will consider only the case in which all the roots are distinct, since, for equations with empirically determined coefficients, that is almost certainly the case. If the roots of A are distinct, the corresponding characteristic vectors, V_1, V_2, \dots, V_n , are linearly independent, for suppose that, to the contrary, one of them, say, V_{r+1} , is a linear combination of r linearly independent ones:

$$(21) \quad V_{r+1} = \sum_{j=1}^r b_j V_j .$$

We could then multiply both sides of (21) by A to obtain

$$(22) \quad \lambda_{r+1} V_{r+1} = \sum_{j=1}^r \lambda_j b_j V_j$$

But we could also simply multiply both sides of (21) by λ_{r+1} and subtract from (22) to get

$$(23) \quad 0 = \sum_{j=1}^r b_j (\lambda_j - \lambda_{r+1}) V_j$$

But since all the $(\lambda_j - \lambda_{r+1}) \neq 0$, equation (23) contradicts the assumption that V_1, \dots, V_r are linearly independent.

We have now found n *different* solutions, $V_1 \lambda_1^t, V_2 \lambda_2^t, \dots, V_n \lambda_n^t$, to the homogenous system of difference equations (5) or (17). You can also easily see that multiplying any one of these solutions by a constant gives a solution and that the sum of several solutions is also a solution. Thus

$$(24) \quad y_t = \sum_{j=1}^n b_j V_j \lambda_j^t$$

is also a solution. We may, in fact, call it the *general* solution of the homogeneous equation, for the b_j can be determined to satisfy any *initial condition*, that is, to give any required value of y_0 , since for $t = 0$, (24) becomes

$$(25) \quad y_0 = \sum_{j=1}^n b_j V_j$$

which is just a non-singular system of linear equations to determine the b 's.

Finally, let x_t be *any* solution of the non-homogeneous equation (1), which we repeat here with x as the argument:

$$(26) \quad x_{t+1} = Ax_t + f_t$$

Note that this x_t may very well not satisfy the initial conditions we will need. The general solution of the system (1) is then

$$(27) \quad y_t = \sum_{j=1}^n b_j V_j \lambda_j^t + x_t$$

That is to say, the values of the b 's can be determined to match any initial condition. In the case of complex roots, the roots occur in conjugate pairs and the corresponding characteristic vectors are also complex conjugates. The b 's corresponding to conjugate vectors will also be conjugate. Thus, all complex numbers occurring in the summation on the right of (27) occur in conjugate pairs, so that the complex parts cancel out. Thus, though the right side of (27) may be strewn with complex numbers, the left side, the part we actually see in the solution, has only real numbers.

2. Frequency Response

We have given much attention in the preceding section to the general solution of the homogeneous equations but said almost nothing about particular solutions to the full, non-homogeneous system

$$(1) \quad y_{t+1} = Ay_t + f_t$$

or, in terms of the single, n^{th} order equation for one of the variables,

$$(2) \quad y_{1,t+n} + c_{n-1}y_{1,t+n-1} + c_{n-2}y_{1,t+n-2} + \dots + c_0y_{1,t} = f(t)$$

We will now employ mainly the single equation form (2) and drop the 1 subscript, since the same

equation right-hand side applies to any of the variables.

In the study of dynamic systems, the function $f(t)$ is often called the *input* function, while y is the *output*, terms which correspond closely to our exogenous and endogenous variables.

There is not a lot that can be said in general about the particular solutions, but for specific $f(t)$ functions, there are easily found particular solutions. For example, if $f(t)$ is a constant, then there is a particular solution which is constant and its value is easily calculated from (2). If $f(t)$ is a polynomial in t , then there is a particular solution having the form of a polynomial of the same degree in t .

If

$$(3) \quad f(t) = \gamma^t$$

where γ is some possibly complex constant, then, as you can see by substitution, there is a particular solution $y_t = b\gamma^t$ where b is a constant with value

$$(4) \quad b = \frac{1}{\gamma^n + c_{n-1}\gamma^{n-1} + c_{n-2}\gamma^{n-2} + \dots + c_0}.$$

This function is called the *frequency response* function for reasons we now need to explain.

As already noted, some of the λ may be complex; if so, they occur in conjugate pairs. In this context, a complex root is most revealingly thought of in its polar representation,

$$(5) \quad \lambda = r(\cos \alpha + i \sin \alpha)$$

as illustrated in Figure 7.1 to the right. The particularly attractive aspect of the polar representation is that multiplication of two complex numbers amounts to the multiplication of their absolute values, r , and *addition* of their angles, α . Thus, if

$$\lambda_1 = r(\cos \alpha + i \sin \alpha)$$

$$\lambda_2 = r(\cos \beta + i \sin \beta)$$

then

$$\begin{aligned} \lambda_1 \lambda_2 &= rs(\cos \alpha \cos \beta - \sin \alpha \sin \beta + i(\sin \alpha \cos \beta + \cos \alpha \sin \beta)) \\ &= rs(\cos(\alpha + \beta) + i \sin(\alpha + \beta)) \end{aligned}$$

by the formulas for the sine and cosine of the sum of two angles. With λ given by (4),

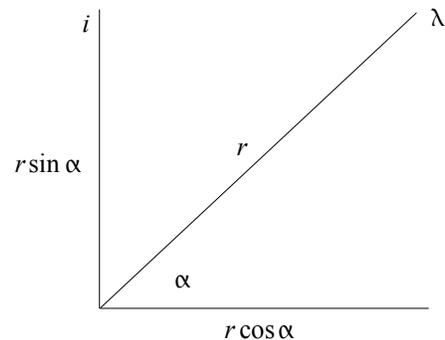


Figure 7.1

$$(6) \quad \lambda^t = r^t(\cos t\alpha + i \sin t\alpha) .$$

Thus, terms of the general solution coming from real roots either grow or decline exponentially, while terms stemming from complex roots oscillate, with the amplitude of the oscillations either growing or declining exponentially depending on the absolute value of the root. The frequency of the oscillation is determined by the angle α of the root; it does not change as the amplitude swells or dampens. If all of the roots of the homogeneous part are less than 1.0 in absolute value, the system is said to be *stable*.

In a stable system, a major question is how it responds to a sinusoidal input represented by (3). According to the absolute value of b in equation (4), the frequency response curve, the system may magnify the oscillations at some frequencies while dampening them at others.

To illustrate these ideas, we may consider a second order system with two complex conjugate roots given by $\lambda_1 = r(\cos \alpha + i \sin \alpha)$ and $\lambda_2 = r(\cos \alpha - i \sin \alpha)$. For this system, the polynomial (7.1.6) is given by

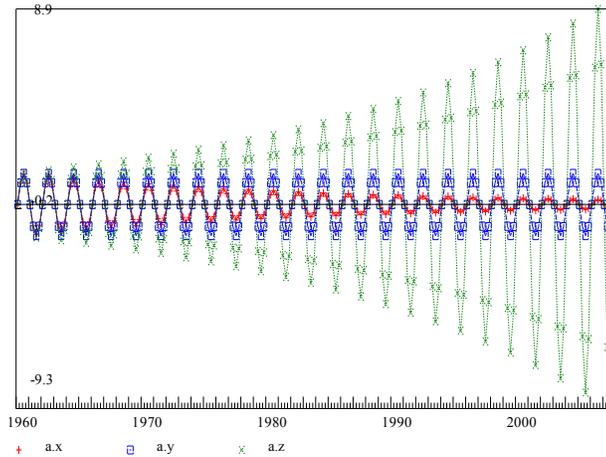
$$(7) \quad (\lambda - r(\cos \alpha + i \sin \alpha))(\lambda - r(\cos \alpha - i \sin \alpha)) = \lambda^2 - \lambda 2r \cos \alpha + r^2 = 0 .$$

Let us take three cases, all with $\alpha = \pi/4$ so that $\cos \alpha = .70711$, and with $r = .99, 1.0$, and 1.01 , respectively. In the first case, the system is just inside the stable zone; in the third, it is mildly explosive; and in the second, it is exactly on the boundary. The three equations, in the form of equation (4), are

$$\begin{aligned} y_t - 1.40008 y_{t-1} + .9801 y_{t-2} &= 0 \\ y_t - 1.41422 y_{t-1} + 1.0000 y_{t-2} &= 0 \\ y_t - 1.42836 y_{t-1} + 1.0201 y_{t-2} &= 0 \end{aligned}$$

These three equations were simulated from 1960.1 to 2007.4 with the initial condition that $y_{1960.1} = 0, y_{1960.2} = 0$, and $y_{1960.3} = 1.0$. The result is shown in the graph labeled “Deterministic Solutions” below. All three equations show a strict periodicity of 8 periods per cycle, as could be predicted from the fact that $\alpha = \pi/4$ radians, so that it requires 8 periods to move completely around the 2π radians of the full circle. The top equation, with $r = .99$ diminishes steadily in amplitude; the middle one, with $r = 1.00$ keeps a constant amplitude; the third, with $r = 1.01$ gradually increases in amplitude.

Deterministic Solutions



(The simulations were made by running a model made from the following Master file:

```

fex y = 0
fex z = 0
fex x = 0
r x = 1.400078*x[1] - .9801*x[2] + e1
d
r y = 1.41422*y[1] - 1.00*y[2] + e1
d
r z = 1.428362*z[1] -1.0201*z[2] + e1
d
end

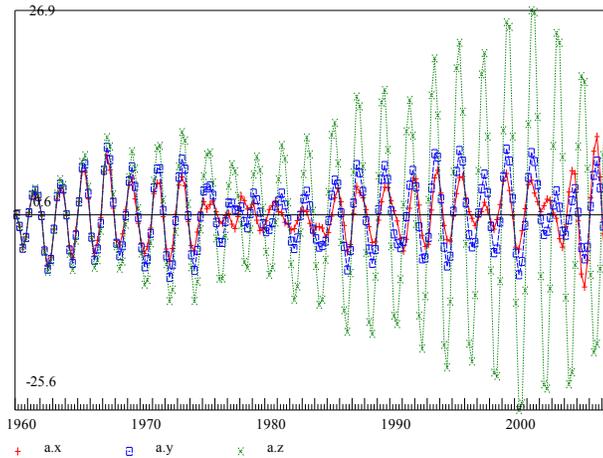
```

and $e1$ was given the value of 1 in 1960.3 and otherwise 0.)

These results are, of course, exactly in line with expectations from the theory. But now let us try something different.

Let us simulate the equations with random error terms, ε_t , generated by the `@normal()` function of *G7* to be a series of independent random variables each normal (0,1). One might imagine that these shocks would cancel out or that the result would be utterly chaotic. In fact, the result is quite different, as shown by the graph below labeled “Stochastic Simulation”. At the beginning, the result is quite irregular and quite similar for all three equations. Then gradually a pattern emerges, namely exactly the 8-period cycle of the deterministic simulations. The equation with the root just outside the unit circle is clearly exploding, but the equation with the root just inside the unit circle is *not* converging to zero but to a sort of limit cycle. The equation with the root on the unit circle seems *labile*, equally prone to expanding or contracting.

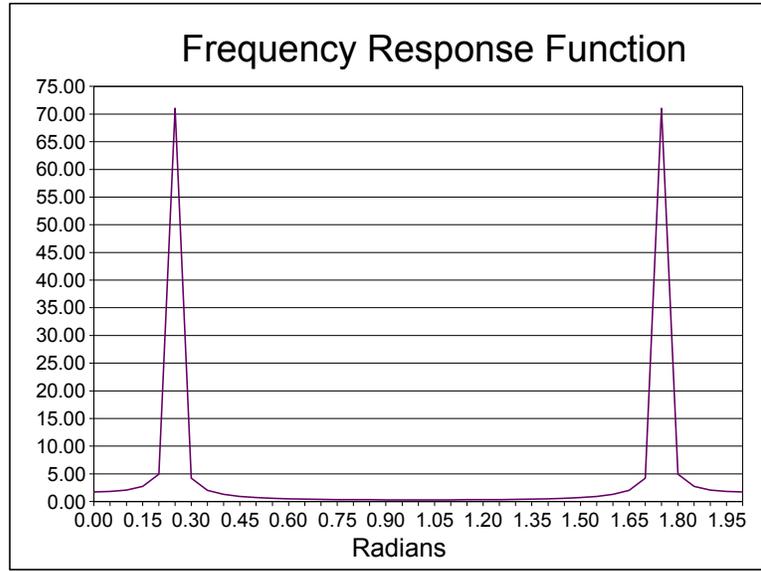
Stochastic Simulation



This tendency of the system to oscillate at a specific frequency when subject to utterly random shocks is surely both beautiful and somewhat mysterious. I will refer to it as the *emergent cycle* phenomenon. You will recognize it as reminiscent of the cycles which appeared in running the *AMI* model of Chapter 1. Our analysis, however, can now throw some light on why it happens.

Think about what equation (3) would mean if γ were a complex number lying on the unit circle in the complex plane. If we think of time as continuous, its real part would follow a cosine function while its imaginary part followed a sine curve. If time is discrete, then we observe only equally spaced points on those curves. The particular solution, $y_t = b\gamma^t$, looks much the same – also sine and cosine functions with the same frequency as the input function – but with a different amplitude given by the absolute value of b . The box to the right shows the absolute value of b calculated by equation (4) for various values of γ lying on the unit circle at values of α/π shown in the left column of the box. At a frequency of $.25\pi$, there is an enormous, 71-fold amplification of the input signal. This frequency corresponds, of course, to the root which lies just inside the unit circle at that frequency. Other frequencies are much less magnified, and the high frequencies are damped. This function is called the *frequency response function* of the system. It is closely related to the *transfer function* commonly used in engineering analysis of systems. A graph of the function for our equation is shown below.

α/π	$ b $
0.00	1.72
0.05	1.80
0.10	2.07
0.15	2.74
0.20	4.95
0.25	71.07
0.30	4.22
0.35	1.99
0.40	1.27
0.45	0.92
0.50	0.71
0.55	0.58
0.60	0.50
0.65	0.43
0.70	0.39
0.75	0.36
0.80	0.33
0.85	0.32
0.90	0.30
0.95	0.30
1.00	0.30



3. Explaining the Emergent Cycle Phenomenon

We can think of the input to the system, the series of random shocks, as the sum of T different functions, where T is the number of periods over which the system is run:

$$(8) \quad f(t) = f_1(t) + f_2(t) + \dots + f_T(t)$$

where $f_j(t) = 0$ for $j \neq t$, and is equal to the value of the shock in period t for $j = t$. Because of the linearity of the system, the solution with the input $f(t)$ is the sum of the solutions for each of the separately. The solution for $f_j(t)$ is zero for $t < j$; thereafter, it follows a damped sinusoidal curve that can be thought of as $y_{j,t} = b_j r^t \gamma^t$, where $r\gamma$ is the root, with γ lying on the unit circle. (In systems of order higher than 2, there may be other terms in the solution for $y_{j,t}$; we assume that all but the largest complex root damp away quick and leave us concerned only with its contribution to the solution.) The complex number b_j is determined by the initial condition. By (8), therefore, the output, $y(t)$, is

$$(9) \quad y_{j,t} = \left(\sum_{j=1}^t b_j r^{t-j} \right) \gamma^t = A_t \gamma^t,$$

where the second equation defines the complex number A_t . The γ^t on the right represents a steady sinusoidal function with a fixed frequency. Multiplication by the complex number A_t in front of it changes its amplitude and shifts its phase – that is, it adds the angle of A_t to that of γ^t – but it does not change its frequency. At first, as the system starts up, each period brings significant changes to A_t , and the output does not look much like a sine curve. As A_t builds up, however, as it will do if r is close to 1, the new addition to A_t , which is probably typical of the terms already in it, will bring little change to the number, so that the change in $f(t)$ over a few periods will be determined chiefly by γ^t

and will look much like a sine curve.

An alternative way to explain the emergent cycle phenomenon is to use a different decomposition of the input function, this time as a sum of sine and cosine functions as follows. Let $T+1$ be the total length of the simulation with observations numbered $t = 0, 1, 2, \dots, T$, and, to keep matters simple, let T be an even number and define $M = T/2$. Further, define the angles

$$(10) \quad \omega_j = \frac{2\pi j}{T+1} \quad \text{for } j = 1, \dots, M$$

and the variables

$$(11) \quad x_j(t) = \cos \omega_j t \quad \text{and} \quad y_j(t) = \sin \omega_j t \quad \text{for } j = 1, 2, \dots, M$$

Notice that as t goes from 0 to T , the cosine function $x_1(t)$ makes one complete cycle (except for the last point), $x_2(t)$ makes two complete cycles (except for the last point), and so on. The same is true for the sine functions. Now let us imagine regressing the input function, $f(t)$ on these $2M$ variables plus an intercept. Since we have T variables and T points, we will get a perfect fit and can write

$$(12) \quad f(t) = a_0 + \sum_{j=1}^M a_j x_j(t) + b_j y_j(t)$$

(There is no problem about the linear independence of the various x_j and y_j variables; in fact, they are orthogonal, so the a 's and b 's can be computed one-by-one, but these facts need not detain us here.)

To use what we know about the frequency response function, equation (4), we just need to convert (16) into a weighted sum of powers of a complex number on the unit circle. To do so, just define $\gamma_j = \cos \omega_j + i \sin \omega_j$ and denote its complex conjugate by $\bar{\gamma}_j = \cos \omega_j - i \sin \omega_j$. Then

$$(13) \quad \gamma_j^t = \cos \omega t + i \sin \omega t \quad \text{and} \quad \bar{\gamma}_j^t = \cos \omega t - i \sin \omega t$$

and

$$(14) \quad \frac{a_j}{2}(\gamma_j^t + \bar{\gamma}_j^t) - \frac{i b_j}{2}(\gamma_j^t - \bar{\gamma}_j^t) = a_j \cos \omega_j t + b_j \sin \omega_j t$$

or, by defining $d_j = (a_j - i b_j)/2$,

$$(15) \quad d_j \gamma_j^t + \bar{d}_j \bar{\gamma}_j^t = a_j \cos \omega_j t + b_j \sin \omega_j t .$$

Equation (12) can now be written as

$$(16) \quad f(t) = a_0 + \sum_{j=1}^M d_j \gamma_j^t + \bar{d}_j \bar{\gamma}_j^t$$

as was desired.

The frequency response function of the system, the absolute value of equation (4), evaluated at each γ_j will thus show how much the system amplifies or dampens the various frequencies in the input function. Naturally, it may take a while for the strongly amplified frequency to emerge, since the initial conditions and the terms from the homogeneous part may play a prominent role at first. But as they fade out, we will observe the emergent cycle phenomenon.

This second way of trying to understand the emergent cycle phenomenon is connected with the subjects of *Fourier* and *spectral* analysis of time series. The word *spectrum* indicates the origins of the subject in the physics of light, while the expression *white noise* for a series of independent, identically distributed random variables is a hybrid between the physics of light and the physics of sound. In many technical studies, ranging from vibration in airplanes to patterns in the human heart beat, it is often just as natural to think in terms of the fraction of the total variance of the series accounted for by sinusoidal curves of different frequencies as to think in terms of the original time series. For a given T, the share of the sinusoidal curves with angular frequency ω_j in the total variance is $(a_j^2 + b_j^2)/2$. If, however, we double T, there will be twice as many frequencies, and the fraction of the variance explained by any one will go down. To get a measure associated with ω_j which is potentially independent of the length of period over which it was estimated, one conventionally concentrates on the function

$$(17) \quad s(\omega_j) = \frac{T}{8\pi} (a_j^2 + b_j^2)$$

which is called the *spectrum* of the series. The real usefulness of this way of looking at a time series is that the spectrum of the output of a linear system is the spectrum of the input multiplied by the frequency response function of the system, as we have essentially already seen.

We have seen, I believe, that the *concepts* of spectral analysis can be useful in *understanding* why models of the economy -- and even the economy itself -- can have a tendency to cycle. Is it useful to look at the spectra of economic time series? I was once hopeful that it might be. And for analysis of seasonal patterns, perhaps it is. But for the study of business fluctuations in the economy, I have been unimpressed. All the interesting historical particularities of each cycle get mushed together. For heart beats or airplane vibrations, the only thing that matters is the repetitious pattern. But we live inside business cycles; we watch them develop, we follow each with its own fascinating quirks. We have stories to tell about what caused this or that turn of a series. All that gets lost in the spectra which, to me at least, look pretty uninteresting.

Looking at the response of a model to a cycle in an exogenous variable, however, could be much more interesting. Most models, of course, are nonlinear, even if only in the value = price*quantity equations. The strict linearity assumptions about the model made in this chapter will not hold in

practice. Nonetheless, you may find here some interesting possibilities for experimentation with your model, as suggested in the following exercise.

Exercise

7.1 Explore the sensitivity of your model to cycles in the exogenous variables. For example, if gR is an exogenous variable, make up this substitute for it:

```
f gRalt = gR + 100.*@sin(2*3.14159*.2*time)
```

This $gRalt$ will have a sine curve of amplitude 100 with a five year period added to the historical values of gR . Save it into a file, edit the file to introduce the variable with “update gR ”, and *add* this file at the] prompt before running the model. Compare the results of the model run with this value of government purchases with the historical simulation. Repeat the experiment with different values of the fraction before the *time* variable. Does your model, or more specifically, $gdpR$ in your model, respond differently to different frequencies of variation in gR ?

Index

accelerator theory of investment.....	3, 35	Fourier analysis.....	133
Accounting Model.....	7, 9	frequency response.....	119, 125, 126, 127, 132
AMI model.....	36, 73	Frobenius matrix.....	122
assigned bank.....	15	g.cfg file.....	59
asymptotic idiocy.....	114	G7 commands.....	
BasePred.....	115	add.....	61
BEA.....	11	bank.....	19
behavioral ratio.....	8, 16, 100	catch.....	29, 30
Beta.....	32	clear.....	63
Build commands.....		con.....	91
add.....	35	do.....	112
addprint.....	82	f.....	15
check.....	16, 17, 66	fdates.....	63
checkdup.....	82	fex.....	16, 88
maxit.....	82	gdates.....	20
bws.....	67	gname.....	20
chain weighting.....	25	gr.....	15
character set.....	60	gr *.....	60
characteristic polynomial.....	123	Graph settings.....	64
characteristic values.....	123	lgr.....	66
characteristic vectors.....	123	lim.....	29
closeness of fit.....	43	mode.....	78
coefficient of multiple determination.....	31, 43	r.....	29, 33
companion matrix.....	122	recur.....	105, 112
Compare.....	68	rhosave.....	76
Compare commands.....	69	save.....	29, 32
*.....	70	sgr.....	66
&.....	70	showt.....	47
computer prices.....	24	sma.....	96
Consumption equation.....	2, 105	sty.....	79
counter-historical simulation.....	1, 16, 17, 73, 74, 109	tdates.....	79
Danilevsky process.....	122	ti.....	59
degrees of freedom.....	31, 44	vaxl.....	65
depreciation.....	11	vrange.....	64
depar.....	78	!.....	33
difference equations.....	119	[] notation.....	29
Distributed lags.....	94	@cum function.....	29, 85
DoFree.....	31	#.....	15
Durbin-Watson.....	31, 44, 45	Gauss-Jordan reduction.....	42
DW.....	31	Hedonic deflator.....	24
Elas.....	31	historical simulation.....	1
elasticity.....	31, 46	identity-centered modeling.....	7, 8
endogenous.....	2	import equation.....	33
exogall.reg.....	67	inflation.....	8, 86
exogenous.....	2	INFORUM.....	iii
exogenous variable.....	20	inventory investment.....	32
expected inflation.....	87	Investment Equation.....	3, 100
false exogeneity.....	74, 103	lagged dependent variable.....	109, 115

least squares.....	28, 37, 56	Run commands.....	
Leverage.....	57	cta.....	75
MAPE.....	31	mul.....	75
marginal explanatory value.....	31, 46, 55	ovr.....	75
master file.....	14, 18	rho.....	75
Matrix.....	48	skip.....	75
Mean.....	32	ti.....	74
mean absolute percentage error.....	31, 44	type.....	74
Mexval.....	31, 46, 55	update.....	74
NAN.....	81	run_gr.....	67
National Income and Product Accounts.....	9	run_nam.....	67
NIPA.....	viii, 9, 23	run_xog.....	67
Non-residential structures.....	108	Savings Function.....	100
NorRes.....	31	SEE.....	31, 43
predicted value.....	42	SEE+1.....	31, 46
QUEST model.....	ix	similar matrix.....	123
Quip databank.....	15, 25	simultaneous equation bias.....	ix, 107
r-bar-square.....	31	skipall_fix.....	67
RBSQ.....	31	SNA.....	16
recursive regression.....	112	Soft constraints.....	90
Reg-coef.....	31	spectral analysis.....	133
regression coefficients.....	31, 32, 46, 50, 91	spectrum.....	133
Residential construction.....	116	Statistical discrepancy.....	13
residuals.....	42	stub file.....	68
RHO.....	31	tautological model.....	7
rho adjustment.....	31, 46, 67, 76, 78	transfer function.....	130
RSQ.....	31	workspace.....	15, 57, 59, 69