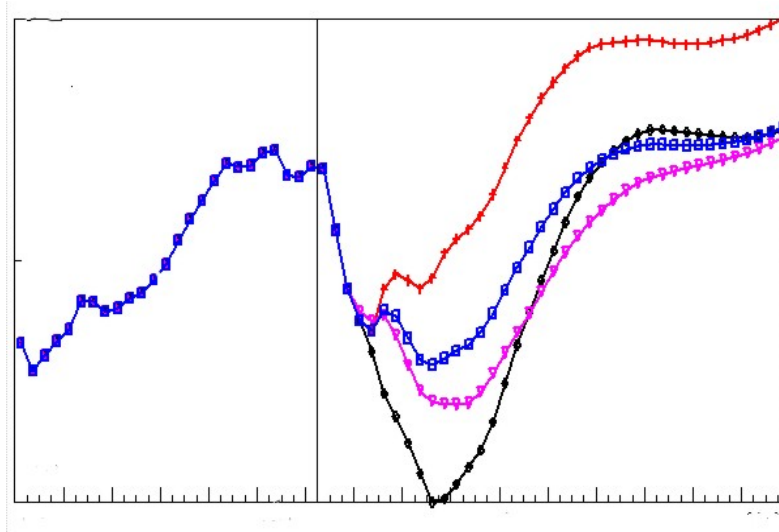# The Craft of Economic Modeling

Part 2

Clopper Almon

Draft of April 2004

The Craft of Economic Modeling

Part 2

Clopper Almon

Department of Economics
University of Maryland
College Park, MD 20742
almon@inforum.umd.edu

Draft of April 2004

CONTENTS

Chapter 8. QUEST - A Quarterly Econometric Structural Model

1. Overview

In Part 1, we developed a very simple model and suggested some directions in which it could be expanded. In the present chapter, we will carry out some of the suggestions while trying to follow the good advice of the last chapter of Part 1. In particular, our model will

refine the consumption and employment functions presented previously.

divide fixed investment into three major components, equipment, residences, and other structures, and develop appropriate equations for each.

develop equations for exports and imports.

complete the income side of the model with equations for capital consumption, profits, dividends, interest rates, interest payments and income, employee compensation and proprietor income.

calculate revenues from various taxes, government expenditures in current prices (from variables exogenous in constant prices), interest payments, and budgetary deficits or surpluses for the federal government and, separately, for the combination of state and local governments.

The word "structural" in the name of the Quest model is noteworthy. Quest is a model intended to embody and test an understanding of how the economy works. It is concerned with how aggregate demand affects employment, how employment affects unemployment, how unemployment affects prices, how prices and money supply affect interest rates and incomes, and how incomes, interest rates, and prices affect investment, consumption, imports, and exports, which make up aggregate. demand. The model embodies a view of how each link in this closed-loop chain works. Satisfactory performance is not to judged by how well it works forecasting a few quarters ahead, but by how well it holds up over a much longer period. Can it keep employment within a few percent of the labor force over decades? Can it keep inflation in line with the increase in money supply though it does not use money supply in the inflation equation? Can it right itself if thrown off course for a few quarters? We will test it in 21-year historical simulation, time enough for it to go seriously astray if it is inclined to do so.

In this respect, Quest is quite different from most quarterly models of my acquaintance. They are usually aimed at short-term forecasting, usually of not more than eight quarters. They can therefore make extensive use of lagged values of dependent variables in the regression equations. The use of these lagged dependent variables gives close fits but leaves little variability for identifying the parameters of the underlying structural equations, which are often rather weak in such models. Our interest centers in the structural equations. In estimating the equations of

Chapter 8. QUEST - A Quarterly Econometric Structural Model

1. Overview

In Part 1, we developed a very simple model and suggested some directions in which it could be expanded. In the present chapter, we will carry out some of the suggestions while trying to follow the good advice of the last chapter of Part 1. In particular, our model will

refine the consumption and employment functions presented previously.

divide fixed investment into three major components, equipment, residences, and other structures, and develop appropriate equations for each.

develop equations for exports and imports.

complete the income side of the model with equations for capital consumption, profits, dividends, interest rates, interest payments and income, employee compensation and proprietor income.

calculate revenues from various taxes, government expenditures in current prices (from variables exogenous in constant prices), interest payments, and budgetary deficits or surpluses for the federal government and, separately, for the combination of state and local governments.

The word "structural" in the name of the Quest model is noteworthy. Quest is a model intended to embody and test an understanding of how the economy works. It is concerned with how aggregate demand affects employment, how employment affects unemployment, how unemployment affects prices, how prices and money supply affect interest rates and incomes, and how incomes, interest rates, and prices affect investment, consumption, imports, and exports, which make up aggregate. demand. The model embodies a view of how each link in this closed-loop chain works. Satisfactory performance is not to judged by how well it works forecasting a few quarters ahead, but by how well it holds up over a much longer period. Can it keep employment within a few percent of the labor force over decades? Can it keep inflation in line with the increase in money supply though it does not use money supply in the inflation equation? Can it right itself if thrown off course for a few quarters? We will test it in 21-year historical simulation, time enough for it to go seriously astray if it is inclined to do so.

In this respect, Quest is quite different from most quarterly models of my acquaintance. They are usually aimed at short-term forecasting, usually of not more than eight quarters. They can therefore make extensive use of lagged values of dependent variables in the regression equations. The use of these lagged dependent variables gives close fits but leaves little variability for identifying the parameters of the underlying structural equations, which are often rather weak in such models. Our interest centers in the structural equations. In estimating the equations of

Chapter 8. QUEST - A Quarterly Econometric Structural Model

1. Overview

In Part 1, we developed a very simple model and suggested some directions in which it could be expanded. In the present chapter, we will carry out some of the suggestions while trying to follow the good advice of the last chapter of Part 1. In particular, our model will

refine the consumption and employment functions presented previously.

divide fixed investment into three major components, equipment, residences, and other structures, and develop appropriate equations for each.

develop equations for exports and imports.

complete the income side of the model with equations for capital consumption, profits, dividends, interest rates, interest payments and income, employee compensation and proprietor income.

calculate revenues from various taxes, government expenditures in current prices (from variables exogenous in constant prices), interest payments, and budgetary deficits or surpluses for the federal government and, separately, for the combination of state and local governments.

The word "structural" in the name of the Quest model is noteworthy. Quest is a model intended to embody and test an understanding of how the economy works. It is concerned with how aggregate demand affects employment, how employment affects unemployment, how unemployment affects prices, how prices and money supply affect interest rates and incomes, and how incomes, interest rates, and prices affect investment, consumption, imports, and exports, which make up aggregate. demand. The model embodies a view of how each link in this closed-loop chain works. Satisfactory performance is not to judged by how well it works forecasting a few quarters ahead, but by how well it holds up over a much longer period. Can it keep employment within a few percent of the labor force over decades? Can it keep inflation in line with the increase in money supply though it does not use money supply in the inflation equation? Can it right itself if thrown off course for a few quarters? We will test it in 21-year historical simulation, time enough for it to go seriously astray if it is inclined to do so.

In this respect, Quest is quite different from most quarterly models of my acquaintance. They are usually aimed at short-term forecasting, usually of not more than eight quarters. They can therefore make extensive use of lagged values of dependent variables in the regression equations. The use of these lagged dependent variables gives close fits but leaves little variability for identifying the parameters of the underlying structural equations, which are often rather weak in such models. Our interest centers in the structural equations. In estimating the equations of

Quest, therefore, we have avoided lagged values of dependent variables in the regression equations. When used for short-term forecasting, Quest uses the rho-adjustment method of error correction described in Chapter 2.

Models often have a special purpose, a special question they are designed to answer. Quest is basically a general-purpose marcoeconomic model, but it would be less than fair to the reader not to mention that there was a particular question on my mind as I worked on it in the summer of 1999. As in the summer of 1929, exactly seventy years earlier, the economy was growing strongly and the stock market was at unprecedented — and, quite possibly, unjustified — highs. The run-up in the stock market was generally attributed to the influx of footloose capital from Asian markets. At the first sign of a drop, this capital could leave as suddenly as it came. The stock market would then fall. But how would that fall affect employment and output in the real economy? As I revised the model in the summer of 2001, the stock market had declined significantly, and economic growth had slowed sharply. How far the fall would go and how sharp the recession would be was still unclear.

The stock market plays no role in the National Income and Product accounts, but its performance can make people feel wealthy or poor and thus influence how they spend or save. It determines how much equity in a firm must be diluted in order to raise a given amount of capital by issuing stock. In this way, it affects the cost of capital as perceived by the owners of companies, and thus may affect investment. We will enter the Standard & Poor index of the prices of 500 stocks as an explanatory variable in a number of behavioral equations, and finally we will try to explain this variable by corporate profits and interest rates. The variable proves very helpful in a number of the equations, but the attempt to explain it is only partly successful. In particular, the rise in 1997- 2000 is very incompletely explained. To test out the rest of the model, we run it with this equation turned off. To get an idea of where the economy would be after a "crash" back to levels explainable by profits and interest rates, we just run the model with it turned on. The results are, shall we say, "instructive." But first we must look at the equations.

2. The Behavioral Equations

Personal consumption expenditures

We work up to the main equation for personal consumption expenditures with two supporting equations, one for expenditures on motor vehicles and one for Interest paid by consumers to business. The interest paid variable is particularly relevant because consumers must pay it out of their disposable income but it is not part of personal consumption. Thus, if interest payments rise relative to disposable income, they must come out of either savings or consumption. We will find out which choice consumers make. The expenditures on motor vehicles is important for total expenditures for two reasons. First, interest payments on car loans is a major component of the Interest paid by consumers to business. (Interest on home mortgages is *not* part of Interest paid by consumers to business, because home ownership is considered a business in the NIPA.) Second, the NIPA consider that an automobile is consumed in the quarter in which it is purchased. Consumers, however, think of the car as being consumed over its lifetime. Thus, if

automobile purchases are particularly strong in a certain quarter, there is a sort of savings in the form of automobiles. It would not be surprising to see all or most of that saving appear as consumption in the NIPA series. Though the same reasoning applies to other durables, their purchases are much less volatile than those of automobiles, so there is not much to be gained by such treatment.

We start with **personal consumption expenditures on motor vehicles.** It uses real disposable income accrued per capita, yRpc, lagged values of its first difference, dyRpc, the Treasury bill rate, rtb, multiplied by yRpc as an indicator of credit conditions, and an estimate of the wear-out of motor vehicles, mvWear.

Disposable income accrued is in most quarters exactly the same as disposable income. In a few quarters, however, billions of dollars of bonuses that should normally have been paid in the fourth quarter of one year were, for tax reasons, paid in the first quarter of the next. Consumers definitely based their consumption on the accrued rather than the disbursed income. We will therefore almost always use Personal disposable income accrued, pidisa, not Personal disposable income, but we will call it simply "disposable income."

The increments in this real disposable income per capita are crucial variables in this equation. Their total is 1.24. Since we are dealing with quarterly flows at annual rates, this 1.24 implies that a rise in annual income of $1 leads to an increase in the stock of motor vehicles of $.31 (= 1.28×.25). We shall return below to look at the pattern of the coefficients.

The deviation of the interest rate, rtb, from a typical value, here taken as 5 percent, is multiplied by yRpc so that the amplitude of its swings will grow at approximately the same rate as the growth in the dependent variable.

The wear-out variable required more than the usual constant wear rate. When a constant rate was used, the equation under-predicted at the beginning of the period and over-predicted at the end. It is common experience that automobiles last longer now than they did thirty years ago, so a declining wear-out or "spill" rate, *spilla*, was introduced. It is 10 percent per quarter in 1974.4, just before the beginning of the fit period, and declines at 2 percent per year. The usual "unit bucket" way of correcting for initial filling of a bucket is not valid with a variable spill rate, but calculations showed that, at these spill rates, filling was not a problem after 15 years, the time between the beginning of the Quip bank and the beginning of this regression. Without any constraint, the coefficient on this variable came out at .99058, thus indicating almost exact dollar-for-dollar replacement of the cars wearing out.

On the other hand, the income variable, yRpc, was not used because, when included, it had a small negative coefficient. That does not mean that motor vehicle expenditures do not depend on income, but rather that the dependence comes about entirely by expansion of the stock in response to an increase in income and then replacement of that stock.

3

In this and subsequent presentations, we do not show the full "catch" file but only the part not obvious from the display of the regression results. Thus, the *catch, save, limits, r, gr, gname,* and *spr* commands have been deleted to save space. In the graphs, the heavy line with no mark is the predicted value.

```
ti Motor Vehicles
subti Personal Consumption Expenditure Per Capita in 1996 $

# cdmvRpc is per capita consumption of motor vehicles in constant dollars
fex cdmvRpc  = cdmvR/pop

#Disposable Income per Capita
fex pidisaR = pidisa/gdpD
f yRpc = pidisaR/pop
f dyRpc = yRpc - yRpc[1]

# Interest rate X ypcR to represent credit conditions
f rtbXypc = .01*(rtb -5.)*yRpc
# (Real rate was tried, but was much less effective.)

# Create wearout of automobiles assuming 8% per quarter wearout rate
f spilla = .10*@exp(-.02*(time -15.))
f mvWearpc = spilla*@cum(mvSt,cdmvR[1],spilla)/pop
sma 50000 a3 a11 1
```


Motor Vehicles

Personal Consumption Expenditure Per Capita in 2000 $

- Predicted   □ Actual

Distributed Lag on Changes in Income
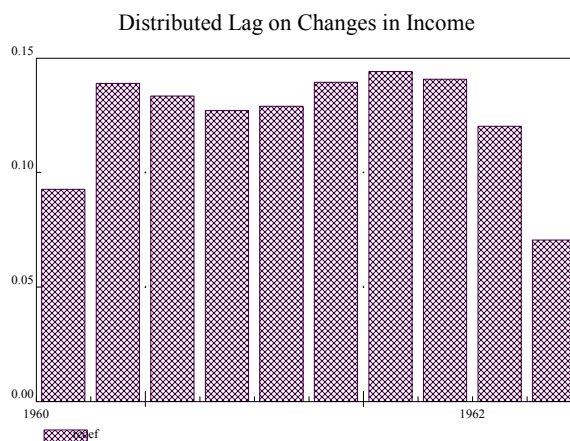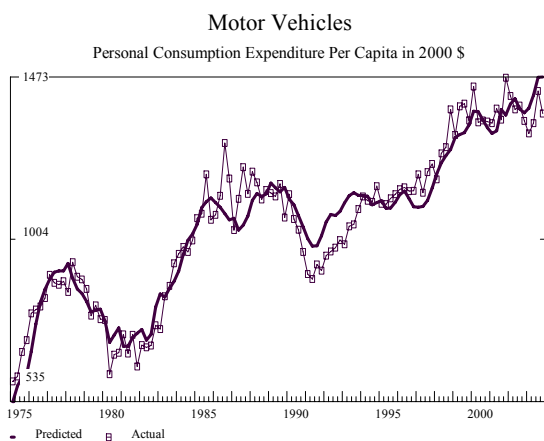
```
:                        Motor Vehicles
  SEE   =      62.26 RSQ  = 0.9237 RHO =   0.56 Obser  =  116 from 1975.100
  SEE+1 =      51.96 RBSQ = 0.9148 DW  =   0.88 DoFree =  103 to   2003.400
  MAPE  =       5.01
    Variable name          Reg-Coef  Mexval   Elas   NorRes    Mean    Beta
 0 cdmvRpc              - - - - - - - - - - - - - - - - -  1044.74 - - -
 1 intercept             -52.73023     0.5   -0.05    12.80     1.00
 2 dyRpc                   0.09268     3.5    0.01    12.79   109.98   0.076
 3 dyRpc[1]                0.13883    10.2    0.01    12.77   109.88   0.113
 4 dyRpc[2]                0.13347    17.8    0.01    12.73   107.49   0.109
 5 dyRpc[3]                0.12704    17.6    0.01    12.69   105.50   0.104
 6 dyRpc[4]                0.12896    19.1    0.01    12.59   100.66   0.109
 7 dyRpc[5]                0.13939    23.4    0.01    12.38   100.41   0.118
 8 dyRpc[6]                0.14416    26.0    0.01    11.99    98.68   0.123
 9 dyRpc[7]                0.14068    24.5    0.01    11.52    94.27   0.120
10 dyRpc[8]                0.12025    17.7    0.01    11.03    88.58   0.102
11 dyRpc[9]                0.07049     8.7    0.01    10.45    97.07   0.058
12 rtbXypc[1]             -0.08131    13.3   -0.02     4.88   201.52  -0.196
13 mvWearpc                1.10607   120.8    0.95     1.00   893.89   0.740
```

4

```
id cdmvR = cdmvRpc*pop
id cdmv = cdmvR*gdpD
```

The fit is shown above in the graph on the left. The graph on the right shows the values of the regression coefficients on dyRpc and its lagged values. It is intended to help interpret these coefficients. It shows how expenditures would respond if, after a long period of being constant, income were to rise by $1.00 and then remain constant at that new value. During the period of constant income, expenditures on motor vehicles would have reached a constant, equilibrium level. In the first quarter of the income rise, motor vehicle expenditures would rise by $.09268. In the second quarter they would be $.13883 (the coefficient on dyRpc[1], which would be 1.00 in that quarter) above the pre-rise equilibrium, and so on, as shown in the graph.

This sort of response will characterize many of our equations. We won't graph the others, but it is important for the reader to visualize these responses. This tendency of consumers to "go on a spree" of automobile buying after an increase in income is both very understandable — the increase in income allows them to borrow the money to buy the cars — and very much a generator of cycles in the economy.   Actually, in this particular case, we have somewhat oversimplified the response, because, four quarters after the response of expenditures begins, the replacement response through the mvWear term begins, faintly at first, then producing a damped wave of expenditures as the initial purchases are replaced.

The fit of the automobile equation is surprisingly good, given the volatile nature of the series. Besides the strong and long transient response to increases in income and the replacement wave, the equation is noteworthy for its negative (theoretically correct) response to interest rates. Just how large a response is this? Perhaps the best answer here is given by the beta coefficient of −.196. That is to say, as the interest rate variable moves by 1.0 standard deviations, the dependent variable moves by .196 of its standard deviations. Another way to look at this question is to ask how much would a one point drop in the interest rate, say from 6 percent to 5 percent, increase expenditures on motor vehicles. At a value of yRpc of 20000, the answer about $16 (= .01*20000*.08131) per person per year. The swing in the dependent variable in from its low point to its high point was over $900, so the sensitivity to interest rates, while not negligible, is not very important.

For **Interest paid by consumers to business**, the dependent variable is expressed as a percent of disposable income. The most important explanatory variable tries to capture the interest payments on past automobile purchases. It is assumed that the loans are paid off at the rate of about 9 percent per quarter, so that about 35 percent is paid off in the first year. The outstanding amount, if all automobiles are bought with loans, is called *autfi* (automotive financing.) The interest on this amount at the Treasury bill rate (*rtb*) is called *autfir*. If the interest rate charged is *rtb+a,* then the payments should be *a*autfi + autfir*. If all automobiles and nothing else were financed, the coefficient on *autfir* should be 1.0.   In the equation as estimated, both these variables are expressed as percent of disposable income, *autfin* and *autfis*, respectively. The coefficient on *autfis* comes out close to the expected 1.0, while the value of *a,* the coefficient of *autfin*, emerges as .01478, so the financing rate appears to be less about 1.5 above the Treasury

bill rate, less than I would have expected. Notice the large values of Beta for *autfis*; the dependent variable is quite sensitive to it.

The other important variable is the exponentially-weighted average — created with the @cum function — of recent values of the savings rate. Its justification is that one way that people can save is by paying off debt on which they are paying interest. It should also be pointed out that interest payments on debt other than automotive, in so far as they are a constant fraction of disposable income, are absorbed into the intercept of the equation.   The last variable, the rate of change of the money supply, was intended to indicate the ease of getting loans. It did not prove successful.

piipcb - Interest Paid by Consumers to Business

Predicted   Actual
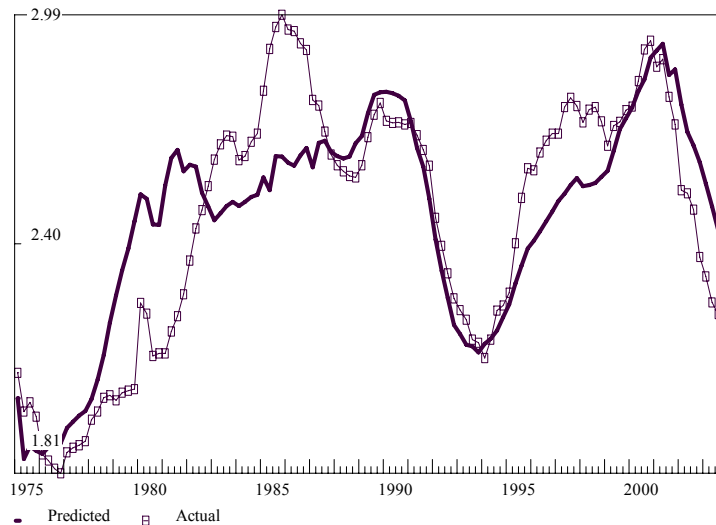
```
title piipcb - Interest Paid by Consumers to Business

#  shipcb is share of interest in disposable income less savings and transfers
fex shipcb =  100.*piipcb/pidisa

# autfi is a consumption of motor vehicles bucket with a spill of 0.09
f autfi  =  @cum(autfi ,.25*cdmv,.09)
f autfin = 100.*autfi/pidisa

f autfir = @cum(autfir,.0025*rtb*cdmv,.09)
f autfis = 100.*autfir/pidisa

#f odurfir = @cum(odurfir,.0025*rtb*(cd -cdmv),.09)
#f odurfis = 100.*odurfir/pidisa

#  savrat is the savings rate
f savrat = 100.*(pisav/pidisa)
#  b1sr is a savings rate bucket with a spill rate of 0.12
f b1sr   = @cum(b1sr,savrat,.12)
f dm1    = (m1 - m1[1])/m1[1]

:               piipcb - Interest Paid by Consumers to Business
```

6

```
SEE    =         0.18 RSQ  = 0.6869 RHO =    0.96 Obser  =  116 from 1975.100
SEE+1 =          0.05 RBSQ = 0.6757 DW  =    0.07 DoFree =  111 to   2003.400
MAPE  =          5.85
   Variable name           Reg-Coef  Mexval  Elas   NorRes    Mean    Beta
0 shipcb                   - - - - - - - - - - - - - - - -    2.46  - - -
1 intercept                1.39906     6.0   0.57    3.19     1.00
2 autfin                   0.05320     1.7   0.26    2.87    12.05   0.138
3 autfis                   1.79729    66.8   0.57    2.21     0.78   1.129
4 b1sr                    -0.01602    45.4  -0.40    1.00    60.67  -1.089
5 dm1                     -0.15048     0.0  -0.00    1.00     0.01  -0.006

id piipcb = 0.01*shipcb*pidisa
```

At last we are ready for the equation with the largest dependent variable in the model, **Personal consumption expenditures**. It is estimated in per capita terms, and the most important explanatory variable is certainly disposable income per capita, *yRpc,* and its first differences. Notice that the signs on the first difference terms are all negative. Instead of the splurge effect which we saw in the case of automobiles, there is a very gradual increase in spending to the level justified by an increase in income. Unconstrained, the coefficient came out implausibly high at 1.20 and led to wrong signs on some of the other variables. A soft constraint has been used to pull the coefficient back just under 1.0. We will return to this coefficient below.

Textbooks of macroeconomics usually make the savings rate — and, therefore, implicitly the consumption rate — depend on the interest rate. Our equation uses the Treasury bill rate less the expected rate of inflation, which I have called the perceived real interest rate. (The actual rate of inflation is not known until after the end of a quarter, so the expected rate may be more relevant for behavior.) To make the amplitude of its fluctuations grow with the growth of the dependent variable, it has been multiplied by real disposable income per capita to make the variable *rtbexXdi.* It has the expected negative sign, but not much importance — as indicated by its mexval — relative to the other variables which never seem to get mentioned in the textbooks.

Savings in the form of automobiles, *sautos*, is the excess of spending on motor vehicles over an estimate of their wearout. Theoretically, its coefficient should be 1.0. It came out at .55, a satisfactory value for such a theoretically constructed variable.

Interest paid by persons to business, it should be recalled, is the principal use of disposable income other than consumption. After converting it to constant price, per capita terms, it is called *piipcbRpc* and also came out with the expected negative sign. Its value indicates that more than all of an increase in these payments will be taken from consumption so that an increase in savings results. This finding is new with the 2003 revision of the NIPA. Previously, only about 40 percent of an increase in these interest payments were found to come out of consumption while 60 percent was be paid by reducing savings. Partly the rise in the coefficient is due to data revisions and partly to the addition of two years of recent data.

Inflation, as we know, influences interest rates and, therefore, interest income of persons. But a savvy investor will recognize that if he spends all his interest in times of rapid inflation, the real value of his interest-yielding assets will shrink. To keep up the value of his investment, he must save the fraction of his interest receipts due to inflation. The variable *intsavpcR* is an attempt to

7

measure this amount in real terms per capita. Theoretically, its coefficient should be -1; it comes out at about -.42, a satisfactory value for a variable whose relevance depends on very conscious consumers. This variable has a profound influence on the macroeconomic properties of the model. For example, if money supply is increased and interest rates lowered, investment is stimulated, unemployment is reduced, and inflation picks up. But as soon as it does, this variable causes an increase in savings and a reduction in consumer spending, which offsets the rise in investment. Thus, monetary policy in a model with this effect is apt to prove a weak instrument. Since the effect is both intuitively evident and quantitatively important, it is surprising that it seems to have gone unnoticed in macroeconomic textbooks.

Contributions for social insurance, even the employee's half of social security, is deducted before reaching Personal income in the NIPA. It would not be irrational, however, for consumers to consider that these contributions are, in fact, a form of saving which substitutes for their private saving. We have included the *consipcR* variable to allow for this possibility. It appears that consumers consider that these contributions are a substitute for saving.

Unemployment may have an influence on consumption. The unemployed are likely to spend a very large fraction of their income, so, given income, we would expect spending to be high when unemployment is high. In the reverse direction, when unemployment is exceptionally low, people may recognize that times are exceptionally good and take the opportunity to increase their savings. This effect could be represented either by *u*, the unemployment rate, or by its reciprocal, *ur = 1/u*. The simple *u* gives the better mexval, 4.1 compared to 1.3, but the historical simulation is strikingly better with *ur*. Without either, unemployment goes decidely negative in the mid 1980's; with *u,* it still goes slightly negative; with *ur* it stays above 2 percent. Thus, this variable – of little importance in the fit of the equation – is essential to the performance of the overall model.

Last but certainly not least, we come to the real stock market value per capita, *sp500Rpc.* It is the Standard and Poor's index of 500 stocks, *sp500*, deflated by the GDP deflator and divided by population. The graph on the left below shows that this variable. Between 1975 and 1985, there was essentially no growth; between 1985 and 1995, it doubled, and then doubled again in the next two years. This sort of growth makes consumers with assets in the stock market feel wealthy. Do they spend accordingly? Indeed they do, as we see from the results, where this variable has a mexval of 23. The variable increased by about $3000 between 1995.1 and 2000.3, thus increasing consumption per capita by $600 (= 3000*.20). During the same period, real savings per capita fell by about $600. Thus, essentially all of this much-publicized decline in saving may be explained by spending based on the rise in the stock market.

The combination of all these variables gives a virtually perfect fit to personal consumption. Given the number of explanatory variables we have used, what is more remarkable is that there was enough variability in the data to identify reasonable effects for all the variables once a soft constraint had pulled the coefficient on income below 1.0. When the equation was estimated over the period 1980.1 - 1994.1, however, no effect was found for the stock market variable. It becomes important only in the later years.

S&P 500 Stock Index in Real, Per Capita Terms

- sp500Rpc

Personal Consumption per capita

- Predicted   □ Actual
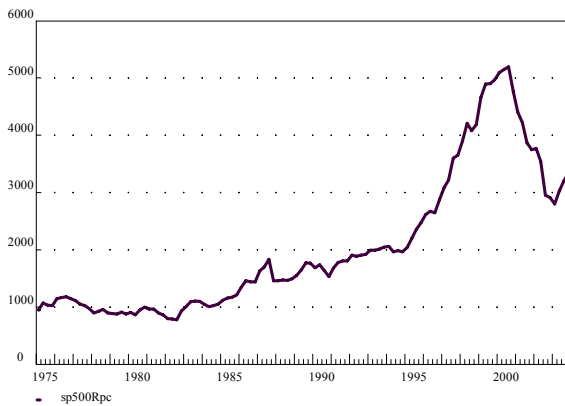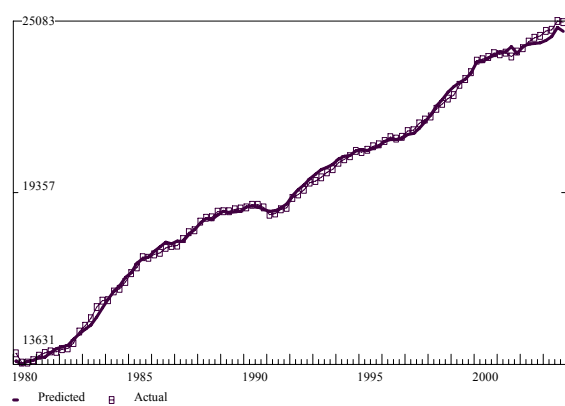
```
ti Personal Consumption per capita
fex cRpc = cR/pop

#Disposable Income per Capita
fex pidisaR = pidisa/gdpD
f yRpc = pidisaR/pop
f dyRpc = yRpc - yRpc[1]

# Interest necessary to maintain real value of assets
# First get inflex, expected inflation
fex lgdpD = 100.*@log(gdpD)
fex infl = lgdpD - lgdpD[4]
f rtbReal = rtb - infl
fex ub10 = @cum(ub10,1.0,.10)

# Expected inflation
f inflex = @cum(cinfl,infl[1],.10)/ub10
f intsavRpc = (inflex/(rtb+3.))*piint/(gdpD*pop)

# Stock Market
f sp500Rpc = sp500/(gdpD*pop)
f dsp500Rpc = sp500Rpc - sp500Rpc[1]

# Perceived real interest rate
f rtbexXdi = (rtb -inflex)*yRpc
# Contributions for Social Insurance
f consiRpc = nconsi/(gdpD*pop)

# savings in autos
f sauto = cdmvRpc - mvWearpc

# Interest paid by consumers to business
f piipcbRpc = piipcb/(gdpD*pop)
f rtbXyRpc = (rtb - 5.0)*yRpc[1]

# Unemployment rate
fex u = 100.*(lfc-emp)/lfc
f ur = 1./(1.+@pos(u-1.))
con 1000000 .95 = a2
sma 100000 a3 a7 1
```

```
:                     Personal Consumption per capita
  SEE    =      133.19 RSQ   = 0.9983 RHO =    0.70 Obser   =    96 from 1980.100
  SEE+1  =       98.79 RBSQ  = 0.9980 DW  =    0.60 DoFree  =    80 to   2003.400
  MAPE   =        0.55
    Variable name           Reg-Coef  Mexval  Elas   NorRes     Mean     Beta
  0 cRpc                    - - - - - - - - - - - - - - - 19443.30  - - -
  1 intercept              -672.86112     3.3  -0.03  634.92      1.00
  2 yRpc                      0.96865   651.1   1.07    4.06  21440.28    0.893
  3 dyRpc                    -0.42041    14.9  -0.00    3.98    111.71   -0.023
  4 dyRpc[1]                 -0.34773    18.4  -0.00    3.80    115.77   -0.018
  5 dyRpc[2]                 -0.28677    15.9  -0.00    3.66    114.67   -0.015
  6 dyRpc[3]                 -0.22002    11.5  -0.00    3.55    112.41   -0.012
  7 dyRpc[4]                 -0.12737     7.3  -0.00    3.38    109.69   -0.007
  8 consiRpc                  0.26276     1.5   0.03    2.92   2029.29    0.029
  9 sp500Rpc[1]               0.25205    24.4   0.03    1.98   2237.53    0.096
 10 dsp500Rpc[1]             -0.17250     1.4  -0.00    1.95     23.90   -0.008
 11 dsp500Rpc[2]             -0.15176     1.1  -0.00    1.92     22.39   -0.007
 12 sauto                     0.52789     3.2   0.00    1.89    163.14    0.019
 13 piipcbRpc                -1.57984     5.7  -0.04    1.16    552.14   -0.047
 14 intsavRpc                -0.42774     6.4  -0.03    1.16   1267.44   -0.038
 15 rtbexXdi                 -0.00368     5.9  -0.01    1.00  51345.60   -0.041
 16 ur[1]                  -390.51412     0.1  -0.00    1.00      0.17   -0.005

id cR = cRpc*pop
id c = cR*gdpD
```

In a conventional textbook on econometric methods, there is sure to be a chapter on simultaneous equation methods, methods for estimation when the dependent variable of an equation may influence one of the dependent variables. The essence of the problem is that, even if we know exactly the structure of the equations that describe the economy but they have random errors which we cannot observe, we may not get unbiased or even consistent estimates of the coefficients by applying least squares to the data. That is, even if we had an infinite number of observations, our estimates of the coefficients would not be right. The problem arises because, through another equation in the simultaneous system, an explanatory variable may be correlated with the error term in the equation in which it is an independent variable. The prime example is precisely income in the consumption equation, for if there is a large positive "disturbance" to the equation – a consumption spree – income will go up. This "backdoor" relationship between consumption and income would make the estimate of the coefficient on income tend to be too large, just as we experienced in our equation. This problem, known as *simultaneous equation bias*, or less correctly as *least squares bias,* was a major theoretical concern in the early days of econometric theory, and various ways were devised to avoid it. Some of these were known as *full-information maximum likelihood, limited-information maximum likelihood, two-stage least squares,* and *instrumental variables.*

How can this bias be avoided? One way -- the instrumental variable approach -- is to regress disposable income, *pidisa* in our case, on other variables not dependent on *c* in the same period, and then to use the predicted value from this regression instead of the actual *pidisa* in estimating the equation. The coefficient should be a lower bound of the true value. I regressed *pidisa* on itself lagged once and on current values of *v, g,* and *x.* When the soft constraint on the coefficient of *yRpc* was removed and predicted values of this equation were used in place of *pidisa* in the consumption equation, there was almost no effect on the coefficient of *yRpc.* In fact, it got slightly larger. Other coefficients were little affected. Thus, it does not appear that simultaneous

equation bias is the cause of the exaggerated value of this coefficient. This finding of rather minimal importance for the simultaneous equation problem is in line with the general experience of practical model builders working with quarterly data. In all other equations, we will use least squares with little concern about the problem. We shall return to the value of this coefficient in the following chapter on optimization.

Investment

Gross private domestic investment in Quest is treated in the four major parts available in even the aggregated version of the NIPA: Producers' durable equipment, Non-residential construction, Residential construction, and Change in business inventories.

The first and largest is investment in **Producers' durable equipment**. The term for replacement is familiar from the equation for investment in AMI. Two small changes have been made in the variable whose first differences are used to indicate the need for expansion investment: (1) it is gross *private* product, since it is being used to explain *private* investment, and (2) it is the @peak function of this variable. The @peak function is the highest value which the variable has ever had up to and including the present. The use of the @peak function makes little difference in estimating the equation, but it makes the model more stable, since the first difference terms cannot go negative in a downturn. Notice the strong positive transient or "splurge" effect of an increase in output. This behavior makes equipment investment one of the primary generators of cycles in the economy.

The real interest rate used is the difference between the Treasury bill rate and the rate of inflation in the GDP deflator. Its mean value is about 2.0, and this mean has been subtracted so that the variable just shows the fluctuations about the mean. This variable is then multiplied by the replacement term divided by its mean, so the amplitude of the fluctuations in the variable will grow more or less in line with the growth of the dependent variable. A change of one percentage point will, when replacement is at its mean, change this variable by one unit. Thus, a reduction of the real interest rate by one percentage point, say from 3 to 2 — a big change -- will increase investment by about $5 billion (.46+1.96+2.23 = 4.65), or a little less than 1.0 percent of its mean value over this period. For an effect that dominates macroeconomics books (via the IS curve), its quantitative importance is embarrassingly small.

The stock market variable is relevant to this equation because it affects the perceived cost of funds to firms. Firms can raise funds for capital investment by selling additional shares, but the profits must then be spread over a larger number of shares and, if a particular individual or group exercises control over the company through the number of shares it holds, it may well be reluctant to see that control weakened by issuing new shares to outsiders. These objections, however, may be overcome if the stock price is high so that a lot of capital is raised with little dilution of ownership. While this effect has long been recognized as possible, it has become practically important only since 1995. Our variable, sp500R, rose by about 700 between 1996.1 and its peak in 2000.3.   According to our equation, this rise adds about $35 billion (= .05*700)

11

to annual investment, and the transient effect may be even larger. Without the use of this variable, the equation fits fine up through 1994, but then falls substantially short.

Equipment Investment
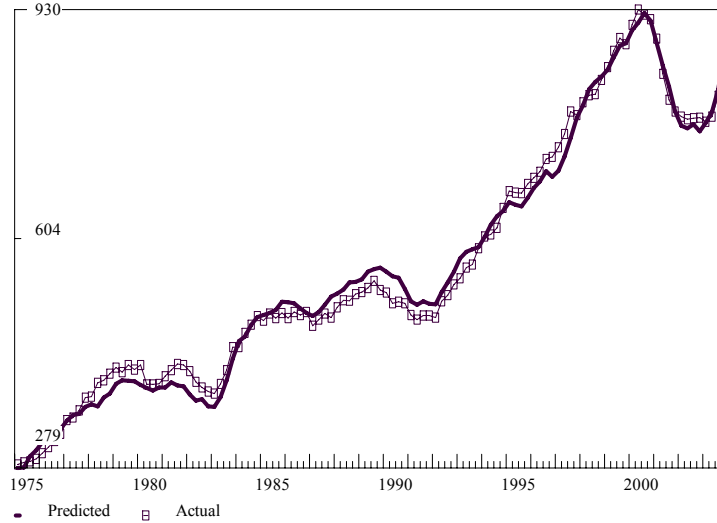
```
ti Equipment Investment
f gppR = (gdp - gdpg)/gdpD
f pgppR = @peak(pgppR,gppR,.0)
f d = pgppR - pgppR[1]
f ub05 = @cum(ub05,1.0,.05)
# Equipment replacement, used also in Labor Productivity
f repEq = @cum(stockEq,vfnreR[4],.05)/ub05
# Compute real interest rate
fex lgdpD = 100.*@log(gdpD)
fex infl = lgdpD - lgdpD[4]
fex ub10 = @cum(ub10,1.,.10)
# inflex is expected inflation
fex inflex = @cum(cinfl,infl[1],.10)/ub10
f rtbReal = rtb - infl
f rrXrepe = (rtbReal-2.)*(repEq/400.)
fex sp500R = sp500/gdpD
f dsp500R = sp500R - sp500R[1]
f ue = lfc -emp
con 10000 1 = a2
sma 1000 a3 a13 1
sma 1 a14 a16 1
sma 100 a18 a24 1
```

```
:                          Equipment Investment
   SEE   =      18.13 RSQ  = 0.9893 RHO =   0.84 Obser  =  116 from 1975.100
   SEE+1 =       9.91 RBSQ = 0.9865 DW  =   0.32 DoFree =   91 to   2003.400
   MAPE  =       3.04
     Variable name          Reg-Coef  Mexval  Elas   NorRes    Mean    Beta
  0 vfnreR                  - - - - - - - - - - - - - - -   561.29  - - -
  1 intercept              74.82013     6.8   0.13  118.17     1.00
  2 repEq                   0.97663   474.9   0.81   10.22   462.83   0.798
  3 d[1]                    0.18058     7.2   0.01    9.33    45.96   0.043
  4 d[2]                    0.20025    18.2   0.02    7.99    44.41   0.046
  5 d[3]                    0.19895    17.2   0.02    6.94    44.06   0.046
  6 d[4]                    0.18103    15.5   0.01    5.95    43.66   0.042
  7 d[5]                    0.16109    13.6   0.01    5.15    43.44   0.037
  8 d[6]                    0.13382     9.8   0.01    4.60    42.76   0.031
  9 d[7]                    0.11025     6.4   0.01    4.23    42.48   0.026
 10 d[8]                    0.08888     4.0   0.01    4.00    42.83   0.021
 11 d[9]                    0.07242     2.6   0.01    3.86    43.37   0.017
```

```
12  d[10]                           0.05775     1.6    0.00    3.79    43.63   0.013
13  d[11]                           0.03692     0.8    0.00    3.73    44.48   0.008
14  rrXrepe[7]                     -0.45730     0.1   -0.00    3.18     0.53  -0.007
15  rrXrepe[8]                     -1.96550     3.8   -0.00    2.94     0.56  -0.027
16  rrXrepe[9]                     -2.23064     3.5   -0.00    2.62     0.61  -0.030
17  sp500R[1]                       0.04911    13.3    0.05    1.93   530.99   0.103
18  dsp500R[1]                      0.10957     2.2    0.00    1.78     6.26   0.024
19  dsp500R[2]                      0.14629     4.9    0.00    1.62     5.40   0.032
20  dsp500R[3]                      0.15877     5.6    0.00    1.42     4.63   0.034
21  dsp500R[4]                      0.13025     3.7    0.00    1.30     4.63   0.028
22  dsp500R[5]                      0.15558     5.4    0.00    1.15     4.55   0.034
23  dsp500R[6]                      0.08072     1.3    0.00    1.12     5.82   0.016
24  dsp500R[7]                      0.02109     0.1    0.00    1.12     6.02   0.004
25  ue                             -7.19462     5.7   -0.10    1.00     7.70  -0.054
```

Investment in **Non-residential construction** — stores, office buildings, industrial plants, pipelines, churches, hospitals, airports, parking lots, and so on — is one of the hardest series to explain. Even the booming economy of the late 1990's barely brought it back to the levels it reached in the recession years of the early 1980's. Our equation is motivated by the idea that investment is proportional to the difference between the desired stock and the actual stock of structures, and that the desired stock is a linear function of the real Gross private product, gppR. Thus, the basic idea is that

$$vfnrsR = \lambda(a + b *gppR - StockSt)$$

where *vfnrsR* is real investment in non-residential construction, and *StockSt* is the stock of those structures. Several depreciation rates have been tried for calculating the stock of structures without much effect on the fit of the equation. One percent per quarter was chosen. By introducing lagged values of the first difference of *gppR*, the desired level of the stock is allowed to rise gradually following an increase in *gppR*.

The natural variable to add next is some sort of interest rate. These all had positive — wrong — signs with lags of three years or less. The real rate with a lag of 16 quarters has been left more or less as a reminder of the perverse results with shorter lags. This strong positive relation with interest rates suggested using interest income, which, indeed proved somewhat helpful. The reasoning is that persons with significant amounts of interest income might be likely to investment in real estates.

The rates of change of the stock market value variable — but not its level — also proved helpful. This variable may be measuring optimism about the future of the economy.

Finally, a special dummy variable was introduced for the period between the 1981 and the 1986 tax acts. The 1981 act allowed passive partners in real estate development (as well as active partners) to count paper depreciation at double declining balance rates against their ordinary income. Investors looking for tax shelters poured billions of dollars into non-residential construction. The 1986 act repealed this provision for non-residential construction. It did not even "grandfather" in the buildings that had been built while the 1981 act was in force. Thus, many investors who had bought tax shelters found themselves with more or less worthless holdings. Though the 1986 act was not passed until the middle of the year, its passage was anticipated, and investment was cut back for the beginning of the year.

vfnrsR - Non-residential Structures
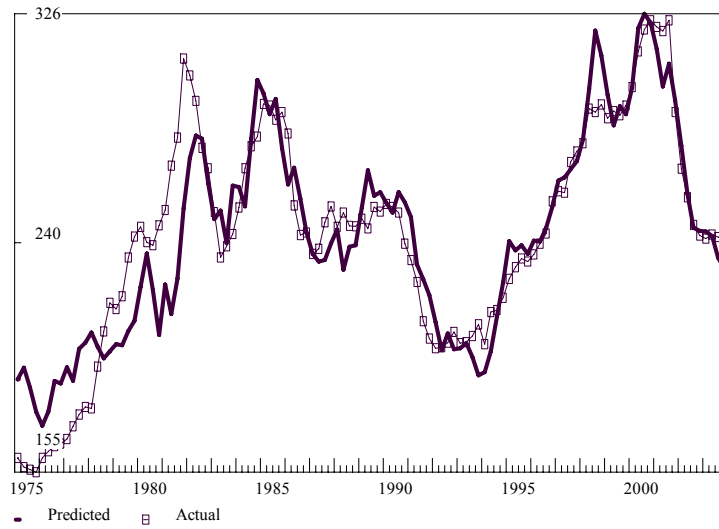

```
ti vfnrsR – Non-residential Structures
fex gppR = (gdp – gdpg)/gdpD
f pgppR = @peak(pgppR,gppR,.0)
f d = pgppR – pgppR[1]
f ub01 = @cum(ub01,1.,.01)
f StockSt = 100.* @cum(cumSt,0.25*vfnrsR[4],.01)/ub01
# Compute real interest rate
fex lgdpD = 100.*@log(gdpD)
fex infl = lgdpD – lgdpD[4]
fex ub10 = @cum(ub10,1.,.10)
# inflex is expected inflation
fex inflex = @cum(cinfl,infl[1],.10)/ub10
fex rtbReal = rtb – infl
f npiniR= piint/gdpD
# 1987 Tax Act
# The stimulus of the 1981 tax act is here shown as beginning in 1982.
# The 1986 repeal of the tax shelters created by the 1981 act was retro-
# active to the beginning of 1986, and this fact was apparently anticipated.
fex taxacts = 0
update taxacts
1982.1  1 1 1 1  1 1 1 1  1 1 1 1  1 1 1 1;
spr taxacts

fdup sp500R = sp500/gdpD
fdup dsp500R = sp500R – sp500R[1]
```

14

```
:                       vfnrsR - Non-residential Structures
  SEE    =      16.86 RSQ   = 0.8285 RHO =   0.78 Obser   =  116 from 1975.100
  SEE+1  =      10.81 RBSQ  = 0.8066 DW   =   0.44 DoFree =  102 to   2003.400
  MAPE   =       5.42
     Variable name           Reg-Coef  Mexval  Elas   NorRes      Mean   Beta
  0 vfnrsR                   - - - - - - - - - - - - - - - -    243.28 - - -
  1 intercept                445.37839   66.3   1.83    5.83      1.00
  2 gppR[4]                     0.04437   43.9   1.10    4.26   6034.55  1.779
  3 d[4]                       -0.05408    0.5  -0.01    3.92     43.66 -0.054
  4 StockSt[1]                 -0.16710   38.8  -3.17    3.92   4616.29 -3.077
  5 taxacts                    31.07698   11.4   0.02    2.85      0.14  0.263
  6 npiniR[1]                   0.76485   16.6   2.38    1.10    756.09  4.021
  7 npiniR[2]                  -0.37319    3.8  -1.15    1.04    751.12 -1.994
  8 rtbReal[16]                 0.54140    0.2   0.00    1.04      2.17  0.040
  9 dsp500R[3]                 -0.06849    0.8  -0.00    1.03      4.63 -0.064
 10 dsp500R[4]                  0.01171    0.0   0.00    1.03      4.63  0.011
 11 dsp500R[5]                  0.00691    0.0   0.00    1.03      4.55  0.006
 12 dsp500R[6]                  0.06682    0.5   0.00    1.01      5.82  0.057
 13 dsp500R[7]                  0.06223    0.4   0.00    1.00      6.02  0.052
 14 dsp500R[8]                 -0.01880    0.0  -0.00    1.00      5.88 -0.016
```

Investment in **Residential constuction**, quite in contrast to non-residential construction, proves to be quite sensitive in the proper, negative direction to interest rates. Otherwise, the approach to the equation is similar except that a combination of disposable income and the stock market value is presumed to determine the desired stock.

Residential Construction

Predicted   Actual
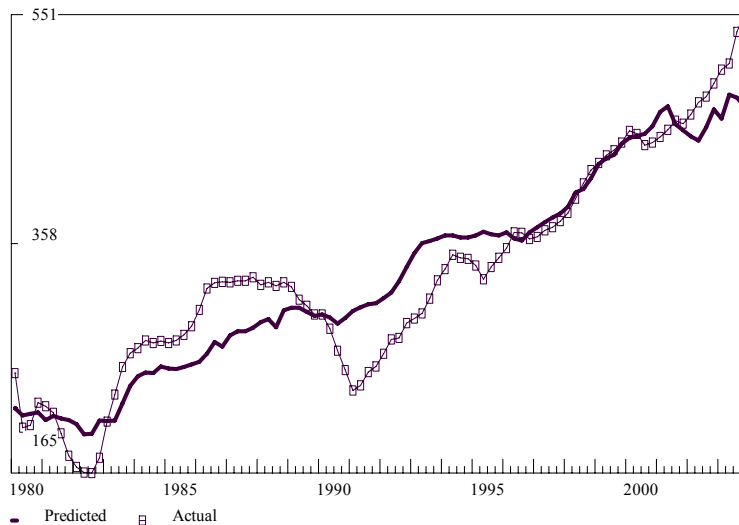
```
ti Residential Construction
fex lgdpD = 100.*@log(gdpD)
fex infl = lgdpD - lgdpD[4]
fex ub10 = @cum(ub10,1.0,.10)
freq ub10   4
# inflex is expected inflation
fex inflex = @cum(cinfl,infl[1],.10)/ub10
fex rtbex = rtb - inflex
f ub01 = @cum(ub01,1.,.01)
f StockHouse = 100.*@cum(cvfrR,0.25*vfrR[2],.01)/ub01
f pidisaR = pidisa/gdpD
f dpidisaR = pidisaR - pidisaR[1]
fdup sp500R = sp500/gdpD
sma 100 a7 a11 1
:                       Residential Construction
  SEE   =      31.12 RSQ  = 0.8801 RHO =   0.94 Obser  =   96 from 1980.100
  SEE+1 =      11.28 RBSQ = 0.8660 DW  =   0.12 DoFree =   85 to   2003.400
  MAPE  =       8.52
    Variable name            Reg-Coef  Mexval  Elas   NorRes     Mean   Beta
  0 vfrR                     - - - - - - - - - - - - - - - -    330.61 - - -
  1 intercept                68.98561    0.4   0.21    8.33      1.00
  2 pidisaR[4]                0.05780    4.5   0.94    1.15   5396.28  0.758
  3 dpidisaR[4]              -0.00031    0.0  -0.00    1.15     42.34 -0.000
  4 dpidisaR[5]               0.06970    0.5   0.01    1.14     42.58  0.035
  5 sp500R[3]                 0.05617    3.5   0.10    1.11    580.96  0.233
  6 StockHouse               -0.01083    0.1  -0.19    1.10   5863.10 -0.100
  7 rtbex[4]                 -2.93275    4.3  -0.02    1.10      2.68 -0.055
  8 rtbex[5]                 -2.25770    4.9  -0.02    1.09      2.71 -0.041
  9 rtbex[6]                 -1.60535    4.0  -0.01    1.05      2.73 -0.029
 10 rtbex[7]                 -1.00449    2.4  -0.01    1.02      2.73 -0.018
 11 rtbex[8]                 -0.47225    1.2  -0.00    1.00      2.74 -0.008
```

Finally, investment in **Change in business inventories** is unchanged from the AMI model but is repeated here for completeness.

viR Change in Inventory

Predicted   Actual
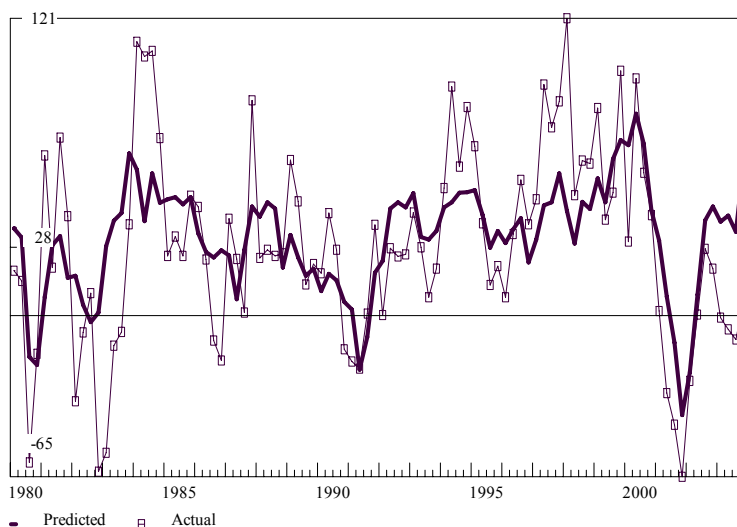
```
title viR Change in Inventory
# fs stands for "final sales"
f fsR = cR + vfR + xR + gR
f dfsR = fsR - fsR[1]
sma 1000 a1 a4 1
:                          viR Change in Inventory
  SEE   =       30.20 RSQ  = 0.4299 RHO =   0.50 Obser   =   96 from 1980.100
  SEE+1 =       26.31 RBSQ = 0.4113 DW  =   1.00 DoFree =   92 to   2003.400
  MAPE  =      301.43
   Variable name            Reg-Coef  Mexval  Elas   NorRes     Mean   Beta
  0 viR                     - - - - - - - - - - - - - - -      28.79 - - -
  1 dfsR[1]                 0.25451   14.3   0.58    1.34      65.84
  2 dfsR[2]                 0.16052   12.2   0.36    1.02      65.22  0.230
  3 dfsR[3]                 0.04982    1.1   0.11    1.00      64.86  0.072
  4 dfsR[4]                 0.01940    0.2   0.04    1.00      63.33  0.028
```

Exports, Imports, and the Terms of Trade

The natural economic variable to use in explaining imports or exports is the domestic price over the foreign price for similar goods, the *terms of trade* for that product. Earlier versions of Quest used a terms of trade variable computed from the overall import deflator relative to the domestic prices of tradable final demand goods. It never worked very well and was hard to model. In this revision, I looked at the import deflator relative to the export deflator for all the major categories of traded goods. The graph below shows the results for three typical product groups. Clearly, there is little or no similarity among them. There is no possibility of finding a single index to represent then all and equally little possibility to explain such different series with similar equations using the same macroeconomic explanatory variables. I have therefore given up on explaining and using a terms of trade variable. Instead, we will use directly in the export and import equations the variables that might have been used to explain terms of trade.

Alternative Terms of Trade
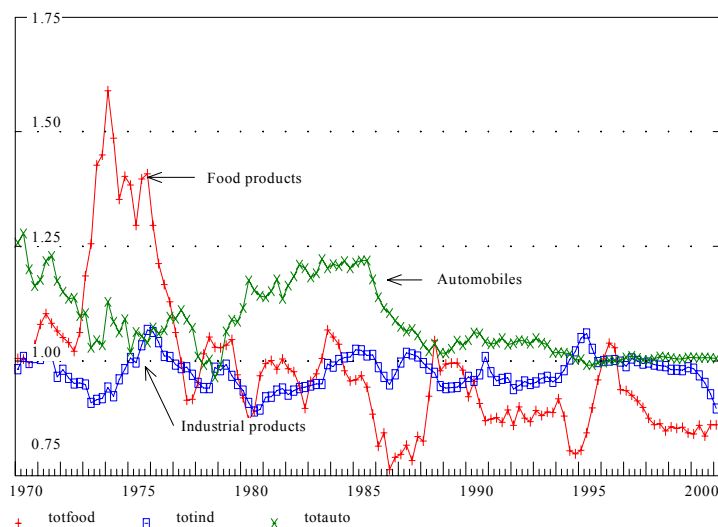
In the 2004 version of Quest, real exports are exogenous. The following section, from an earlier version of the model may be of interest.

The primary variable in the explanation of **exports** is foreign demand, *fgndem*. This variable, a by-product of the Inforum International System of multisectoral models, is a combination of the real imports of the major trading partners of the United States, weighted together with their shares in U.S. exports in 1992. The dependent variable of our equation, *xRat,* is the logarithm of the ratio of our real exports to this variable. The unemployment rate enters the explanation because at times of low unemployment U.S. firms may not be able give good prices or delivery times to foreign customers, who then turn elsewhere for suppliers. Consequently, a high unemployment rate should make for a high *xRat*. Our result shows that a one percentage point increase in our unemployment rate increases our exports by over 5.0 percent. The real interest rate can be important, because at times of high interest rates, foreigners buy dollars to get the high interest, thus running up the value of the dollar and limiting U.S. exports. According to our equation, a one point increase in the real interest rate can decrease exports by 3.7 percent. A similar argument applied to the stock market. A strong market attracts foreign investors, who buy dollars to buy American stocks, thereby pushing up the dollar and making it difficult for U.S. manufacturers to compete abroad. The variable used for the stock market is the S&P 500 index relative to nominal GDP; this variable has roughly the same value today as it did forty years ago, and thus appears to be stationary. The final variable used, *d80*, is a dummy which assumes positive values only in the period from 1979 to 1982. During this period, the fit of the equation without *d80* had large positive errors in that period. I was unable to find a variable to eliminate these errors but added *d80* so that simulations of the model beginning with 1980 would not start off with large errors in exports. This equation was fit over a rather long period because fitting from 1980 forward gave a wrong sign on real interest variable; the experience of the 1970's is necessary for the program to be able to find the logically correct relationship.

Exports Relative to Foreign Demand


```
ti Exports Relative to Foreign Demand
fex xRat = @log(feR/fgndem)

f pgdp = fe+vf+c
f sprat = sp500/pgdp
fex d80 = 0
update d80
1979.1 .2 .4 .6  .8
1980.1  1. 1. 1. 1.  1. 1. 1. 1.   .8 .6 .4 .2
```

```
:                     Exports Relative to Foreign Demand
  SEE   =        0.11 RSQ   = 0.6982 RHO =   0.89 Obser  =   125 from 1970.100
  SEE+1 =        0.06 RBSQ  = 0.6882 DW  =   0.23 DoFree =   120 to    2001.100
  MAPE  =        9.57
    Variable name             Reg-Coef  Mexval  Elas   NorRes      Mean   Beta
  0 xRat                      - - - - - - - - - - - - - - - -      0.97  - - -
  1 intercept                  0.80538    25.2   0.83     3.31     1.00
  2 unemp[1]                   0.05037     9.1   0.33     2.65     6.32   0.356
  3 rtbReal[1]                -0.03674    33.0  -0.08     2.07     2.15  -0.528
  4 sprat[1]                  -1.44963     3.2  -0.11     1.73     0.08  -0.210
  5 d80                        0.39038    31.6   0.04     1.00     0.10   0.521

id feR = @exp(xRat)*fgndem
```

The equation for **imports** is similar but uses components of aggregate demand, consumption, investment, and exports in place of the foreign demand variable. Because these different demand components may have different import content, the shares of two of them, exports and investment, in the total are used as explanatory variables and prove to have positive effects, that is, they are more import-intensive than is the third component, consumption. The the stock market index is included here for the same reason as it was included in the export equation, though with the opposite expected sign.

Import Ratio to Private Demand

ti Import Ratio to Private Demand

```
# Compute real interest rates
fex lgdpD = 100.*@log(gdpD)
fex infl = lgdpD - lgdpD[4]
f rtbReal = rtb - infl

f pgdp = x+vf+c
f imprat = 100.*m/pgdp
f xrat = x/pgdp
f vfrat = vf/pgdp
f sprat = sp500/pgdp
```

```
:                          Import Ratio to Private Demand
  SEE    =         0.66 RSQ   = 0.7663 RHO  =    0.88 Obser  =    96 from 1980.100
  SEE+1  =         0.33 RBSQ  = 0.7587 DW   =    0.23 DoFree =    92 to   2003.400
  MAPE   =         3.97
    Variable name            Reg-Coef  Mexval  Elas   NorRes     Mean     Beta
  0 imprat                   - - - - - - - - - - - - - - -     12.52  - - -
  1 intercept                10.10427   24.7   0.81     4.28    1.00
  2 xrat                      3.55243    0.1   0.03     3.28    0.10   0.031
  3 vfrat                    -4.58474    0.4  -0.06     3.25    0.17  -0.047
  4 sprat[1]                 35.85794   80.2   0.23     1.00    0.08   0.844


id m = imprat*pgdp/100.
id mR = m/gdpD
```

Productivity, Employment, and Unemployment

As an exercise in Chapter 3, we added to the original AMI model and equation for employment which simply regressed employment on real Gross domestic product. Implicitly, this equation made all the growth in productivity depend on the growth in real GDP. Here we need to examine that growth more closely.

First of all, we need to note that our employment variable, *emp*, is civilian employment and does not count members of the military. As far as I can see, people in the military do not exist for the Bureau of Labor Statistics (BLS). All of the familiar data on labor force, employment, and unemployment statistics are for civilians only. I have been unable to find a BLS series on military employment. The right way to handle this problem would be to construct a quarterly series on military employment and use it to convert all of the BLS series to a total labor force basis. The difficulty of maintaining this series, however, and the loss of comparability with familiar BLS statistics has led me to go into the other direction, namely, to deduct real compensation of the military – which is readily available in the NIPA – from gdpR to get gdpcR, real civilian GDP and to use it to explain civilian employment.

Our dependent variable will therefore be the logarithm of gross civilian labor productivity, real civilian GDP divided by civilian employment. Regressed simply on *time*, over the period 1980.1 - 2001.1, the coefficient on *time* is .01716, that is, 1.7 percent per year. Besides time, however, there are at least two other factors readily available which should be tried. From the investment equation, we have available the stock of equipment from which we can make up a capital-output ratio. This ratio was more volatile than the dependent variable, so it was smoothed. To avoid spurious correlation from having real GDP in the denominator of both variables, we have used only lagged values in this variable, *capouts*.

Another factor is real GDP itself. It could influence productivity by economies of scale and by the opportunities which growth gives to eliminate inefficiencies without the painful process of laying off workers. When it was introduced into the equation, it was very successful; and the coefficient on *time* fell to only .00473. There is, however, a problem with this variable, for it occurs in the numerator of the dependent variable. Thus, any random fluctuation in it will show up automatically as a similar fluctuation in productivity. Thus, if we are really looking for long-

20

term relations, the *gdpR* variable may get too high a coefficient relative to the *time* variable. To control for this situation, the equation was run with gdpR[1] as the most recent value of this variable. The coefficient on *time* rose to .00687. We then constrained the coefficient at that value, restored the use of the current value of *gdpR,* and re-estimated the equation.

Fluctuations in productivity are explained largely by the lagged values of the percentage change in real GDP, here calculated as the first difference of the logarithm. Notice the big surge in productivity which follows an increase in real GDP. It is initally produced by existing employees simply working harder and longer and perhaps by some postponable work simply being postponed. Gradually, however, employment is brought up to the levels appropriate for the level of output. For every 1 percent increase in real GDP, we find an increase of 0.32 percent in productivity.

Labor Productivity
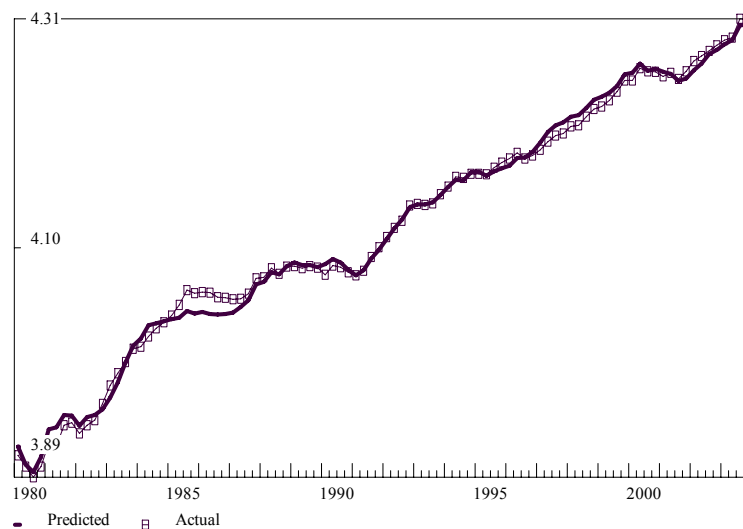

```
ti Labor Productivity
# Military compensation in real terms
fex gfdccemR = gfdccem/gdpD
# Create Civilian GDP
f gdpcR = gdpR - gfdccemR
fex lLabProd = @log(gdpcR/emp)
f lgdpcR = @log(gdpcR)
f pcGdpcR = lgdpcR - lgdpcR[1]
fdup repEq = @cum(stockEq,vfnreR[4],.05)/ub05
f pgdpcR=@peak(pgdpcR,gdpcR,.0)
f capout = repEq/pgdpcR
f lcapouts = @log(.5*capout[1]+.3*capout[2]+.2*capout[3])
sma .001 a4 a11 1
con 100 .00623 = a2
```

```
:                          Labor Productivity
  SEE   =       0.01 RSQ   = 0.9957 RHO =    0.90 Obser   =   96 from 1980.100
  SEE+1 =       0.00 RBSQ  = 0.9951 DW  =    0.20 DoFree  =   84 to   2003.400
  MAPE  =       0.14
    Variable name           Reg-Coef  Mexval  Elas   NorRes     Mean   Beta
  0 lLabProd                - - - - - - - - - - - - - - -     4.12  - - -
  1 intercept               1.37790   109.7   0.33   296.48     1.00
  2 time                    0.00622   740.2   0.05    61.32    32.12   0.383
  3 lgdpcR                  0.30842   599.1   0.67     2.69     8.90   0.605
  4 pcGdpcR                 0.46350    28.4   0.00     2.65     0.01   0.032
  5 pcGdpcR[1]              0.43106    46.8   0.00     2.44     0.01   0.030
  6 pcGdpcR[2]              0.39895    51.7   0.00     2.09     0.01   0.028
  7 pcGdpcR[3]              0.36553    43.0   0.00     1.80     0.01   0.025
  8 pcGdpcR[4]              0.32600    33.3   0.00     1.58     0.01   0.023
  9 pcGdpcR[5]              0.27418    25.0   0.00     1.44     0.01   0.020
 10 pcGdpcR[6]              0.20480    18.1   0.00     1.33     0.01   0.015
 11 pcGdpcR[7]              0.11218    12.0   0.00     1.19     0.01   0.008
 12 lcapouts                0.08169     9.0  -0.05     1.00    -2.71   0.037

f LabProd = @exp(lLabProd)
id emp = gdpcR/LabProd
```

With labor productivity known, **employment** is just computed by dividing real GDP by it; **unemployment** is computed by subtracting employment from the labor force.

Interest rates

The key to obtaining a somewhat satisfactory explanation of the interest rate was to use as the dependent variable the "expected" or "perceived" real interest rate — the nominal rate on 90-day Treasury bills minus the expected rate of inflation. The sole explanatory variable is the velocity of M1 together with lagged values of its first difference, and it product with time. The negative coefficient on the product of velocity and time indicates a gradual reduction in the requirements for M1. The positive signs on the first differences indicate that the immediate impact on interest rates of a change in money supply relative to GDP is substantially greater than the long-term impact. Seemingly, the financial institutions adjust to the available money supply. During an earlier period, M2 would have been the appropriate measure of money; but during the period studied here, it has little value in explaining interest rates.

Treasury Bill Rate


```
ti Treasury Bill Rate
f lgdpD = 100.*@log(gdpD)
f infl = lgdpD - lgdpD[4]
fex ub10 = @cum(ub10,1.0,.10)
freq ub10   4
# inflex is expected inflation
f inflex = @cum(cinfl,infl[1],.10)/ub10
fex rtbex = rtb - inflex
f v1 = gdp/m1
f dv1 = v1 - v1[1]
sma .1 a3 a7 1
:                        Treasury Bill Rate
  SEE   =       0.84 RSQ  = 0.7858 RHO =   0.66 Obser  =   92 from 1981.100
  SEE+1 =       0.63 RBSQ = 0.7679 DW  =   0.67 DoFree =   84 to   2003.400
  MAPE  =      93.71
    Variable name           Reg-Coef  Mexval  Elas    NorRes     Mean   Beta
  0 rtbex                   - - - - - - - - - - - - - - - -      2.49 - - -
  1 intercept               -0.17176     0.0  -0.07     4.64     1.00
  2 v1                       1.29362    21.6   3.81     4.50     7.34  0.625
  3 dv1                      4.11350    32.4   0.03     4.41     0.02  0.255
  4 dv1[1]                   3.43991    53.7   0.03     3.98     0.02  0.215
  5 dv1[2]                   2.75072    42.9   0.02     3.48     0.02  0.173
  6 dv1[3]                   1.95052    22.8   0.01     3.16     0.02  0.122
  7 dv1[4]                   1.02634    11.4   0.01     2.83     0.02  0.064
  8 time*v1                 -0.02910    68.3  -2.84     1.00   243.19 -1.195

ti rtb positivity
id rtb = rtbex + inflex
cc if(rtb< 0.5) rtb = 0.5;
```

The Income Side of the Accounts

To understand the connections and relevance of the remaining equations, one needs to recall the basic identities of the income side of the NIPA. In the following quick review, the items for

which regression equations have been developed are shown in bold. All other items are either determined either by identities or by behavioral ratios or are left exogenous.

```
#gnp — gross national product
#   + gdp      Gross domestic product
#   + irrow    Income receipts from the rest of the world
#   - iprow    Income payments to the rest of the world
#   = gnp      Gross national product

id gnp = gdp + irrow - iprow

# Net National Product

#    + gnp     Gross national product
#    - ncca    Capital consumption allowances with IVA and CCAdj
#    = nnp     Net national product

id nnp = gnp - ncca

# ni  -- National income  — from the product side
#    + nnp      Net national product
#    - nsd      Statistical discrepancy
#    = ni       National income

id ni = nnp - nsd

# The alternative, income-side definition of national income.

#    + niceprop Compensation of employees and Proprietor income
#    + niren    Rental income
#    + niprf    Corporate profits
#    + netint   Net interest
#    + nmiscpay Misc. payments (rents & royalties)
#    + nitpi    Taxes on production and imports
#    - nisub    Less: Subsidies
#    + nbctpn   Business current transfer payments
#    + nisurp   Current surplus of government enterprises
#    = ni       National income


# pi — Personal Income
#    + ni       National income
#    - niprf    Corporate profits with IVA and CCA
#    + pidiv    Personal dividend income
#    - netint   Net interest
#    + piint    Personal interest income
#    - nconsi   Contributions for social insurance
#    + ngtpp    Government transfer payments to persons
#    - nbctpn   Business current transfer payments (net)
#    + nibctpnp Business current transfer payments to persons
#    - nsurp    Surplus of government enterprises
#    - nwald    Wage accruals less disbursements
#    - nitpils  Taxes on production and imports less subsidies
#    + pigsb    Government social benefits to persons
#    = pi       Personal income


# npini — Personal interest income
```

```
# npini = + netint   Net interest
#         + gfenip   Net interest paid by the Federal government
#         + gsenip   Net interest paid by state and local governments
#         + piipcb   Interest paid by consumers to business
```

Notice that we have two different definitions of National income, one derived from GDP and one from adding up the five types of factor income which compose it. We will compute it both ways but scale the positive components of the income definition to match the product definition.

In all, there are eight different items to be determined by regression: Capital consumption allowances, four components of National income, Personal dividend income, and two Net interest payments by government. One other item, Interest paid by consumers to business, has already been discussed.

Capital consumption allowances

The computation of capital consumption allowances was explained in Chapter 1. Here we are seeking just a rough approximation of this process. We divide investment into two types: equipment and structures. For each, we set up a two-bucket wear-out system. For equipment, both buckets have a spill rate of 5 percent per quarter; for structures, both buckets have a spill rate of 1 percent per quarter.   The weights on the spill streams from the two equipment buckets are softly constrained to add to 1.0, as are the weigts on the spill streams from the two structures buckets. Finally, a variable called *disaster* allows for the exceptional capital consumption by hurricane Andrew and by the Los Angeles earthquake of 1994. The fit was extremely close.

ncca -- capital consumption allowance

Predicted   Actual
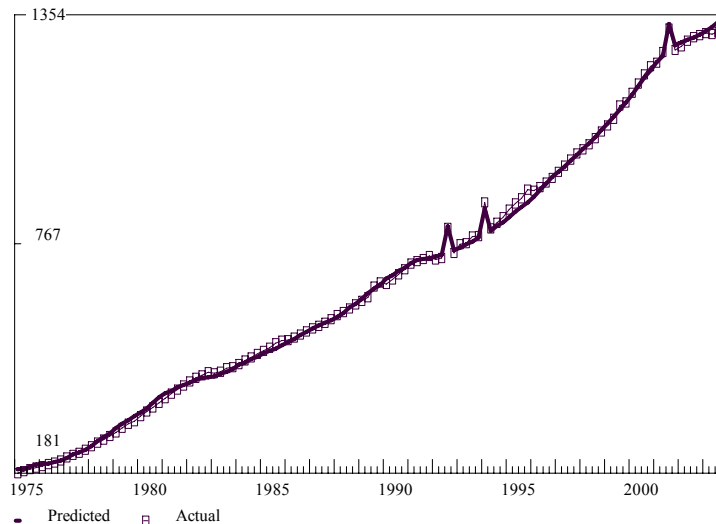
```
ti ncca -- capital consumption allowance
# Wearout of Equipment
f ub05 = @cum(ub05,1.,.05)
f repEq1R = @cum(c1vfnreR,vfnreR,.05)/ub05
f repEq2R = @cum(c2vfnreR,repEq1R,.05)/ub05

# Equipment wearout in current prices
f repEq2 = repEq2R*gdpD
f repEq1 = repEq1R*gdpD

# Wearout of Structures
f ub01 = @cum(ub01,1.,.01)
f vfsR = vfrR + vfnrsR
f repSt1R = @cum(c1vfsR,vfsR,.01)/ub01
f repSt2R = @cum(c2vfsR,repSt1R,.01)/ub01

# Structure wearout in current prices
f repSt1 = repSt1R*gdpD
f repSt2 = repSt2R*gdpD

fex disaster = 0
# disaster 92.3 = Hurricane Andrew;  94.1 = L.A. earthquake
# 01.3 World Trade Center attack
update disaster
1992.3 1 0 0 0 0  1;
update disaster
2001.3 1;
con 500 1 = a2 + a3
con 500 1 = a4 + a5

:                     ncca -- capital consumption allowance
  SEE   =       9.18 RSQ  = 0.9993 RHO =   0.78 Obser =  116 from 1975.100
  SEE+1 =       5.96 RBSQ = 0.9992 DW  =   0.45 DoFree =  110 to   2003.400
  MAPE  =       1.27
    Variable name          Reg-Coef  Mexval  Elas   NorRes    Mean    Beta
 0 ncca                   - - - - - - - - - - - - - - -   683.74 - - -
 1 intercept              -36.10743    45.9  -0.05 1232.14    1.00
 2 repEq1                   1.13638    93.5   0.65   15.03   392.17   0.708
 3 repEq2                  -0.10752     0.8  -0.05   14.32   333.61  -0.056
 4 repSt1                   0.14913     0.5   0.07    2.50   331.75   0.058
 5 repSt2                   0.98660    15.3   0.38    2.15   262.40   0.288
 6 disaster                66.22402    46.6   0.00    1.00     0.03   0.031
```

Components of national income

Compensation of employees and **Proprietor income** are modeled together since our
employment variable does not separate employees from proprietors. The ratio of the
combination to total employment gives earnings per employed person, which, when put into real
terms, is regressed on labor productivity and the unemployment rate. Since employment appears
in the denominator of both the dependent and independent variables, I checked for spurious
correlation by using only lagged values of labor productivity. The coefficient on labor
productivity actually rose slightly, so there is little reason to suspect spurious correlation. The
use of the unemployment variable in this equation is a mild infraction of the rule against using a
stationary variable to explain a trended one, but percentage-wise the growth in the dependent

variable has not been great in recent years. Both the dependent variable and labor productivity are in logarithmic terms, so the regression coefficient is an elasticity. This elasticity turns out to be slightly less than 1.0. Note that while the mexvals on the two lagged values of the unemployment rate are both very small, the combined effect, as seen in the NorRes column, is substantial.

Real Earnings per Employed Person

- Predicted   ☐ Actual

```
ti Real Earnings per Employed Person
fex lwageR = @log(((nice+niprop)/emp)/gdpD)

:                           Real Earnings per Employed Person
  SEE   =        0.01 RSQ   = 0.9909 RHO =   0.91 Obser  =  116 from 1975.100
  SEE+1 =        0.00 RBSQ  = 0.9906 DW  =   0.18 DoFree =  111 to   2003.400
  MAPE  =        0.25
    Variable name            Reg-Coef  Mexval   Elas    NorRes     Mean    Beta
  0 lwageR                   - - - - - - - - - - - - - - - -      3.66  - - -
  1 intercept                -0.07523     1.0  -0.02    110.15     1.00
  2 lLabProd                  0.71342     7.4   0.80      1.40     4.08   0.736
  3 lLabProd[1]               0.21058     0.6   0.23      1.33     4.08   0.218
  4 u[3]                     -0.00307     0.3  -0.01      1.01     6.40  -0.035
  5 u[4]                     -0.00298     0.3  -0.01      1.00     6.40  -0.034

f nicepro = @exp(lwageR)*emp*gdpD
```

Rental income is the smallest component of national income. It is the income of persons (not corporations) from renting out a house, a room or two in a house, or a commercial property. In particular, in includes the net rental income imputed to owner-occupants of houses, that is, the imputed space rental value less mortgage interest, taxes, and upkeep expenses. In view of this content, it is not surprising that the stock of houses should be one of the explanatory variables. It is not, however, able to explain why rental income, after decades of virtual constancy, began to rise rapidly in 1994. The only variable at our disposal to explain this takeoff is the stock market value variable. Perhaps the rise in the stock market was accompanied by a parallel rise in the value of commercial real estate, which shows up in the rental income.

Rental Income, Real

Predicted ■   Actual ▫

```
ti Rental Income, Real
fdup sp500R = sp500/gdpD
f nirenR = niren/gdpD
# StockHouse defined in vfrR.reg
fex StockHouse = 100.*@cum(cvfrR,0.25*vfrR[2],.01)/ub01
r nirenR = StockHouse[8],sp500R
:                          Rental Income, Real
  SEE   =       16.71 RSQ  = 0.8561 RHO =   0.93 Obser  =  104 from 1978.100
  SEE+1 =        6.39 RBSQ = 0.8533 DW  =   0.15 DoFree =  101 to   2003.400
  MAPE  =       17.96
    Variable name          Reg-Coef  Mexval  Elas   NorRes     Mean   Beta
  0 nirenR               - - - - - - - - - - - - - - - - -    95.04 - - -
  1 intercept            -120.65316    19.0  -1.27     6.95     1.00
  2 StockHouse[8]           0.03549    31.3   2.06     1.15  5514.17  0.657
  3 sp500R                  0.03486     7.1   0.21     1.00   572.89  0.295

f niren = nirenR*gdpD
```

The **Corporate profits** modeled here are the "economic" profits of the NIPA, not the "book" profits that appear in the financial reports of the corporations. The difference lies in the two factors Inventory valuation adjustment (IVA) and Capital consumption adjustment (CCA) which eliminate from profits distortions caused by inflation. The equation is quite simple. It uses only real Gross private product and changes in its peak value. When real GDP rises by $1, profits rise permanently by $0.11, but in the same quarter with the rise in GDP, they go up by a stunning $0.60. Sixty percent of the increase goes into profits. Thus, profits are much more volatile than GDP. Now does this volatility amplify or dampen business cycles? Because profits are *subtracted* from GDP in the course of calculating Personal income, the volatility in profits actually makes Personal income more stable and contributes to overall economic stability.

28

niprfR -- Corporate Profits with IVA and CCAdj

```
title niprfR -- Corporate Profits with IVA and CCAdj
f gppR = (gdp - gdpg)/gdpD
f pgppR = @peak(pgppR,gppR,.0)
f d = pgppR - pgppR[1]
f niprfR = niprf/gdpD
sma 1000 a3 a6 1
:                niprfR -- Corporate Profits with IVA and CCAdj
  SEE   =       63.69 RSQ   = 0.8937 RHO =   0.88 Obser  =   116 from 1975.100
  SEE+1 =       30.63 RBSQ  = 0.8889 DW  =   0.24 DoFree =   110 to   2003.400
  MAPE  =        9.69
    Variable name          Reg-Coef  Mexval  Elas   NorRes     Mean    Beta
  0 niprfR                - - - - - - - - - - - - - - - - -    596.93  - - -
  1 intercept             -94.84816     7.1  -0.16    9.39      1.00
  2 gppR                    0.10484   174.5   1.09    1.17   6224.96   0.890
  3 d                       0.41867     4.1   0.03    1.09     46.39   0.089
  4 d[1]                    0.27117     4.2   0.02    1.02     45.96   0.058
  5 d[2]                    0.12474     0.8   0.01    1.00     44.41   0.025
  6 d[3]                    0.03953     0.1   0.00    1.00     44.06   0.008

id niprf = niprfR*gdpD
```

Net interest is all interest paid by business less interest received by business. It is modeled by estimating the debt of business and multiplying it by the interest rate. Business debt is taken to be its initial amount at the beginning of the estimation period, D_0, plus accumulated external financing since then, bdebt. This need for external financing is investment minus internal sources of funds — profits and capital consumption allowances less profits taxes and dividends paid (which are equal to dividends received plus dividends paid abroad minus dividends received from abroad). The external financing can be accomplished either by borrowing or by issuing equities. We will derive the net interest equation as if all of the funding was by debt; we can then recognize that part of it will be financed by issuing stock. Not all debt is refinanced ever quarter, so we smooth the Treasury bill rate, producing *srtb*. Business does not necessarily pay the Treasury rate, so we add to srtb a constant, *a*, to approximate the rate it does pay. Theoretically, then, we should have

netint $= D_0$*(a +srtb) + bdebt*(a+srtb).
$\qquad = aD_0 + D_0$*srtb + a*bdebt + bdebt*srtb

The fit obtained with this regression is acceptable, but the regression coefficients were not entirely consistent with expectations. The coefficient on *srtb*bdebt*, which should have been 1.0, came out when unconstrained a bit above 1.0 and was constrained down to 1.0. The coefficient on business debt, which should surely be less than .1 by the theory, came out at 0.30. But the main discrepancy is that the coefficient on *srtb*, which should be the initial debt — and therefore positive — is decidedly negative. Perhaps high interest rates induce firms to switch away from debt financing and towards equities.

netint -- Net Interest

- Predicted   Actual
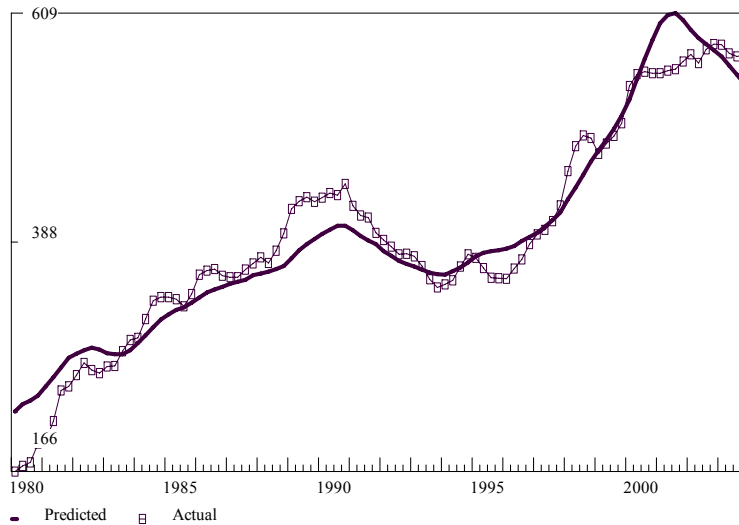
```
title netint -- Net Interest
f ub100 = @cum(ub10,1.,.1)
f srtb = 0.01*@cum(crtb,rtb[1],.1)/ub100
f bdef = v - (ncca + niprf - nictax - nicdiv)
# business deficit
fdates 1980.1 2005.4
f bdebt = @cum(bdebt,.25*bdef,0.0)
f rXbdebt = srtb*bdebt
# netint = bdebt(0)*(a +srtb) + bdebt*(a+srtb) ; and divide both sides by deflate
con 10000 1 = a4
```

:   netint -- Net Interest
  SEE   =   25.49 RSQ   = 0.9391 RHO =   0.93 Obser   =   96 from 1980.100
  SEE+1 =   10.90 RBSQ = 0.9371 DW   =   0.14 DoFree =   92 to   2003.400
  MAPE =   6.04

Variable name	Reg-Coef	Mexval	Elas	NorRes	Mean	Beta
0 netint	- - - - - -	- - - -	- - -	- - - -	391.84	- - -
1 intercept	191.30097	40.4	0.49	31.77	1.00	
2 srtb	351.83645	1.5	0.06	23.15	0.07	0.074
3 bdebt	0.12839	109.9	0.31	16.78	961.17	0.794
4 rXbdebt	1.00868	309.6	0.14	1.00	53.10	0.241

30

Dividends

The most important determinant of **dividends**, not surprisingly, is profits; and most of our equation just amounts to a long distributed lag on past profits. Because appreciation of the value of stock can also substitute, in the eye of the investor, for dividends, we have also included changes in the value of the stock market, which gets the expected negative sign.
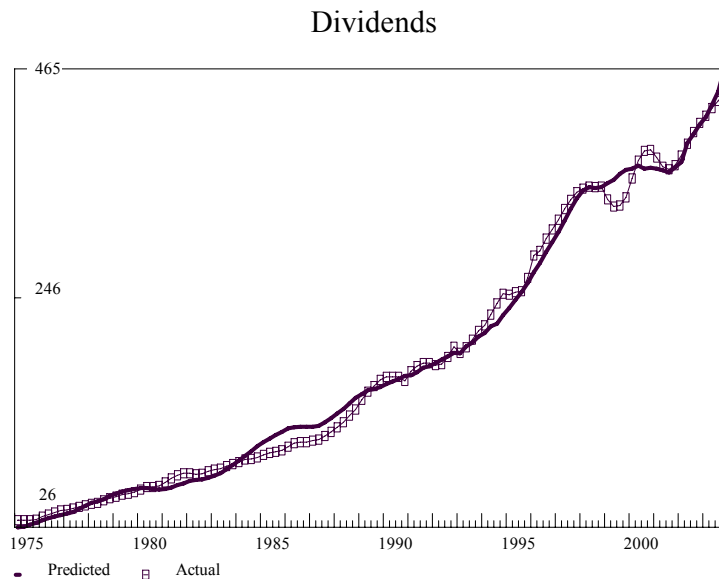
Dividends

```
title Dividends
# nicatax -- Profits after tax
f nicatax = niprf - nictax
f ub10 = @cum(ub10,1.,.10)
f sprf = @cum(cprf,nicatax,.10)/ub10
r nicdiv =  nicatax, nicatax[1], nicatax[2],
  sprf[3]
:                              Dividends
  SEE   =      9.69 RSQ  = 0.9940 RHO =   0.89 Obser  =  116 from 1975.100
  SEE+1 =      4.53 RBSQ = 0.9938 DW  =   0.22 DoFree =  111 to   2003.400
  MAPE  =      5.69
    Variable name         Reg-Coef  Mexval  Elas   NorRes      Mean    Beta
  0 nicdiv                - - - - - - - - - - - - - - -     184.47  - - -
  1 intercept            -19.48886    44.1  -0.11   166.20     1.00
  2 nicatax                0.03904     0.6   0.07     7.54   338.22   0.065
  3 nicatax[1]             0.04002     0.3   0.07     5.63   330.43   0.064
  4 nicatax[2]             0.11190     3.3   0.20     4.12   323.22   0.175
  5 sprf[3]                0.51639   103.1   0.77     1.00   273.75   0.699
```

Government budget

The basic accounting of federal government expenditures in the NIPA may be summarized in the following table. The state and local account is similar except that the grants-in-aid item, *gfegia*, is a receipt rather than an expenditure.

+ gfr   Current receipts

31

```
      + gfrt          Current tax receipts
          gfrtp          Personal tax receipts
          gfrti          Taxes on production and imports (Excises, duties, licenses)
          gfrtc          Corporate income taxes
          gfrtr          Taxes from rest of the world
      + grfcsi        Contributions for social insurance
      + gfra          Income receipts on assets
          gfraint        Interest receipts
          gfraroy        Rents and royalties
      + gfrct         Current transfer receipts
          gfrctb          From business
          gfrctp        From persons
      + gfrsurp       Current surplus of government enterprises

  - gfe   Current expenditures
      + gfece         Consumption expenditures
      + gfet          Transfer payments
          gfetsbp        Government social benefits to persons
          gfetsbr        Government social benefits to rest of the world
          gfetogia       Grants-in-aid to State and local governments
          gfetorow       Other current transfer payments to the rest of the world
      + gfeint        Interest payments
          gfeintp        To persons and business
          gfeintr        To the rest of the world
      + gfesub        Subsidies
      - gfewald       Less: Wage accruals less disbursements

  = gfsav Net Federal Government Saving

  + gfct        Capital transfer receipts (Estate and gift taxes)
  - gfctp       Captial transfer payments
  - gfv         Gross government investment
  - gfnpnpa     Net purchases of non-produced assets
  + gfconfc     Consumption of fixed capital

  = gfnet       Federal net lending (+) or borrowing (-)
```

In Quest, the **Personal tax receipts** are calculated by behavioral ratios (*gfrtpBR* and *gsrtpBR* for
federal and state-and-local cases, respectively) relative to a specially created variable called
pTaxBase defined as

> \+ Personal income
> \+ 0.5*Contributions to social insurance
> \- Government social benefits to persons

Half of Contributions to social insurance are added because in the federal most state income
taxes, one is taxed on income *inclusive* of the employee's half of the Social security tax, but
these contributions have been subtracted from Personal income in the NIPA. We have subtracted
Government transfer payments to persons on the grounds that most of these payments are either
explicitly non-taxable or go to people with low incomes and are taxed at low rates.

The **Corporate profits taxes** are calculated by behavioral ratios – essentially tax rates --
(*gfrtcoBR* and *gsrtcBR*) relative to Corporate profits. The voluntary remittances of the Federal
Reserve System to the Treasury are considered corporate taxes in the NIPA and are treated as
exogenous in real terms (gfrtcfR). **Taxes on production and imports**, in the federal case, are
mostly gasoline, diesel, alcohol and tobacco taxes and customs duties, so they are modeled by a

behavioral ratio (*gfritiBR*) relative to Personal consumption expenditure. In the state-and-local case, they also include retail sales taxes and franchise and licensing taxes. This broader base led to taking GDP as the base of the behavioral ration (*gsrtiBR*). Finally, **Contributions for social insurance** are modeled by behavioral ratios (*gfrcsiBR* and *gsrcsiBR*) relative to earned income, approximated by National income less Net interest and Corporate profits.

Turning to the expenditure side, the GDP component, Government purchases of goods and services, is specified exogenously in real terms in three parts, federal defense (*gfdR*), federal non-defense (*gfnR*) and state and local (*gsR*). In addition, we specify exogenously in real terms government investment (*gfvR* and *gsvR*). **Current consumption expenditures** are then calculated by the identities

$$gfece \quad = \quad gfd + gfn - gfv$$
$$gsece \quad = gs - gsv$$

Transfer payments, at the federal level, are divided among Unemployment insurance benefits, Transfers to foreigners, and Other. Unemployment insurance benefits are singled out for special treatment to get their automatic stabilizer effect. A behavioral ratio (*pigsbuBR*) makes them proportional to unemployment in real terms. The other two transfer payments are exogenous in real terms through the exogenous variables *gfetsbrR* and *gfetsbpoR*. The last is, of course, the huge one. **Grants-in-aid**, *gfetogiaR,* is also exogenous in real terms.

Both the federal government and the state and local governments both borrow and lend money. Consequently, they have both interest payments and receipts. The difference between the two levels of government, however, is profound; and the approach which works well for the federal government does not work at all for the state and local governments. For the **Net interest paid by the federal government**, which is a huge net borrower, we can calculate the overall deficit or surplus in each quarter and cumulate this amount to obtain a rough estimate of the net amount on which the government is earning or paying interest. By use of G's *fdates* command, we make the cumulation of the deficit or surplus begin at the same time that the regression begins. (The *fdates* command controls the dates over which the *f* commands work.) Because not all debt is refinanced instantly with the change in the interest rate, we use an exponentially weighted moved average of the rates, *frtb* or *srtb*, to multiply by the debt. We should then have

$$gfenip = InitialDebt*frtb + fcumdef*frtb$$

where *fcumdef* is the cumulated deficit of the federal government. The InitialDebt thus becomes a parameter in the regression equation. Notice that there is no constant term in this equation. We have therefore forced G to omit the constant term by placing a ! after the = sign in the *r* command. We have also included *rtb* as a separate variable in addition to *frtb* so that the regression can take an average of them to produce the best fit.

The same approach will not work at all for the **Net interest paid by state and local governments**, largely because these governments can borrow at low rates because the interest

they pay is exempt from federal income tax. Thus, the rate they pay on their debt is far below the rate they receive on their assets, so the net indebtedness is not sufficient to make even a rough guess of the interest payments. Indeed, over the last twenty years the net indebtedness has grown while the net interest paid has become more and more negative. (The increase in the indebtedness is not immediately apparent from the NIPA, which show a positive surplus, *gssurp* in our bank.. The problem is that this surplus is not reckoned with total purchases of goods and services, *gs*, but only with consumption expenditures, *gsece*. The difference is that *gs* includes capital outlays while *gsece* excludes capital outlays but includes imputed capital conscumulated surplus relevant for our purposes would be calculated with total expenditures, *gs*, and that surplus is negative throughout most of the last twenty years.)

In this situation, we have had recourse to a simpler device and assumed that state and local governments have tried to maintain both financial assets and liabilities roughly proportional to total purchases of goods and services, *gs*. Under that assumption, net interest payments should depend on *gs* and on its product with the interest rate. The fit is satisfactory and the elasticity of interest receipts with respect to *gs* just a little above 1.

gfenip -- Net Interest Paid by the Federal Government

Predicted ▪   Actual ▫
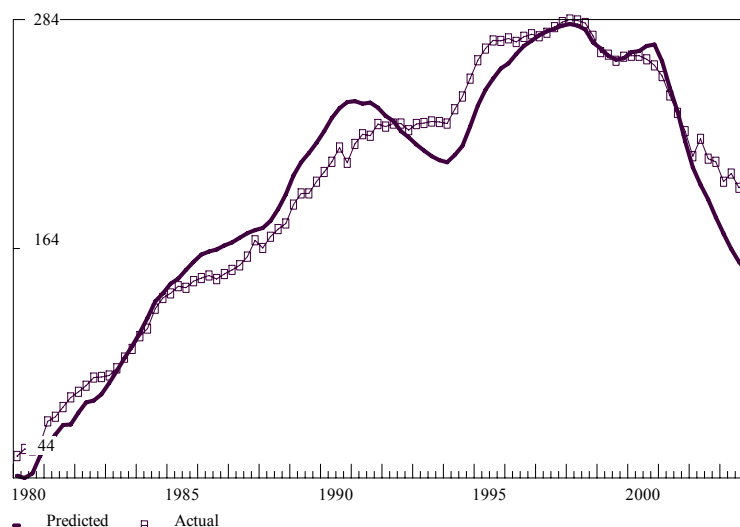
```
title gfenip -- Net Interest Paid by the Federal Government

f ub100 = @cum(ub100,1.,.1)
f frtb = @cum(cfrtb,.01*rtb,.1)/ub100
# Calculate federal government deficit
fdates 1979.4 2010.4
f fcumdef = @cum(fcumdef,-.25*gfnet,0.0)
fdates 1960.1 2010.4
f frXfcumdef = frtb*fcumdef[1]
f rXfcumdef = rtb*fcumdef[1]
f gfenip = gfeint - gfraint
con 300 1 = a3
r gfenip =  ! frtb, rtb, frXfcumdef, rXfcumdef
```

```
:             gfenip -- Net Interest Paid by the Federal Government
  SEE    =      15.20 RSQ   = 0.9476 RHO =    0.96 Obser   =    96 from 1980.100
  SEE+1  =       4.52 RBSQ  = 0.9459 DW  =    0.08 DoFree =    92 to   2003.400
  MAPE   =       6.98
   Variable name              Reg-Coef  Mexval  Elas   NorRes     Mean    Beta
  0 gfenip                    - - - - - - - - - - - - - - -     195.58 - - -
  1 frtb                     407.31672     4.7   0.14    48.94     0.07
  2 rtb                        0.79530     0.2   0.03    44.49     6.19   0.037
  3 frXfcumdef                 1.48250   147.2   0.79     1.01   104.30   1.090
  4 rXfcumdef                  0.00076     0.7   0.04     1.00  9056.66   0.062
```

Net Interest Paid by State and Local Governments

Net Interest Paid by State and Local Governments

Predicted   Actual

```
title Net Interest Paid by State and Local Governments

f gsXrtb = gs*rtb
f gsenip = gseint - gsraint
r gsenip =   gs, gsXrtb
:            Net Interest Paid by State and Local Governments
  SEE    =       3.48 RSQ   = 0.5499 RHO =    0.95 Obser   =    96 from 1980.100
  SEE+1  =       1.16 RBSQ  = 0.5402 DW  =    0.09 DoFree =    93 to   2003.400
  MAPE   = 242077.88
   Variable name              Reg-Coef  Mexval  Elas   NorRes     Mean    Beta
  0 gsenip                    - - - - - - - - - - - - - - -      -2.57 - - -
  1 intercept                 -4.83002     5.2   1.88     2.22     1.00
  2 gs                         0.01110    37.6  -3.22     1.29   744.78   0.634
  3 gsXrtb                    -0.00155    13.7   2.34     1.00  3874.16  -0.363
```

Subsidies less current surplus of government enterprises are small and have been taken
exogenously in real terms for all levels of government. **Wage accruals less disbursements** are
generally zero and have been left exogenous in nominal terms.

With these items, we are able to calculate the **Net federal government saving** and **Net state
and local government saving** on the NIPA basis. To calculate **Net lending (+) or borrowing (-
)**, however, we need a few more items. The most important of these is **consumption of fixed
capital.**

Until fairly recently, all government purchases were considered current expenditures in the NIPA. Thus, the construction of a road entered into the GDP only in the year it was built; services from the road were not counted as part of the GDP. In the private sector, however, the consumption of fixed capital, depreciation expense, enters into the price of goods consumed. Thus, a capital expenditure in the private sector is counted in GDP twice, once as fixed investment in the year in which it is made and then again in the prices of goods and services as it is consumed in future years. (In Net Domestic Product, this second appearance has been removed.) To give government capital formation similar treatment, the NIPA have recently begun to distinguish between current expenditures and capital expenditures. The capital expenditures are then amortized to create a consumption of fixed capital expense. Our technique for estimating this consumption given previous investment is similar to what we used in the private sector. Here are the equations for the two level of governments.

FederalConsumption of Fixed Capital

Predicted   Actual
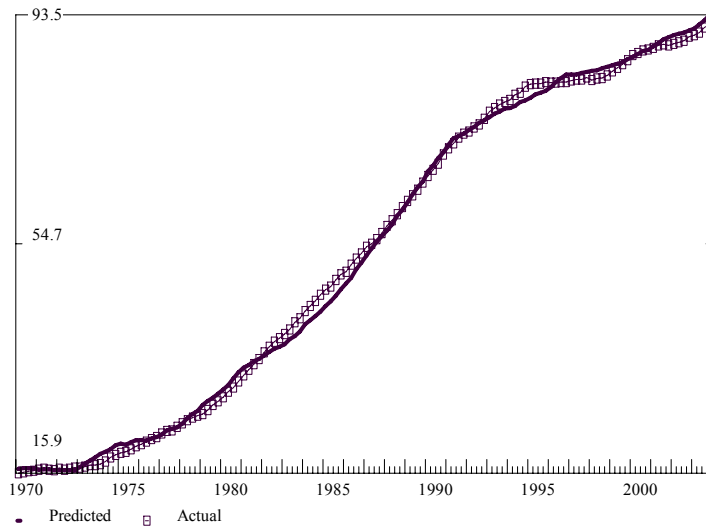
```
ti FederalConsumption of Fixed Capital
fex gfvR = gfv/gdpD
f gfv = gfvR*gdpD
f ub02 = @cum(ub02,1.,.02)
f gfvrep = gdpD*@cum(gfvstk,gfvR,.02)/ub02
```

```
:                     FederalConsumption of Fixed Capital
  SEE   =        1.22 RSQ   = 0.9979 RHO =    0.96 Obser   =  136 from 1970.100
  SEE+1 =        0.34 RBSQ  = 0.9979 DW  =    0.08 DoFree =  134 to    2003.400
  MAPE  =        2.38
    Variable name            Reg-Coef  Mexval  Elas   NorRes     Mean    Beta
  0 gfconfc                  - - - - - - - - - - - - - - - -    53.17 - - -
  1 intercept                -2.70111    38.3  -0.05   481.38     1.00
  2 gfvrep                    1.03406  2094.0   1.05     1.00    54.03  0.999
```

36

State and Local Consumption of Fixed Capital

Predicted   Actual
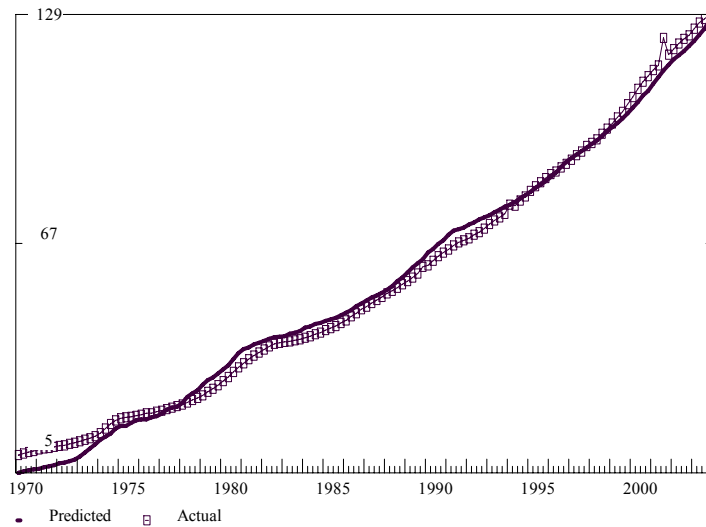
```
ti State and Local Consumption of Fixed Capital
fex gsvR = gsv/gdpD
f gsv = gsvR*gdpD
f ub02 = @cum(ub02,1.,.02)
f gsvrep = gdpD*@cum(gsvstk,gsvR,.02)/ub02

:                 State and Local Consumption of Fixed Capital
  SEE   =       1.41 RSQ   = 0.9983 RHO =   0.83 Obser  =  136 from 1970.100
  SEE+1 =       0.81 RBSQ  = 0.9983 DW  =   0.35 DoFree =  134 to   2003.400
  MAPE  =       3.06
    Variable name         Reg-Coef  Mexval  Elas   NorRes     Mean   Beta
  0 gsconfc               - - - - - - - - - - - - - - - - -   56.48 - - -
  1 intercept            -10.24848   248.4  -0.18   599.23     1.00
  2 gsvrep                 0.75899  2347.9   1.18     1.00    87.92  0.999
```

The spill rates were chosen after some experimentation to get a good fit. The replacement calculated for the federal government is fairly close to the NIPA capital consumption series; for state and local government, however, the calculated replacement is much above that used in the NIPA.

As a result of recent changes in the NIPA, **Estate and gift taxes** are no longer counted as government revenues but appear, more correctly, as **Capital transfers.** For the Federal case, they have been made exogenous in real terms (*gfctpR*) while for state and local governments, they are proportional (*gsrtrpBR*) to disposable income.

The final item in the government accounts is the **Net purchases of non-produced assets** such as land or existing buildings. These purchases cannot go into GDP, precisely because the land or the stock is not produced. On the other hand, they enter the cash "bottom line" of the governments. The federal item is taken as a behavioral ratio, *gfnpnaBR,* to total federal purchases of goods and services. The state and local is exogenous in real terms, *gsnpnaR*.

37

From these variables, the complete government accounts as set out at the beginning of this section can be computed.

The Stock Market Value

The **real stock market value** variable, *sp500R* – the Standard and Poor 500 index deflated by the GDP deflator – has been used in a number of equations. Now we turn to trying to explain the variable with other variables in the model. Fundamentally, the value of a stock should be present value of the stream of future profits discounted by the rate of interest. If we put the profits in real terms, then the interest rate used should be a real rate. Basically, our equation for *sp500R* relates it to the present value of future profits by presuming that both profits and interest rates are expected to remain at their present level in real terms. Both profits and interest rates have been exponentially smoothed to reduce variability that was not reflected in the stock market series. Profits are likely to be discounted at rates considerable above the Treasury bill rate. After trying several values, we settled on adding 5 percentage points to the "perceived" Treasury bill rate. The regression coefficient on this variable was then constrained to give it an elasticity of 1. A time trend was also allowed.

The results below show this equation estimated only through 1994.4, roughly the beginning of the present bull market. Notice that the 1987 "correction" brought the market back close to the value calculated by this equation. The lines to the right of the vertical line compare the actual values of the stock market variable with the values which would be "justified" by the equation estimated over the previous fifteen years. The time trend fortunately turns out to be small, a quarter of a percent per year of the mean value of the index.

Real S&P 500 Index
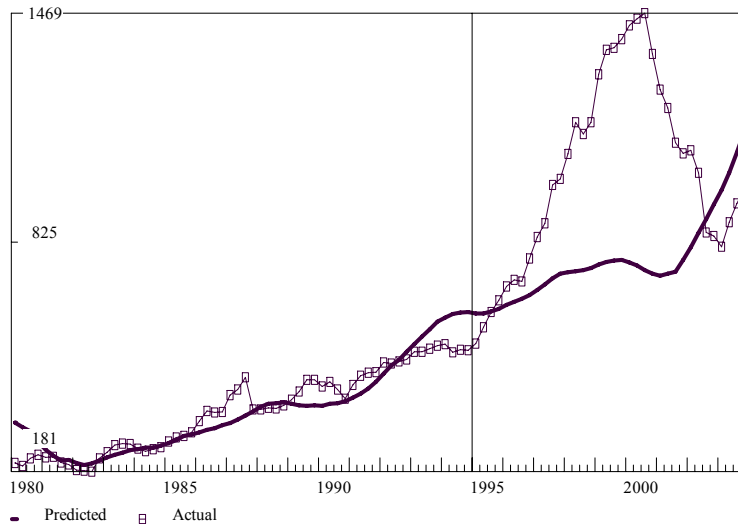
ti Real S&P 500 Index

```
f ub10 = @cum(ub10,1.,,.1)
f rtbexs = @cum(crtbex,5.+rtbex,.10)/ub10
f niprfs = @cum(cniprf,niprf,.10)/ub10
fex sp500R = sp500/gdpD
f DiscProfit = (niprfs/rtbexs)/gdpD
# constrain to give Discounted Profits an elasticity of 1.
con 1000 7 = a2
r sp500R = DiscProfit,time
:                              Real S&P 500 Index
  SEE  =       49.22 RSQ  = 0.8115 RHO =    0.90 Obser  =   60 from 1980.100
  SEE+1 =      25.56 RBSQ = 0.8049 DW  =    0.19 DoFree =   57 to   1994.400
  MAPE  =       10.52 Test period:  SEE   370.41 MAPE    25.43 end  2003.400
    Variable name        Reg-Coef  Mexval  Elas   NorRes    Mean   Beta
  0 sp500R              - - - - - - - - - - - - - - - -    353.90 - - -
  1 intercept          -231.48904    22.7  -0.65   24.47     1.00
  2 DiscProfit            6.67555   343.6   1.08    1.36    57.07  0.825
  3 time                  7.39873    16.6   0.58    1.00    27.62  0.283

id sp500 = sp500R*gdpD
```

The Exogenous Variables

To facilitate the use of the model, here is a list of all the exogenous variables in one place.

lfc	Civilian labor force
pop	Population
gm1	Growth rate of M1
fgndem	Foreign demand, used in export equation
relpri	Prices of imports relative to prices of exports, used in inflation equation
fefaci	Exports of factor income
fifaci	Imports of factor income
taxacts	Dummy for tax acts affecting construction
d80	Dummy in the export equation
disaster	Dummy for hurricane and earthquake in capital consumption
nbtrpBR	Behavioral ratio for business transfer payments
nbtrppBR	Behavioral ratio for business transfer payments to persons
nsd	Statistical discrepancy
nwald	Wage accruals less disbursements

In the government sector, there are usually parallel variables for federal (in the first column below) and state-and-local governments (in the second column). All variables ending in *R* are in constant prices. Those ending in BR are ratios to some other variable as explained in the government section above.

Federal	S&L	Description
gfdR		Purchases of goods and services for defense
gfnR	gsR	Purchases of goods and services, non-defense
gfvR	gsvR	Capital investment
pituibBR		Unemployment insurance benefit rate

gfetpfR		Transfer payments to foreigners
ogfetpR	gsetpR	Other transfer payments
gfeifBR		Interest payments to foreigners
pitfBR	pitsBR	Personal tax rates
gfribtBR	gsribtBR	Indirect business tax rate
gfrprfBR	gsrprfBR	Profit tax rates
gfrcsiBR	gsrcsiBR	Social security tax rates
gfctrBR	gsctrBR	Estate and gift tax rates
gfpnaBR	gspnaBR	Ratio for purchases of non-produced assets
gfeslsR	gseslsR	Subsidies less surplus of government enterprises
gfetpfR		Transfer payments to foreigners
gfegiaR		Federal grants in aid to state and local government
gfeald	gseald	Wage accruals less disbursements

3. Historical Simulations

In the following graphs, the heavy line with no marking (blue, if you are reading this in color) is the actual historical course of the variable. The (red) line marked with +'s is the simulation with the stock market variable at its historical values, while the (green) line marked with x's is the simulation using the equation for the stock market equation.

u -- Unemployment
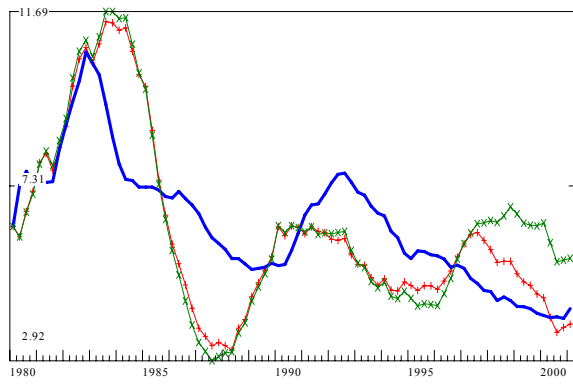
gdpR -- Real Gross Domestic Product
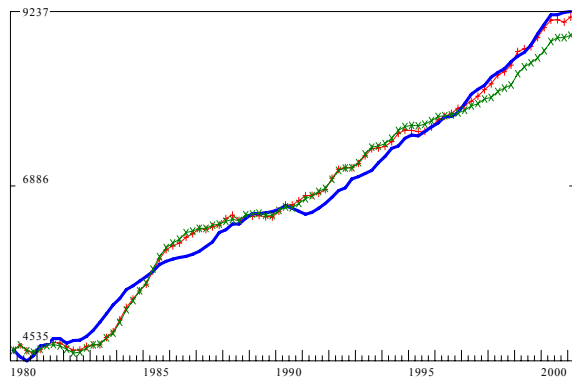
cR -- Personal Consumption Expenditure
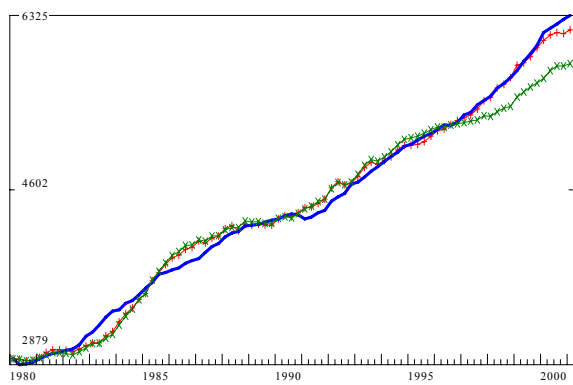
vfnreR -- Equipment Investment
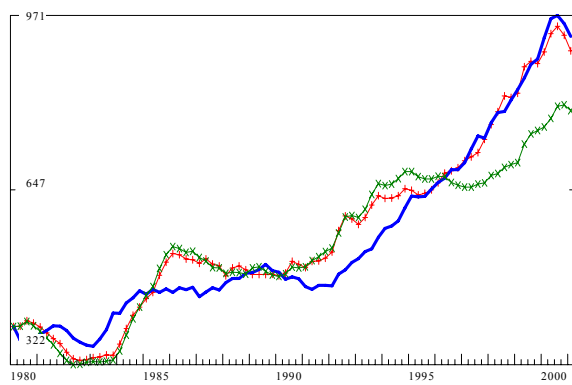
vfnrsR -- Non-residential Structures

vfrR -- Residential Construction

41
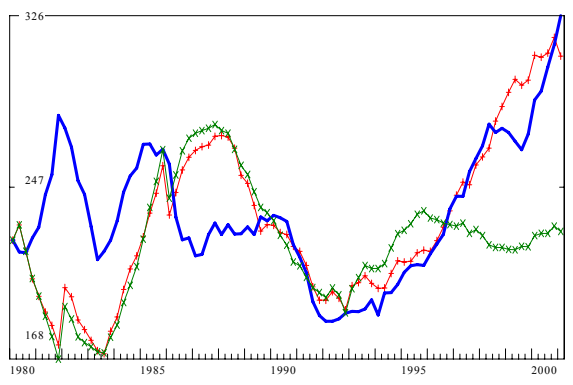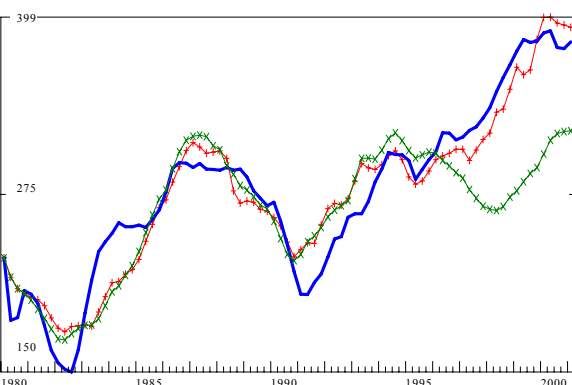
feR -- Exports
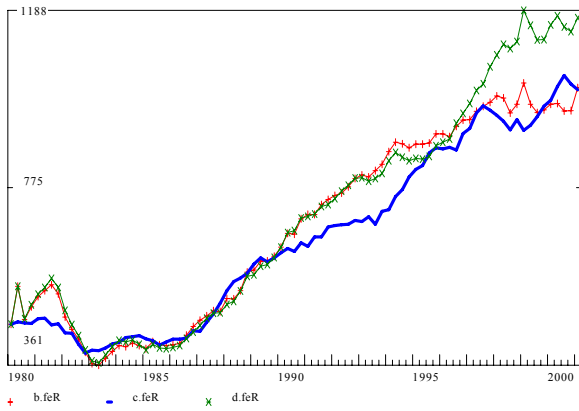
fiR -- Imports
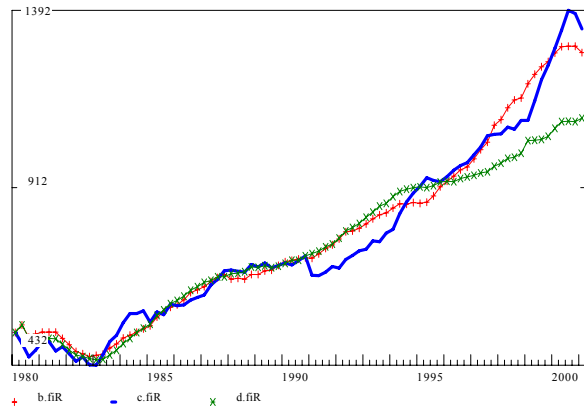
gdpD -- GDP Deflator
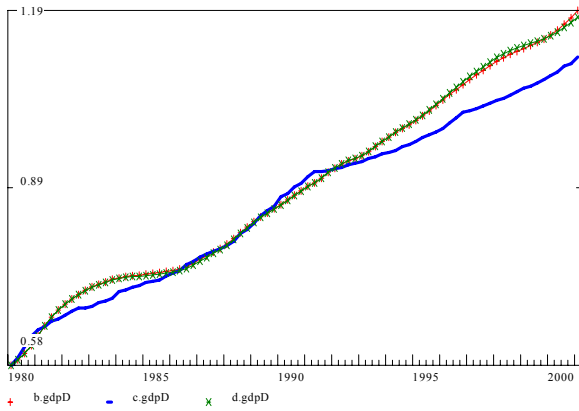
infl -- Inflation
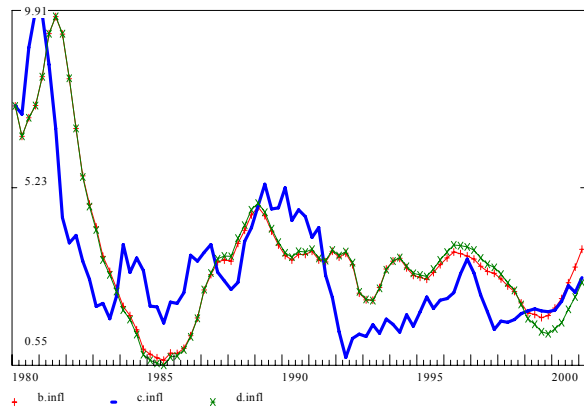
pisav -- Personal Saving
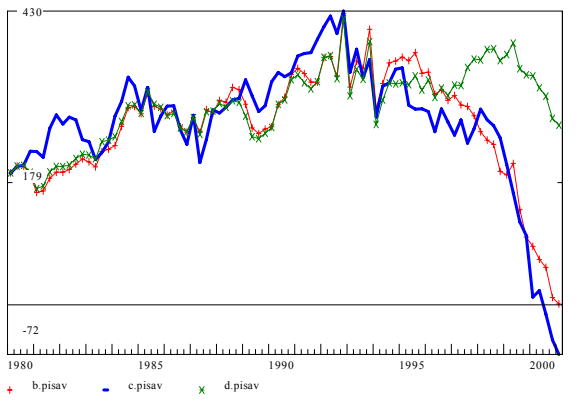
sp500R -- S&P 500 Stock Market Index, Real

42
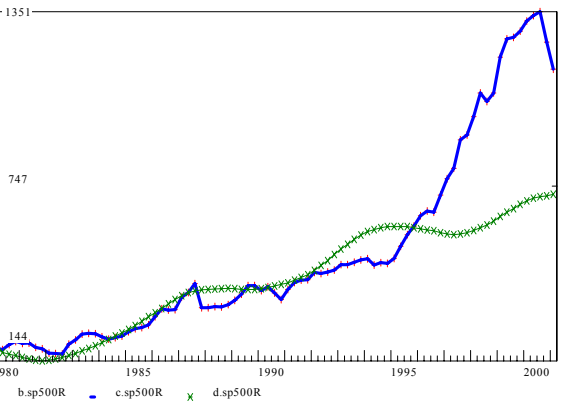
rtb -- Rate on New 90-day Treasury Bills

v1 -- M1 Velocity

gfenet -- Federal Lending (or Borrowing)

gsenet State and Local Lending

43
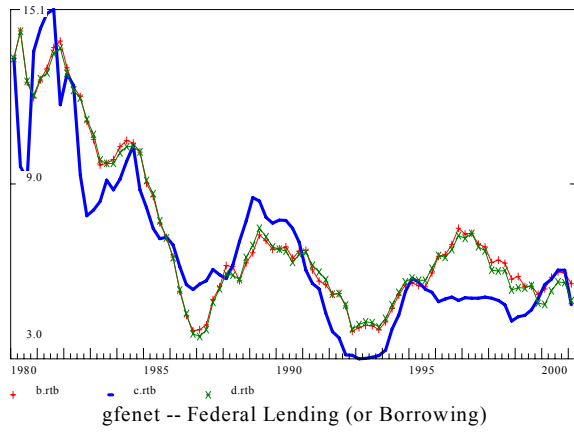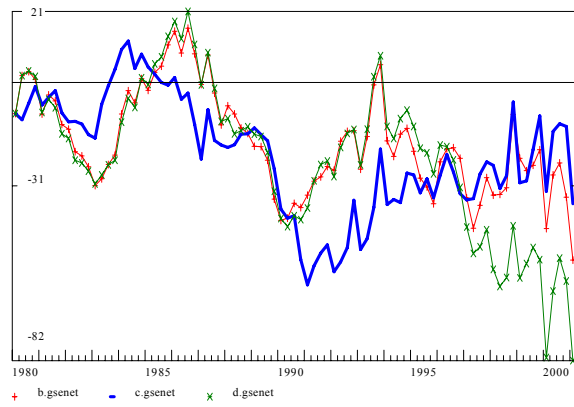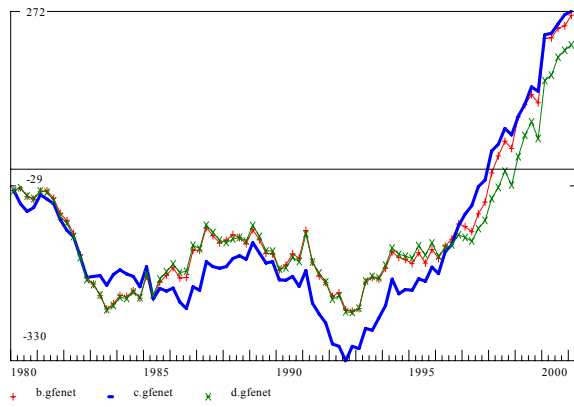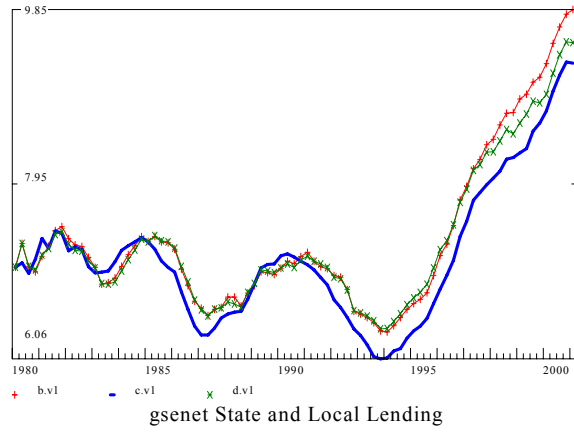
Up through 1992, the stock market equation worked well and there is essentially no difference between the two simulations. Both track the real variables, such as GDP, Personal consumption expenditure, residential construction, equipment investment, and employment fairly well with the exception that the model produces a stronger boom in 1986 and 1987 than actually happened. After 1996, the story is quite different. The simulation with the stock market taking its normal course as an endogenous variable shows, to be sure, a steady, moderate growth in the stock market but a significant recession in 1996-1997 followed by a weak recovery with a rising unemployment rate that almost reached the levels of 1981-1982 in 1999 before a slight recovery in 2000. In sharp contrast, the simulation with the stock market variable set exogenously to its actual, historical values gave fairly close simulations of the real variables up through 2000. In particular, the personal savings rate falls and Personal consumption expenditures rise in this simulation very much as they actually did historically.

The story is a little different for the price level. The simulations track it quite well up to about 1990; thereafter it gets above the historical values and stays there to the end of the period. In other words, the inflation rate misses on the high side for a year or so and then remains very close to the actual. In theory, tight money (indicated by a high monetary velocity) should have reigned in the economy by reducing investment and consumption. The M1 velocity graph, however, shows that the differences of the simulation from the historical velocity were small in comparison with the changes which were taking place historically in the velocity. It was therefore difficult to find a measure of monetary tightness which would show up as statistically useful in estimating the equations.

The conclusions I draw from these results are:
> The stock market is quite important to the economy.
> Given the stock market behavior, the model can predict the rest of the economy,
>   especially its real variables, fairly well.
> The boom in the stock market which began in 1995 is responsible for the strong economy
>   of the period 1996 - 2000.
> This the causes of this boom in the market lay outside the U. S. economy.

These external causes are not hard to find. Beginning in 1996, weakness in Asian and other economies led to an influx of foreign investment into the U.S. stock market. Without the externally driven rise in the stock market, the years 1996 - 2000 would have shown weak but positive growth. The exceptional prosperity of the period was the result of the bull market superimposed on a fundamentally stable but not especially dynamic economy.

4. Alternative Forecasts

To study the effect of the stock market on the cyclical evolution of the American economy in the coming years, we have formulated four alternative projections. They differ only in the projection of the real value of the stock market index, *sp500R*. All four alternative projections

are made by adding a factor to the endogenous equation for *sp500R*. In naming the alternatives, we expand on the custom of distinguishing between "bulls" and "bears". The alternatives are:

Name	Mark	Description
Bull	+ plus	The add factor reaches a minimum in 2001.3, climbs back to its highest historical value by 2002.4, and continues to grow at the same rate at which it grew from 1996 to the peak in 2000. (Red, if you are reading on a screen.)
Sheep	□ square	The add factor stays where it is likely to be in 2001.3. (Blue)
Bear	▼ triangle	The add factor is generated automatically by the rho-adjustment process. (Purple)
Wolf	◆ diamond	The add factor, which was 400 in 2001.1 hits 100 by 2001.4 and drops on down to -100 by 2001.4, where it stays for a year before moving up to -10 by the end of 2005. (Black)

All of the alternatives reflect the Bush tax cut of 2001 and otherwise use middle-of-the-road projections of exogenous variables. Here are the comparison graphs.

45

gdp -- Gross Domestic Product

b.gdp   c.gdp   d.gdp   e.gdp

cR -- Personal Consumption Expenditure

b.cR   c.cR   d.cR   e.cR

vfnreR -- Equipment Investment

b.vfnreR   c.vfnreR   d.vfnreR   e.vfnreR

vfnrsR -- Non-residential Structures

b.vfnrsR   c.vfnrsR   d.vfnrsR   e.vfnrsR

vfrR -- Residential Construction

b.vfrR   c.vfrR   d.vfrR   e.vfrR

feR -- Exports

b.feR   c.feR   d.feR   e.feR

46

fiR -- Imports

b.fiR   c.fiR   d.fiR   e.fiR

u -- Unemployment

b.u   c.u   d.u   e.u

rtb -- Rate on New 90-day Treasury Bills

b.rtb   c.rtb   d.rtb   e.rtb

infl -- Inflation

b.infl   c.infl   d.infl   e.infl

v1 -- M1 Velocity

b.v1   c.v1   d.v1   e.v1

pisav -- Personal Saving

b.pisav   c.pisav   d.pisav   e.pisav

47

rtb -- Rate on New 90-day Treasury Bills

b.rtb   c.rtb   d.rtb   e.rtb

gfenet -- Federal Lending (or Borrowing)

b.gfenet   c.gfenet   d.gfenet   d.gfenet

gsenet State and Local Lending

b.gsenet   c.gsenet   d.gsenet   e.gsenet

pgppR -- Peak Real Gross Private Product

b.gdpR   c.gdpR   d.gdpR   e.gdpR

All the alternatives agree that we are in for a considerable recession beginning in the last quarter of 2001. For comparison, it is useful to remember that the recession beginning in 1990 lasted three quarters and saw a drop of 1.7 percent in real GDP. The one just ahead should also last three quarters (or four for the Wolf scenario) but the drop in real GDP may be on the order of 2 or 3 percent. Looking over the graphs above show much greater drops in consumption and investment. Exports and imports, however, act as very strong stabilizers, and -- in this model -- respond very quickly to changes in the stock market. The response is so fast that Bull, which activates the export-import stabilizers least of the four, turns out to have the sharpest and deepest recession, at 3 percent in three quarters. Wolf, which activates them most, has a 2.9 percent drop over four quarters, while Sheep loses 2.6 percent over three quarters and Bear drops only 2.0 percent over three quarters. The details of this short-run response can be seen clearly in the following graph.

Once the recovery is underway, the alternatives assume the expected order according to speed of Bull, Sheep, Bear, and Wolf. The maximum difference between Wolf and Bull is 4.2 percent.

The combination of the tax cuts and the recession will wipe out half of the Federal budget surplus. The model does not distinguish between the social insurance trust funds and the general budget, but it is clear that the general budget will be in deficit during the recession. The State

and Local deficit is sharply increased by the recession, and one can expect cut backs in expenditures to avoid these deficits.

After the recession, unemployment stabilizes at about 5.5 percent and inflation at about 2.5 percent. Personal savings, after rising during the recession, shows a disturbing tendency to diminish.

All in all, it appears that the model is capable not only of generating a substantial cycle but also, when the exogenous variable are stable, of producing stable growth at plausible levels of unemployment and inflation.

Chapter 9. Optimization in Models

Up to this point, we have estimated equations in isolation and then combined them into a model and observed how the model worked. Occasionally, we have revised the estimate of some regression coefficient to improve the functioning of the model. In this chapter, we will see how to modify coefficients in a comprehensive way to improve the performance of the model in historical simulation. The same techniques, with a different objective function and different parameters, can then be used to design policies. Let us begin, however, with improving the performance of the model in historical simulation.

1. Improving the historical simulation

Creating an Objective Function

The first step in optimizing must be to create an objective function. This objective function must be built into our model. Our software uses the convention that it **minimizes** the value in the last period of the simulation of some specified variable. (How we tell the program which variable it should minimize will be shown in the next section.). For example, to optimize the performance of the Quest model in historical simulation, we would probably initially want to concentrate on real GDP (*gdpR*) and the GDP deflator (*gdpD*). Let us say that we want to minimize the sum of the squares of their relative, fractional differences from their historical values. We then need to record the historical values in variables which will not be changed in the model, so we create two exogenous variables, *gdpRX* and *gdpDX* for that purpose by the equations:

    fex gdpRX = gdpR
    fex gdpDX = gdpD

The relative difference between the model's real GDP in any period and the historical value for that period would be (gdpR-gdpRX)/gdpRX and for the GDP deflator it would be (gdpD-gdpDX)/gdpDX. The contribution to the objective function from these discrepancies in any one period would be

    f miss = @sq((gdpR-gdpRX)/gdpRX)+@sq((gdpD-gdpDX)/gdpDX)

where *@sq( )* is the squaring function. Finally, the objective function itself, the sum over all periods of these period-by-period contributions, would be the value in the last period of the simulation of the variable *misses* defined by

    f misses = @cum(misses, miss, 0.)

These statements can be conveniently placed at the end of the Master file of the model just before the "check" commands.

Selecting parameters to vary

With the objective function in place, the next step is to select from all the regression coefficients s in the model those which will be varied in looking for an optimum. One might ask, "Why not vary all of them?" Our objective function, however, is quite a complicated function of all these coefficients, so the only feasible optimization techniques are those that involve some sort of trial-and-error search with the whole model being run to evaluate the objective function for each proposed point, that is, for each set of regression coefficient values. The number of points that has to be searched increases with the dimension of the point. We will see, however, that optimizing with respect to a relatively small number of coefficients – a dozen or so – can produce a substantial improvement in the Quest model.

The optimization method we will use is known as the simplex method . A *simplex* in n-dimensional space is a set of n+1 points in that space. For example, a triangle is a simplex in 2-dimensional space and a tetrahedron is a simplex in 3-dimensional space. The method requires that we specify an initial simplex of points; it will then take over, generate a new point, and, if that point is better than the old worst point in the simplex, drop the worst point and add the new point to the simplex. It has four different ways of generating new points. First it *reflects* the worst point through the midpoint of the other points. If that works, it tries to *expand* by taking another step of the same size in the same direction. If the expansion gives a better point than did the reflection, that point is added to the simplex and the worst point is dropped. If the reflection gave a point better than the worst point but the expansion did not improve on it, the reflected point is added to the simplex and the worst point dropped. If the reflection failed to give a point better than the worst point, the algorithm *contracts*, that is, it tries a point halfway between the worst point and the midpoint of the other points. If this point is better than the worst point, it is added to the simplex and that worst point dropped. Finally, if all of these trials have failed to yield a point better than the worst point, the algorithm *shrinks* the simplex towards the best point by moving all the other points halfway towards it. When the value of the objective function is practically the same at all the points and the points are close together, it stops.

Our task is to supply the initial simplex. One obvious point for inclusion is the values of the coefficients estimated by the original regressions. We specify the other points by varying each coefficient, one-by-one, from this base. For each coefficient, we will specify a "step size" for this variation. The initial points of the simplex are then the original values of the parameters that may be varied and then, for each parameter, a point with that parameter increased by its "step size" and all the other parameters at their original values. Note that with n parameters, this method will give n+1 points, a simplex in n-dimensional space.

Mechanically, how do we specify the parameters to be varied and their step sizes? An example for Quest will be helpful. We will optimize on parameters from the consumption function, that is, the equation for *cRpc,* and the most important of the investment equations, that for *vfnreR*. For ease of reference, here are excepts from the regression results of the consumption equation.

```
:                       Personal Consumption per capita
    Variable name           Reg-Coef  Mexval  Elas   NorRes     Mean    Beta
  0 cRpc                  - - - - - - - - - - - - - - - - 17419.94 - - -
  1 intercept              785.42866     1.1   0.05   792.49     1.00
  2 yRpc                     0.77579    31.8   0.86     9.50 19284.28   0.711
  3 dyRpc                   -0.39068     8.5  -0.00     9.25   104.29  -0.018
 ....
 13 piipcbRpc               -0.24243     0.1  -0.01     1.29   507.34  -0.008
 14 intsavRpc               -0.48752    10.1  -0.03     1.09  1151.80  -0.044
 15 rtbexXdi                -0.00161     1.5  -0.01     1.03 55178.74  -0.015
 16 ur                    -1417.2942     1.3  -0.01     1.00     0.17  -0.020
```

Examination of the historical simulations shown in the previous chapter shows that the equipment investment equation is a major generator of the boom in the mid 1980's that was much stronger in the historical simulation than in reality. Could inclusion of an unemployment variable in this equation help stabilize the model? One could argue that, in times of tight employment, capacity constraints may result in orders for capital goods may not be filled promptly so that actual investment may be less than would be desired on the basis of other factors. The number of persons unemployed, *ue*, was put in with the following results:

```
f ue = lfc -emp
:                       Equipment Investment
    Variable name           Reg-Coef  Mexval  Elas   NorRes     Mean    Beta
  0 vfnreR                - - - - - - - - - - - - - - - - 510.68  - - -
  1 intercept               36.82448     1.7   0.07   120.92     1.00
  2 repEq                    0.97140   465.9   0.77    10.00   404.92   0.629
  .....
 24 dsp500R[7]               0.02980     0.1   0.00     1.05     8.60   0.005
 25 ue                      -4.60458     2.4  -0.07     1.00     7.64  -0.036
```

The unemployment variable got a negative coefficient, which would only make the cycles worse. No doubt we have here a case of simultaneous equation bias, for booming investment will drive down unemployment. Rather than try instrumental variables or other simultaneous equations techniques, let us just make this coefficient one of the variables on which we optimize.

The specification of which parameters to use in optimization and their step sizes is now provided by the following file, which we may call Fit.opt.

```
misses
20
vfnreR
#  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25
   .1 0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0  0 .1
cRpc
#  1     2 3  4  5  6  7  8  9 10 11 12 13    14 15 16
   .1 .001 0  0  0  0  0  0  0  0  0  0  0  .005  0  1
```

The first line of the file gives the name of variable whose last value is to be minimized. The second line specifies the maximum number of parameters which will be varied in the course of the optimization. It does not hurt if it is larger than the number actually used. Here we have set the maximum at 20 but will only use 6.

The next line says that some parameters will come from the equation for *vfnreR*. The third line begins with a # which marks it as simply a comment ignored by the program. For us, however, it is very useful since it numbers the 25 regression coefficients which occur in the equation for *vfnreR*. The line below it gives the step sizes for each of these 25 coefficients. A coefficient given a step size of 0 is not involved in the optimization. Thus we see that coefficient 1, the intercept, is given a step size of .1 and that the coefficient of *ue* is also given a step size of .1.

The next triplet of lines does the same for three coefficients in the *cRpc* equation, the intercept, the coefficient of the inflationary interest that "should" be saved, and the reciprocal of the unemployment rate.

Note that in both equations, the intercept is included among the variables on which we optimize. The reason is that, unless a variable happens to have a mean of zero, changing the coefficient on it will require a change in some other variable's coefficient to keep the sum of errors in the equation zero. The intercept is a natural choice for this other variable since it seldom has an economic significance which we want to preserve.

With this file created, we are ready to optimize our objective function.

Optimizing

When the model with the objective function has been built (by clicking Model | Build in G), we can run it in optimizing mode. Click Model | Run and then in the top right corner of the screen in the panel labeled "Type of Simulation" click the radio button for "Optimizing". Fill in the dates of the simulation and the "fix" file as usual. Specify the name of the bank which will contain the optimized model run. I usually call it "Optima", but any word of 8 or less letters and numbers will do. Finally, in the window labeled "Optimization file name", give the name of the file created in the previous step. In our case, it is OptSpec.opt, which is what the program puts in that window by default. The root-name of this file (the part before the .opt) will be used to label several of the files resulting from the optimization. Then click OK. You will then get a black DOS screen with the usual] prompt. You can provide a title for the run with a "ti" command or supplement the "fix" file. When running Quest over history, I often give the "skip sp500R" here to use historical values of the S&P 500 index. When you have no further fixes to add, give the command "run" as usual.

When optimizing, the model does not print dates and the values of variables being checked. Instead, it reports for each move of the simplex whether the action was to reflect, expand, contract, or shrink. It also shows the value of the objective function at the best and worst points of the simplex.

The implementation of the simplex method used by our program is borrowed from section 10.4 of *Numerical Recipes in C* by William H. Press *et al.* (Cambridge, 1988; the code and text is available on the Internet at www.nr.com .) This code seems prone to reach local minima. Therefore, when an optimum is reported by the borrowed code, our routine takes it as a starting

point and then uses the step sizes to vary it. If one of the new points is better than the supposed optimum, the algorithm is started again, with the message "Starting or restarting optimization" printed on the screen.

When no further improvement appears possible, you will get a list of the parameters with their starting values and their optimized values. This information will also be written into the file Changes.chg. When you then tap any key the model will be run with the optimized parameters and the results stored in the bank you indicated on the Run model screen.

When Quest was optimized with the objective function given above with respect to the parameters specified by the OptSpec.opt file shown above, the coefficients were changed as follows:

```
Resulting coeficients after maximization (183 runs).
Variable      Old:          New:
vfnreR
  intercept   36.8245        36.2423
  ue          -4.6046        -0.6041
cRpc
  intercept   785.4286       804.7291
  yRpc           0.7758        0.7669
  intsavRpc     -0.4875       -0.4898
  ur          -1416.2942  -1464.8260
```

One might suppose that these changes are so small that the optimization must have made little difference in the objective function. That impression, however, is quite misleading as shown in the graphs below. In them, the heavy (blue) line with no marking of points is the actual, historical line. (In the first two graphs, it lies along the horizontal axis, for of course the historical data fits itself perfectly.)   The (red) line marked with + is generated by the model before optimization; the (green) line marked with x is from the optimized model. Remember that we are trying to minimize errors, so lower is better.

objective -- Objective Function

obj -- Increment to Objective Function

gdpD -- GDP Deflator

gdpR -- Real Gross Domestic Product

cR -- Personal Consumption Expenditure

vfnreR -- Equipment Investment
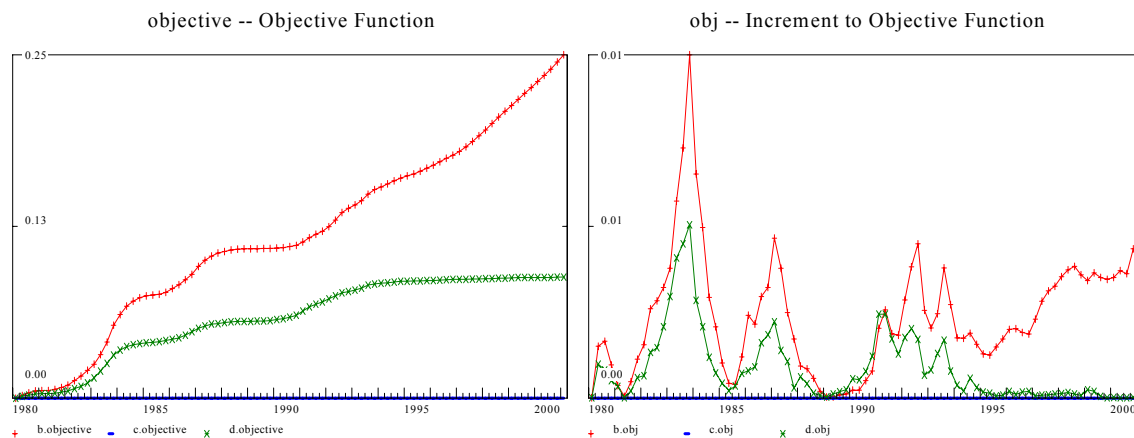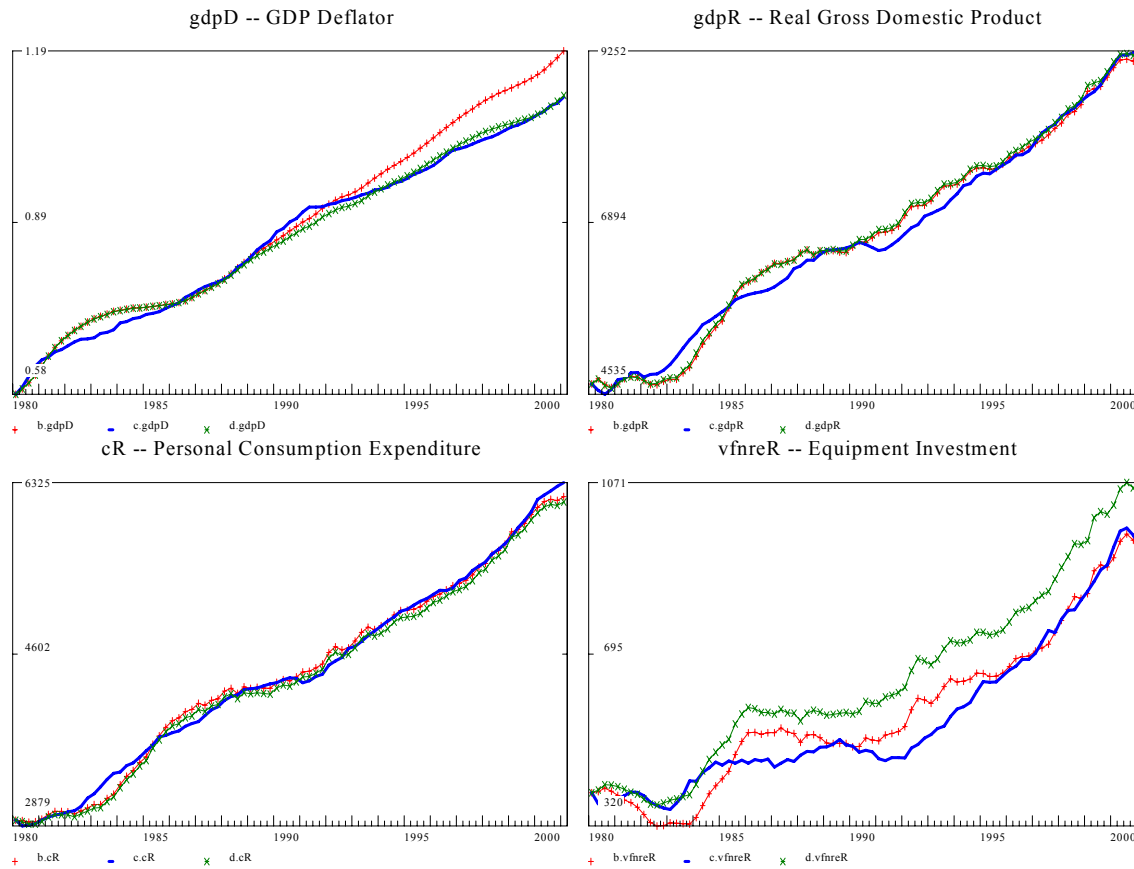
From the first graph, we see that the optimization achieved a 65 percent reduction in the objective function. The second graph shows that the contribution to the error fell essentially to zero over the last five years. I must confess that I was surprised by how much was achieved by such small changes in so few parameters. The second and third graphs show that the main improvement lay in the GDP deflator, while real GDP was little changed.

However, the last two graphs, especially the last, point to a problem. The simulation of equipment investment in the optimized model is terrible! In specifying our objective function, we implicitly hoped that if we had a good simulation for real GDP, we would have a good fit for its components. That hope, however, proved false. The lesson seems to be that if some parameters of the equation for a particular variable are included in the optimization, that variable needs to be in the objective function.

With that lesson in mind, we go back and respecify the objective function to include both equipment investment and personal consumption as follows:

```
fex gdpRX = gdpR
fex gdpDX = gdpD
fex vfnreRX = vfnreR
fex cRX = cR
f miss = @sq((gdpR-gdpRX)/gdpRX)+@sq((gdpD-gdpDX)/gdpDX) +
    0.1*@sq((vfnreR-vfnreRX)/vfnreRX) + 0.1*@sq((cR-cRX)/cRX)
```

55

With this revised objective function, the optimized coefficients in comparison to the original values were as follows

```
        Resulting coeficients after optimization (108 runs).
        Variable      Old:        New:
        vfnreR
          intercept   36.8245    -86.2489
          ue          -4.6046      9.5125
        cRpc
          intercept  785.4286    797.5327
          yRpc         0.7758      0.7600
          intsavRpc   -0.4875     -0.3995
          ur        -1416.29     -767.88
```

With this objective function, the change in the equipment investment equation is more substantial, and its unemployment term takes on a stabilizing role. In the consumption equation, on the contrary, the stabilizing role of the *ur* is reduced. The coefficient on income, where we were concerned about simultaneous equation bias, is little changed from the least-squares estimate. The reduction in the coefficient on *intsavRpc* also reduces the stabilizing effect of this variable.

As before with the simpler objective function, we get a substantial reduction in the objective function, in this case, 57 percent. Again, the biggest improvement is in the GDP deflator, where we achieve essentially a perfect simulation over the last eight years. The equipment investment simulation, as hoped, is much improved, though the performance in the last few years is not quite as good as in the model before optimization. Its weight in the objective function should perhaps be increased. All in all, however, the optimization appears to have fixed the most striking problem with the original Quest, namely, the upward creep of the GDP deflator.

gdpD -- GDP Deflator

gdpR -- Real Gross Domestic Product

cR -- Personal Consumption Expenditure

vfnreR -- Equipment Investment
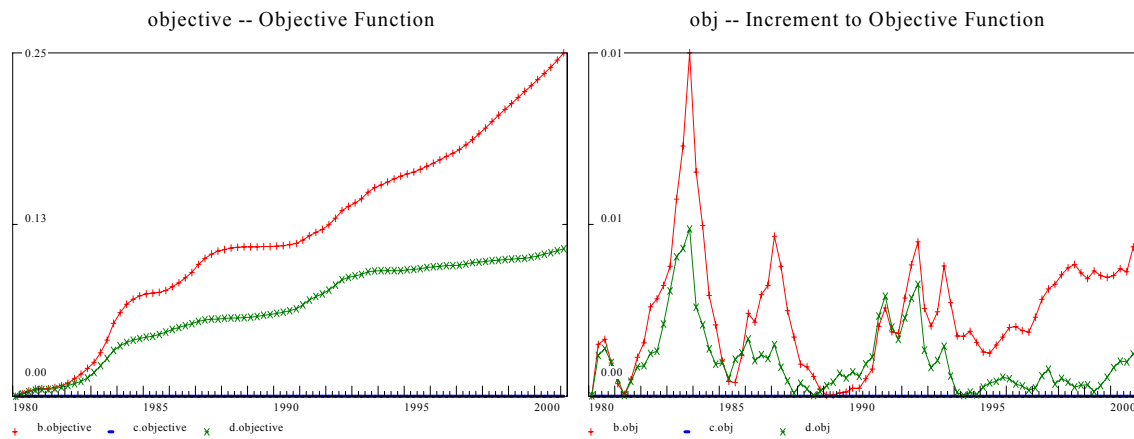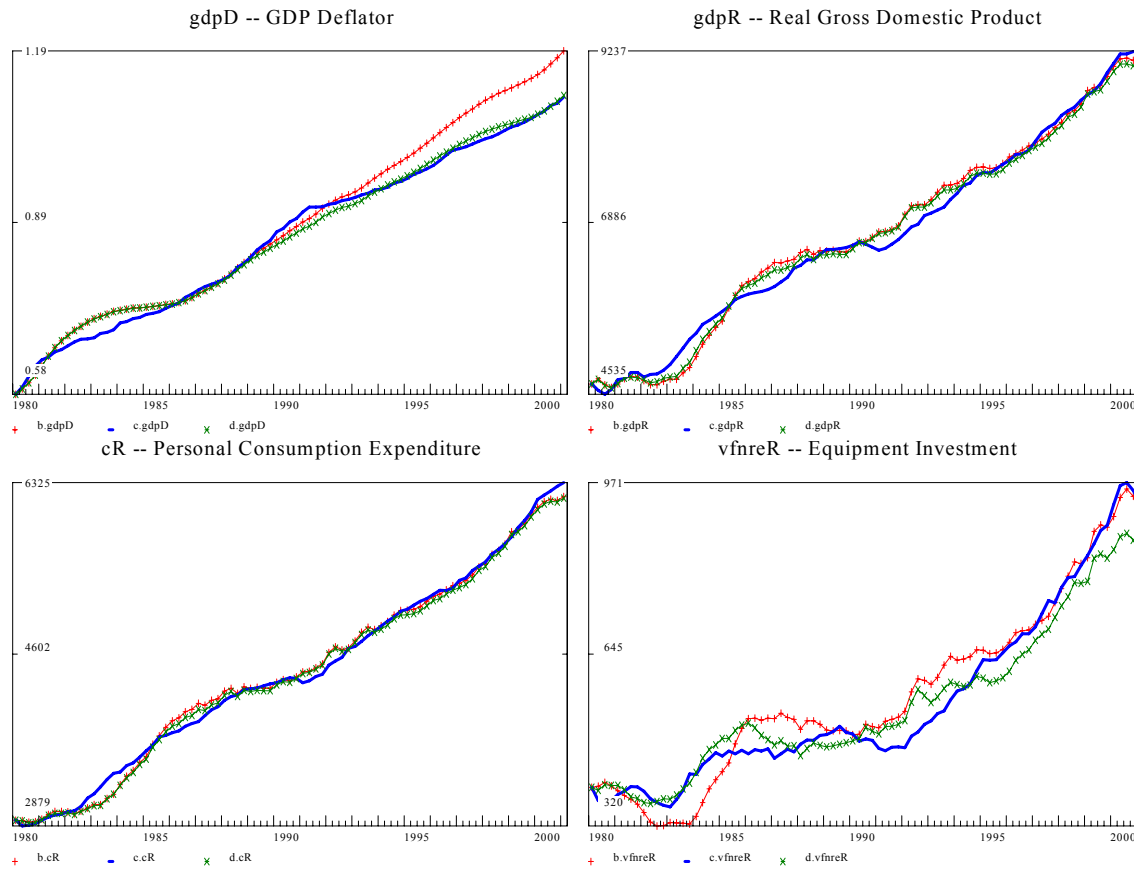
Using the optimized model

How can one use the optimized model for simulation or forecasting? Let us assume that you used Fit.opt as the name of the optimization specification file. Then the optimization created a file by the name of Fit.dat in the directory with the model. It is of exactly the format of the heart.dat file which is created to hold the coefficients for your model when you ran Build. All that you need do to run the optimized model is simply to give this file the name "heart.dat". You can simply type "dos" in the G command line box and then, in the DOS window which opens type

        copy heart.dat orig.dat
        copy fit.dat heart.dat
        exit

If you now do Model | Run, the model you run will be the optimized one.

A word about step sizes

The efficiency, and indeed the success, of the optimization can depend on the step sizes. If they are taken too large, the model can be thrown into an unstable region in which it does not converge and the optimization fails. If they are chosen too small, either many iterations may be necessary to find an optimum, or, if they are really small so that there is little difference in the objective function at the different points and the points are very close together, the optimality

test may be passed almost immediately and the process halted before it has really begun. As a rule of thumb, I usually have taken the step sizes at about one percent of the parameter's initial value. If the size of your coefficients make you want to use step sizes below about .01, you should probably change the units of the variables so as to get bigger coefficients. Thus, you may need to experiment with step sizes and the units of variables to get the optimization to run smoothly.

2. Finding optimal policies

Let us turn now to finding optimal policies in a model. We will, of course, need a different objective function, one based not on closeness of fit to history but on achieving desirable social goals. We must also find a way to represent the policy variable as the dependent variable in a regression. Since this second matter requires a new technical wrinkle, let us deal with it first.

Representing policy variables by regression equations

We would like to be able to approximate a policy variable such as *pitfBR,* the federal income tax rate, by a piece-wise linear function of a relatively small number of constants, which will appear as regression coefficients and can be varied by our optimization process. Such a function is shown in the graph below.

pitfBR -- Federal Personal Tax Rate
Actual and Piecewise Linear Interpolation
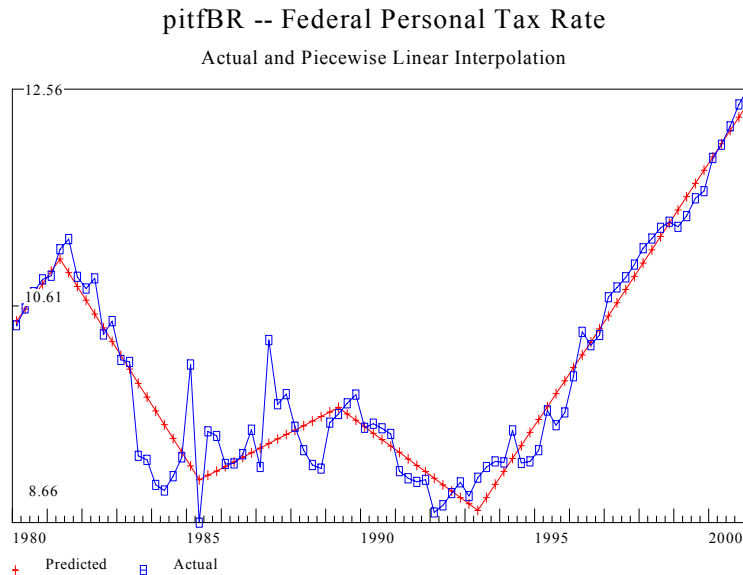
To generate the approximation by regression, we need a series of what I shall call linear interpolation functions. Each of these begins at 0 and remains 0 until its particular time interval comes; then it rises by 1 each period until the end of its interval, whereafter it remains constant at whatever value it has reached. For representing the federal personal tax rate, I took the beginning of the intervals to be the third quarter of the first year of each presidential term. Thus, except for the first which represented the tail end of the Carter policies, each of the variables rises from 0 to 16, the number of quarters in a four-year term. Here is a graph of these variables.

58

Interpolation Functions
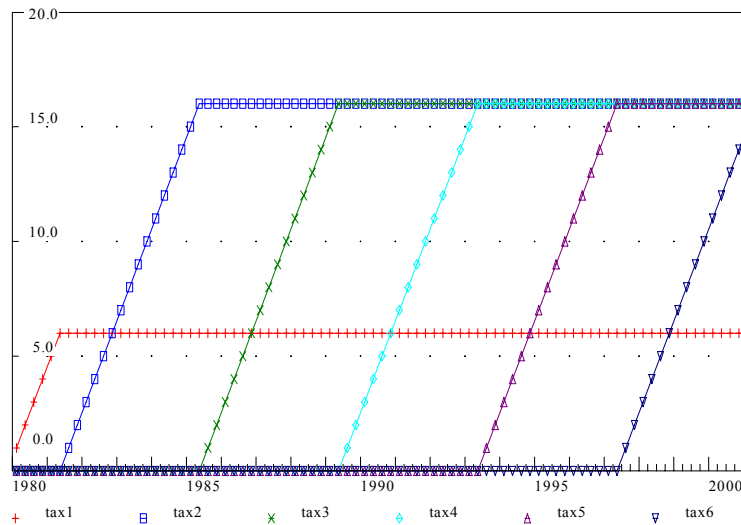
I have called these functions tax1, tax2, ..., tax6. Once we have them, we can obtain the piecewise linear approximation by a simple regression:

```
r pitfBR = tax1, tax2, tax3, tax4, tax5, tax6
```

The regression coefficients in this equation are the precisely the parameters with respect to which we optimize to find the optimal tax policy.

We could, of course, create these interpolation variables by hand and introduce them via *fex* and *update* commands into the model. G, however, offers a simpler way of generating them automatically by the *intvar* command. The command necessary to generate our six variable is

```
intvar tax 1980.1 1981.3 1985.3 1989.3 1993.3 1997.3
```

The word after the command, "tax" in this example, provides the root of the variable names which will be created by appending 1, 2, 3, etc. to this root. The dates which follow then mark the beginning of each variable's activity.

The complete regression file to compute the representation of *pitfBR* follows:

```
catch pitfBR.cat
add lim80
#  pitfBR -- Federal Personal Tax Rate

fex pTaxBase = pi - ngtpp + 0.5*nconsi + nibtax
fex pitfBR = 100.*gfrptx/pTaxBase
save pitfBR.sav
intvar tax 1980.1 1981.3 1985.3 1989.3 1993.3 1997.3
ti pitfBR -- Federal Personal Tax Rate
subti Actual and Piecewise Linear Interpolation
r pitfBR = tax1,tax2,tax3,tax4,tax5, tax6
```

59

```
        save off
        gname pitfBR
        gr *
        catch off
```

(The two *fex* commands above the *save* command are so placed because they are provided in the
Master file.) The results of the regression are

```
:                       pitfBR -- Federal Personal Tax Rate
  SEE   =       0.27 RSQ   = 0.9250 RHO =   0.37 Obser  =   85 from 1980.100
  SEE+1 =       0.25 RBSQ  = 0.9192 DW  =   1.26 DoFree =   78 to   2001.100
  MAPE  =       1.95
     Variable name        Reg-Coef  Mexval  Elas   NorRes    Mean   Beta
  0 pitfBR             - - - - - - - - - - - - - - - - -   10.02 - - -
  1 intercept           10.35670   407.2   1.03   13.33    1.00
  2 tax1                 0.11258     3.4   0.07   13.07    5.82   0.091
  3 tax2                -0.12431    67.4  -0.17   12.47   13.46  -0.655
  4 tax3                 0.04077    11.9   0.04   10.75   10.45   0.297
  5 tax4                -0.05790    23.3  -0.04    9.32    7.44  -0.445
  6 tax5                 0.11661    73.5   0.05    2.24    4.42   0.790
  7 tax6                 0.11938    49.8   0.02    1.00    1.41   0.436
```

Because of the progressivity of the income tax, growth in real income increases this average tax
rate. This steady upward movement during the Carter and Clinton administrations is evident in
the coefficients of *tax1, tax5*, and *tax6*; the sharp cuts of the first Reagan administration shows
up in the negative coefficient on *tax2*. The administration of George Bush, contrary to the
impression of many, cut taxes substantially, as seen in the coefficient of *tax4*.

Once this regression has been performed, it is introduced into the Master file just as any other
regression with the lines

```
        #  pitax -- personal taxes and non-tax payments
        f pTaxBase = pi - ngtpp + 0.5*nconsi + nibtax
        fex pitfBR = 100.*gfrptx/pTaxBase
        # add regression for tax rate to allow optimization
        add pitfBR.sav
        id   gfrptx = .01*pitfBR*pTaxBase
```

(There is a reason for the factor of 100 in the definition of pitfBR; originally it was not there, and
all the regression coefficients were 1/100 of the values shown above. The appropriate step size
in the optimization therefore seemed to be about .00001. With this step size, the optimization
stopped very quickly at a point very close to the initial point. In other words, it failed to
optimize. Evidently, the small step size allowed the termination test to be passed long before it
should have been. From this experience came the advice given above that the step sizes should
not be too small.)

Putting in this additional regression meant that the optima.dat file from the optimization of the
previous model no longer matched the heart.dat file for this new model. Consequently, before
putting in a new objective function, I reoptimized this model with the historical fit objective
function to get an Optima.dat file which could later be copied to Heart.dat so that the tax
optimization should be done with the model optimized for fit. In this step, I gave at the] prompt

not only the "skip sp500R" command but also "skip pitfBR" command to use precise historical tax rates in optimizing for fit.

In the next section we will develop a "socially deplorable" objective function to be minimized to which we may give the name *misery*. The specification of parameters to be varied to minimize the last value of this "misery" function are given by the following FedTax.opt file:

```
misery
20
#Optimize tax rate
pitfBR
#   1     2     3     4     5     6     7
    1   .01   .01   .01   .01   .01   .01
```

A Misery function

Specifying a socially desirable objective function, or its negative to be minimized, is not necessarily easy. I began with minimizing what has been called the "misery index," the sum of the unemployment rate and the unemployment rate. The optimization quickly drove unemployment negative so that $1/u$ in the consumption function became a huge negative number and the model simply broke down with attempts to take logarithms of giant or negative numbers. I then went over to the sum of the *squares* of these two misery indicators. That worked better, but took no account of the budget deficit. Paying interest on the federal debt imposes an efficiency loss in collecting the taxes with which to pay it, so I added a third misery indicator, the ratio of interest on the federal debt to GDP. Finally, to give about equal weight to all three, I took 2 percent unemployment as ideal, rather than 0 percent. The resulting objective function was then expressed by these lines in the Master file.

```
# For optimal tax
fex obj1 = 0
fex obj2 = 0
fex obj3 = 0
f obj1 = @sq(u - 2.)
f obj2 = @sq(infl)
f obj3 = @sq(100.*gfenip/gdp)
f obj = obj1+obj2+obj3
fex misery = 0
f misery = @cum(misery, obj, 0.)
```

Note that both objective functions, *misery* and *misses* and perhaps others, can be included in the model. To optimize policy in a model that has already been optimized for fit, I copied the Optima.dat file (created in optimizing for fit) to Heart.dat with the G command
        dos copy optima.dat heart.dat
and then did Model | Run again to minimize *misery* using the FedTax.opt file shown in the preceeding section.

The old and new coefficients are shown below.
        Changes in Federal Taxation
        Variable   Historical: Optimal

Intercept	10.3567	10.1689
Carter	0.1126	0.2090
Reagan I	-0.1243	-0.0450
Reagan II	0.0408	0.0807
Bush	-0.0579	-0.2317
Clinton I	0.1166	-0.0622
Clinton II	0.1294	0.2347

The new tax rates resulting from the optimization are shown by the (red) line marked with + in the first graph below. The optimal policy would have been higher taxes in the Reagan years, a rapid drop in the Bush administration, continued low rates in the first Clinton administration, followed by a sharp rise in the second.   The second graph shows that, quite unlike the objective in the optimization for fit, in this policy optimization the historical policy would have been better than the optimal one up to 1995. We seem to have a clear case of the usual macroeconomic dilemma: what is pleasant in the short run is painful in the long run and vice-versa.

The next three graphs show the effects of the tax change on the three components of the misery index we are minimizing. All three are plotted on the same scale to facilitate comparison of the contribution. The following three show these variables in the misery index in more customary units without squaring; the last two graphs show real GDP and the GDP deflator.

The optimal tax policy accepted a bit more unemployment and some loss in real GDP early in the simulation in order to get higher real GDP, lower unemployment, and much lower prices near the end of the period. Inflation with the optimized tax rate is lower throughout the period except for the last three years where it rises slightly. The interest component of the objective function is uniformly reduced. Though this component does not have the spike in the early 1980's that the others do, the difference between the two lines is of similar magnitude to the differences of the other two indicators.

Federal Personal Tax Rate
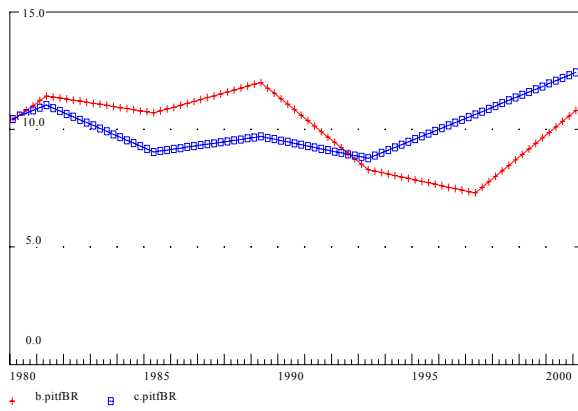
Objective
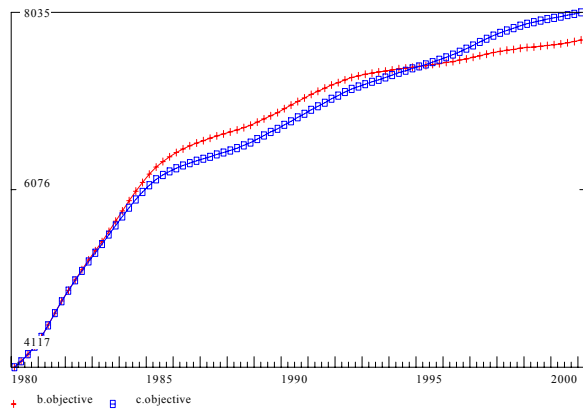
Obj1 Unemployment
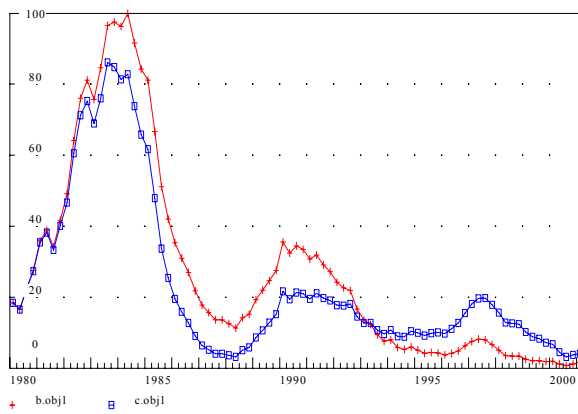
Obj2 Inflation
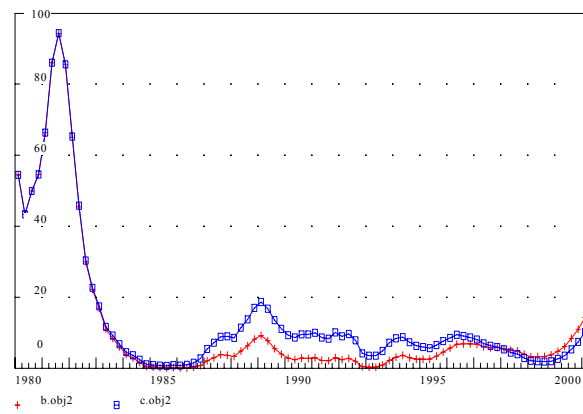
Obj3 Interest
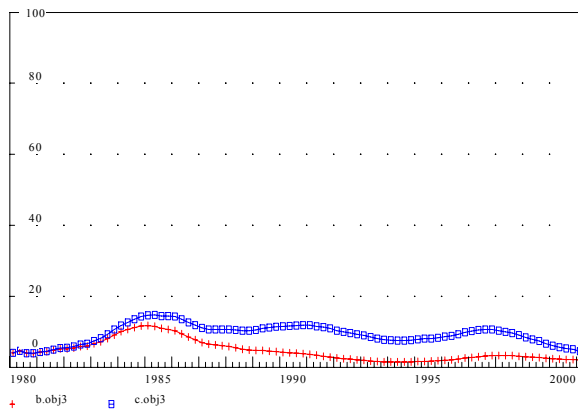
u -- Unemployment
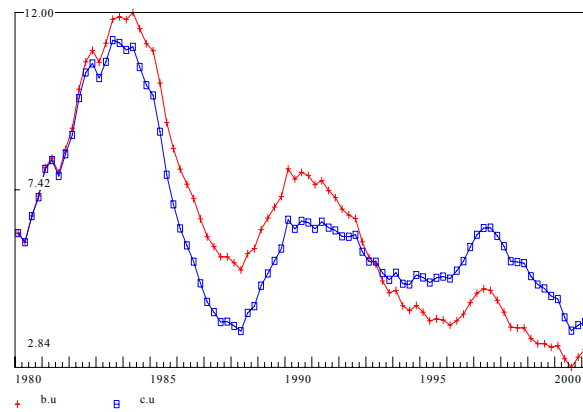
infl -- Inflation Rate of GDP Deflator
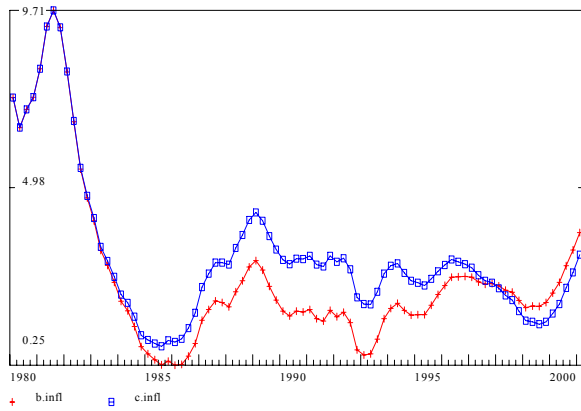
Federal Interest as Percent of GDP
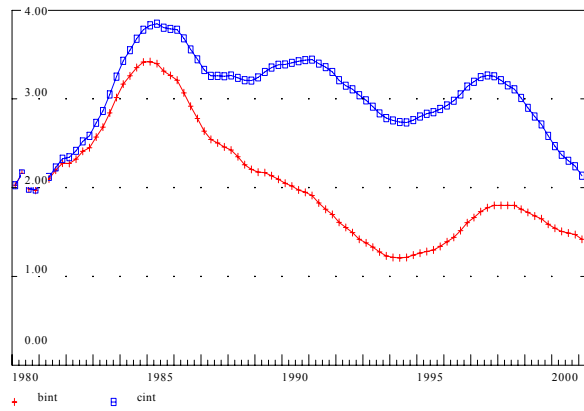
gdpR -- Real Gross Domestic Product
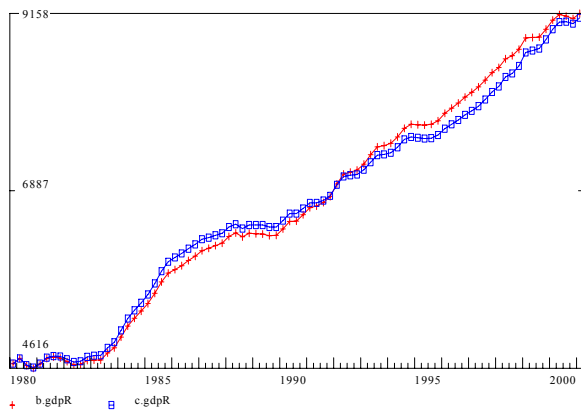
gdpD -- GDP Deflator
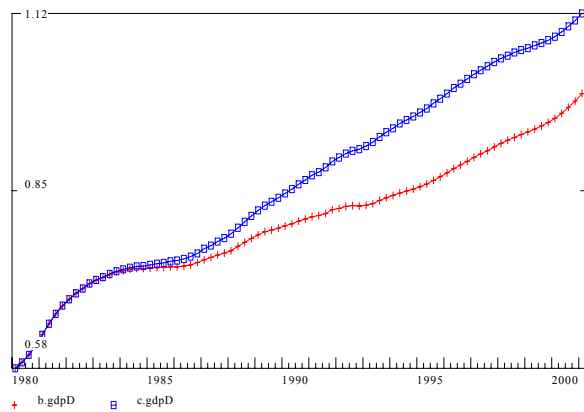
Though these results bear out my own beliefs that the Reagan tax cuts were utterly irresponsible, different objective functions would give different optimal policies. The exercise does, however, illustrate how models can be used in designing of policies.

Chapter 10. Probability Theory and Regression

If you have studied regression previously, you may have been slightly shocked that I have not yet said anything about "testing of hypotheses" or the "significance" of variables. My reticence to write about these topics stems from a profound doubt of their appropriateness for use in model building. There is, to be sure, a standard method which is explained with conviction in scores of text books. In years of experience, I have found that the assumptions on which it rests have little to do with the reality in which I, as a builder of economic models, am working. I have therefore emphasized other ways of working.

If, however, I were to omit all mention of this other, probabilistic approach and of the beautiful theorems which can be proved using it, then, if you already know this material, you might well conclude that I am ignorant of this wonderland. If, on the other hand, you are new to this work and, having finished this book, go out into the world and find people discoursing learnedly of efficient, consistent, blue estimators about which you know nothing, you may feel that your education was deficient. So in this one chapter, and to some extent in the last, we need to look at what this other approach has to offer. You must allow me, however, to explain from time to time as we go along why I do not put much stock in the methods.

To avoid boring you -- and to stress the metaphysical, transcendent nature of the assumptions -- I will begin the account in the form of a fable.

1. The Datamaker Fable

These methods begin from what is best regarded as a sort of creation fable. According to this fable, there is a "true" equation with coefficients β and the Great Datamaker, who knows β, has picked a matrix of explanatory variables, *X,* once and for all and has then generated many vectors, *y*, by picking vectors of random errors, *e*, and calculating

$$y = X\beta + e.$$

(Datamaker is called The Data Generating Process by many of his devotees.) Because *y* depends on random elements, it is said to be stochastic. *X*, by contrast, is fixed, non-stochastic. Datamaker has thrown into the universe many such y vectors, each bundled with the true *X* matrix. One of these struck the earth, and we have had the good fortune to come upon it. Our job is to figure out what β is. Datamaker sometimes plays a trick on us and includes in *X* a variable which in fact was not used in making up *y* -- or which had a zero value in β. We must be very careful to detect any such jokers and must not include them in our estimated equation.

Everyone who has caught such a sample computes the least-squares estimate of β by

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

Now there are many, many beings throughout the universe who catch the packets thrown by Datamaker. Each one computes b, all using the same value of X but each having a different y. (There are no others on Earth, however, because on Earth the y of th bundle happens to be called something like "Consumer expenditures on automobiles in the USA, 1960 to 1995." We never see that piece of history rerun with other values of e, so we must suppose that Datamaker has sent his other bundles flying elsewhere in the universe.) The b's computed by beings all over the universe are thus random variables, since each depends upon the e used by Datamaker in making up its y. We may therefore speak of their expected values (or means), their standard errors, their variances and covariances, just as of any other random variables. Expressing our b in terms of the true β and the random error vector, e, that happened to be used in making up the particular y vector we caught, we have

$$b = (X'X)^{-1} X'(X\beta + e) = \beta + (X'X)^{-1} X'e.$$

If we assume that the expected value of e is 0, $(E(e) = 0)$ then

$$E(b) = \beta + (X'X)^{-1}X'E(e) = \beta.$$

(The first equation follows because X is constant and non-random; the second because $E(e) = 0$.) Thus, the expected value of b is β, and we say that b is an unbiased estimate of β. That means that if the b's computed throughout the universe are all sent to Universal Central Data Processing and averaged, the average would be β. That is supposed to make us feel good about the one and only b we will ever see.

If we assume that the elements of e are independent and all have the same variance, σ^2, -- so that $E(ee') = \sigma^2 I$ -- then we can calculate the variances and covariances of the elements of b by taking the expected value of $(b - \beta)(b - \beta)'$, thus

$$E(b - \beta)(b - \beta)' = E((X'X)^{-1} X'e \ e'X(X'X)^{-1})$$
$$= (X'X)^{-1} \sigma^2 I X(X'X)^{-1}$$
$$= \sigma^2 (X'X)^{-1}$$

so the variances of the b's are the diagonals of this matrix; the standard deviations are their square roots. If we knew σ^2, we could calculate the standard deviations precisely. In fact, we never know σ^2 and must estimate it. The most natural estimate might be r'r/T, the variance of the residuals. This estimate would be biased, for -- as we shall show --

$$E(r'r) = (T - n)\sigma^2,$$

where T is the number of observations or rows of X and n is the number of independent variables, or columns of X. To see why this formula holds, note first that

$$r = y - Xb = X\beta + e - X(X'X)^{-1}X'(X\beta + e)$$
$$= e - Me$$

66

where $M = X(X'X)^{-1}X'$. This M is a remarkable matrix. Note that $M = M'$, and $M'M = MM = M$ and that

$$tr\ M = tr\ (X(X'X)^{-1})X' =\ tr\ X'X(X'X)^{-1}\ =\ tr\ I = n,$$

where *tr* indicates the *trace* of a square matrix, the sum of its diagonal elements, and I is the (n,n) identity matrix. (The second equality uses the property that tr(AB) = tr(BA) if both products are defined..) Now

$$r'r = (e - Me)'(e - Me) = e'e - 2e'Me - e'M'Me = e'e - e'Me.$$

Since r'r is (1,1), r'r = tr r'r. So

$$
\begin{aligned}
E(r'r) &= E(tr\ r'r) = E(tr(e'e - e'Me)) = E(e'e) - E(tr(ee'M)) \\
&= T\sigma^2 - tr(E(ee'M))\quad \text{(Since expected value of a sum is the sum of the expected} \\
&\qquad\qquad\qquad\qquad\qquad\qquad\text{values.)} \\
&= T\sigma^2 - tr(\sigma^2 IM)\qquad \text{(Where I is T by T)} \\
&= T\sigma^2 - \sigma^2(tr\ M) = (T - n)\sigma^2.
\end{aligned}
$$

Thus, if we use $s^2 = r'r/(T - n)$, we will have an unbiased estimate in the sense that $E(s^2) = \sigma^2$.

This is indeed a remarkable result, for it tells us the variance of all the b estimates flowing into Universal Central Data Processing solely on the basis of our one pathetic sample!

The *t-values* are the ratio of each regression coefficient to the estimate of its standard deviation made using this s^2.

If the e are normally distributed, and if the true value of some β_i is zero, this ratio will have a Student t distribution. (A good bit of mathematics is required to back up that simple statement; see my book *Matrix Methods in Economics* (Addison-Wesley, 1967) Chapter 6. I have used the expression "t-value" to mean something which, under some assumptions, has a Student t distribution without, however, alleging that those assumptions are in fact valid.) This Student distribution depends on T-n, but for values of T - n over 30, the distribution is indistinguishable from the normal. So if T-n > 30, then under all of the previous assumptions -- namely the existence of a true equation of the form we are estimating, X non-stochastic, and the elements of *e* independent of each other but all having a normal distribution with zero mean and the same variance -- we can say, "If the true value of the regression parameter is zero, the probability that we will observe a t-value of over 2.0 in absolute value is less than .05." If we observe such a value, we are then supposed to be "95 percent *confident*" that the true value is different from zero, and we are entitled to say that our variable is "statistically significant at the 5 percent level."

67

You may be advised to discard variables that are not statistically significant at some specified level, often 5 percent, and then to re-estimate the equation so as to get an equation in which all variables are significant. There is, however, a serious problem in following this advice, as we shall see in the next section.

A further commonly used statistic must be mentioned, namely Fisher's F, named for Sir Ronald A. Fisher (1890 - 1960), who found the exact mathematical distribution of this and numerous other statistics and (alas!) popularized the use of significance tests in the social sciences. The F-test uses exactly the same assumptions as does the t-test but may be applied to test whether several elements of β are all zero.

If one regression has used m independent variables and produced a sum of squared residuals of SSR_m and a second regression has just added more independent variables to it to reach a total of n and produced a sum of squared residuals of SSR_n, then the F statistic for testing the significance of the extra variables is

$$F = \frac{(SSR_m - SSR_n)/(n - m)}{SSR_n/(T - n)}.$$

This F is said to have n - m degrees of freedom in the numerator and T - n in the denominator. Tables of values of F for various levels of significance may be found in most statistics textbooks. The derivation of the distribution of the F statistic is fully derived in my book cited above.

If you want G to show you the t- and F-values, give it the command

```
showt y
```

To turn off the showing of these values, use

```
showt n
```

The t-value that appears for each variable is for testing whether its β coefficient is zero. The F-value for each variable is for testing whether *its and all following* β coefficients are zero.

If you are seriously interested in testing, you should also ask whether the error terms in your equation are normal. The usual procedure is to examine moments of the residuals, μ_2, μ_3, and μ_4. The β_1 test statistic for symmetry and the β_2 test statistic for peakedness or kurtosis are then

$$\beta_1 = \frac{\mu_3^2}{\mu_2^2} \quad \text{and} \quad \beta_2 = \frac{\mu_4}{\mu_2^2}$$

For the normal distribution, $\beta_1 = 0$ and $\beta_2 = 3$. If one is willing to make the assumption that the distribution belongs to the rather large class of distributions known as the Pearson family (which includes the normal), then a convenient test statistic is offered by the Jarque-Bera statistic

$$ J = T\left(\frac{\beta_1}{6} + \frac{(\beta_2 - 3)^2}{24} \right). $$

which, under the hypothesis that $\beta_1 = 0$ and $\beta_2 = 3$, has a chi-square distribution with two degrees of freedom. J will be less than 5.99 95 percent of the time and less than 9.21, 99 percent.[1] To get these statistics in G, use the command

      normality <y | n | f>

This command turns on (or off) tests of normality of the error terms. The command can be abbreviated to just *norm*. If the option chosen is 'y', then the Jarque-Bera test appears on the standard regression result screen labeled "JarqBer". If the option 'f' (for full) is chosen, then before the regression results are shown, a table appears with moments and measures of symmetry and peakedness, as well as the Jarque-Bera statistic. If a "catch" file is active, this table will go into it.

A final question is the relation of the t-values to the mexvals presented by G. In the notation of Part1, Chapter 2, Seciton 8, we have seen that

$$ mexval_i = 100 * (sqrt(1+(a^2_{im}/a_{ii}a_{mm}))-1). $$

In this same notation, the t-value for the ith variable is

$$ t_i = a_{im} / sqrt(a_{mm}a_{ii}/(T - n)) $$

So in terms of t, the mexval for the same variable is

$$ mexval = 100 * (sqrt(1 + t^2/(T-n)) - 1). $$

Thus, in the same equation where T - n is the same for all variables, t-values and mexvals convey the same information. The difference is in ease of interpretation, ease of explanation, and "honesty". The mexval is exactly what it claims to be; the t-value has a t-distribution only if the Datamaker assumptions are valid.

In comparing the usefulness of variables in equations of differing values of T - n (the "degrees of freedom") there is, of course, a difference in informational content. A variable with a t-value of 3 in a regression with 10 degrees of freedom would be sorely missed if dropped -- mexval = 40; whereas a variable with the same t-value in an equation with 1000 degrees of freedom, while more "significant" by the t-test, could be dropped without noticeable effect on the fit -- mexval = 0.45. If you believe in Datamaker, you will like t-values; if not, mexvals may tell you exactly

---

[1] This test is suggested in Jan Kamenta, *Elements of Econometrics,* 2nd Edition, New York, Macmillan 1986, pp. 286-287. He cites an unpublished paper by C.M. Jarque and A.K. Bera from 1981.

what you want to know in the two cases, while you will find the t-values to be tricky to compare.

2. Datamaker with a Stochastic X Matrix

In hardly any of the behavioral equations of the Quest model or other econometric model are the independent variables non-stochastic; in most of them the independent variables are current or lagged values of endogenous variables of the model. It is, of course, wholly illogical to assume that what is a stochastic variable in one equation is non-stochastic in the another. It is therefore natural to ask, Can we extend the "unbiased-estimator" result in some way to stochastic X matrix?

Although one can still do a few things with the expected value concept, it will prove much more fruitful to use the concept of a *probability limit* or *plim*. Suppose that x is a random variable and that m(t) is a function of t observations on x. (For example, m(t) might be the mean of t observations of x.) If there exists a number a such that for any $\varepsilon > 0$ and any $\delta > 0$ there exists a positive integer N such that when $T > N$,

$$\text{prob}(\, | \, m(T) - a \, | > \delta \,) \; < \; \varepsilon,$$

then a is said to be the probability limit of m(t) or plim m(t) = a.

Probability limits have an important property not shared by expected values. If $E(x) = a$ and $E(y) \, = \, b$, it is not generally true that $E(xy) = ab$. If, however, plim x = a and plim y = b, then plim xy = ab and plim x/y = a/b if b is not zero. This should be more or less obvious if you are accustomed to working with limits. Its detailed proof is more laborious than enlightening and we will skip it.

Suppose now that

$$y(t) = x(t)\beta + e(t)$$

where x(t) is a row vector of the independent stochastic variables. Let us now make estimates of β using T observations, thus

$$b(T) = (X(T)'X(T))^{-1}X(T)'y(T)$$

where we have shown the familiar X and y matrices explicitly as X(T) and y(T) to emphasize that they have T rows. If now plim b(T) = β, we say that b(T) is a *consistent* estimate of β. This "consistency" concept corresponds in the case of stochastic x(t) to the "unbiased" concept for the non-stochastic X matrix.

Is the least-squares estimator consistent? If we multiply both sides of the first equation of this section by x(t)' and average over T observations, we get

$$(\Sigma x(t)'y(t))/T = ((\Sigma x(t)'x(t))/T)\beta + (\Sigma x(t)'e(t))/T.$$

Note that if we neglected the last term on the right and solved for β we would have the least-squares estimator of ß. If we can argue that the plims of all three terms exist and that the plim of the last term is zero, then we can reasonably claim -- using the algebraic properties of plim mentioned above -- that our estimator is consistent. The existence of the first two plims is usually handled by assumption, namely, we assume that the sample grows, not by actually going forward in time into new, uncharted seas, but by rerunning history, by calling on the cosmic Datamaker to give us more observations from the same historical period on which we already have one. Then we don't have to worry about x variables with trends that don't have plims.

What can we say about the last term, the plim $(\Sigma x(t)'e(t))/T$? If the elements x(t), though random variables, are actually determined before t -- perhaps they are lagged values of endogenous variables -- and e(t) is not correlated with the error terms which went into determining them, then we can reasonably assert that plim $\Sigma x(t)'e(t)/T = 0$ and that the least squares estimates are consistent.

 But consider two other possibilities.

1.   One of the elements of x(t) is determined in period t and depends on y(t). For example, in the AMI model of Chapter 1, consumption depends on income in the same period and this income depends, via the GDP identity, on consumption. Thus it becomes impossible to hope, much less to argue, that income is uncorrelated with the errors in the consumption function. The least squares estimator of the consumption equation is then inconsistent, and increasing the sample size will only make it home in on the wrong values. This situation, mentioned in the development of the Quest model, is known as *simultaneous equation bias* or inconsistency.

2.   The elements of x(t) were all determined before period t but e(t) is autocorrelated, that is, plim e(t)e(t-1) ≠ 0. Suppose, to take the worst case, that y(t-1) is among the elements of x(t). Then e(t) is correlated with y(t-1), which is an element of x(t), so plim $\Sigma x(t)'e(t)$ cannot be zero and the least-squares estimator is inconsistent. Even if y(t-1) is not among the elements of x(t), there could be other endogenous variables determined at time t-1 and depending on y(t-1) so that a relation between them and e(t) would creep in. Note that autocorrelation of the residuals, a sign only of inefficiency under the assumption that X is non-stochastic, becomes -- in the stochastic case -- a warning of the possibility of the more serious sin of inconsistency. Indeed, if the lagged value of the dependent variable is among the independent variables, it is as good as conviction.

Clearly, there are degrees of inconsistency. It may exist without being a serious problem if the relation between e(t) and the suspect element of x(t) is weak or if the fit of the equation is very close. But we may need ways to deal with it. Some are presented in the rest of this chapter.

3. Does the Datamaker Fable Apply to Our Work?

Clearly, Datamaker has a lot going for him. The assumption of his existence makes all this beautiful mathematics applicable to the real world. Indeed, there is much more mathematics that can be developed by elaborating on Datamaker. There is a whole profession that works on these further elaborations. To question the existence of Datamaker is even more socially disturbing than harboring doubts about Santa Claus. And yet, we cannot avoid the question, Does the fable apply to our work?

What possible meaning can be given to β or to the variances and covariances of b? Can we take seriously the idea that there is a true equation of the form that we are fitting? Suppose, for example, that we are studying the demand for automobiles. In fact this demand depends upon the decisions of myriads of individuals subject to myriad influences. One person buys a new car because he has just wrecked his old one; another, who was planning to buy, postpones her purchase because the price of personal computers has dropped and she has decided to buy a computer instead of a car. Another couple is having a baby and needs a different kind of car. We formulate an equation that says that automobile purchases depend on relative prices and income. Can we take seriously the idea that there is a "true" equation of this vastly over-simplified form? Can we honestly suppose that we know exactly what variables are in the X matrix when we know that inside of five minutes we will try other independent variables? Can we even make believe that the variables in X are non-stochastic, fixed values when we know that tomorrow or next week we may be studying almost any one of them as the dependent -- and therefore stochastic -- variable in another regression?

Though my answer to all these questions is a resounding "No" for most regressions based on time series, there is a situation where speaking of β and of the standard deviation of the estimated regression coefficients seems to be to be perfectly meaningful. Indeed, this situation shows clearly why these concepts are not meaningful in most regressions based on economic time series.

This other situation may arise when regression is applied to a sample of cross section data. Suppose that *y* is salaries of faculty members at the University of Maryland where I teach. I draw a random sample of fifty salaries from the personnel records and compute the mean, that is, I regress the salaries on just a constant term. What is this regression coefficient an estimate of? Clearly it is an estimate of the mean of the whole population, that is, of all faculty members at the University of Maryland, and it makes perfect sense to compute its standard deviation. Now suppose I add to the independent variables dummy variables for the academic rank of the faculty members and do a regression. What now are the b's estimates of? Equally clearly, it seems to me, *they are estimates of the coefficients which would be found if the regression were done on the whole population,* that is, on all faculty at this university. Their variances and covariances are

perfectly meaningful. No more than with the mean, however, are they meant to be an estimate of the true way that the university sets salaries. They are just one way of describing the salary structure at the university.

All is fine so long as we are working with a sample. But what if we now get the personnel records of all faculty and run the regression on the whole population. The b which we then get is what a moment ago we were calling β. The regression program will, of course, if you let it, spit out the variances and covariances of these $β$, but they are utterly meaningless, for $β$ is a constant vector whose true value we have now found!

In doing this regression, we are looking for a *description* of the salary structure at this university. We are not claiming that any such equation and random mechanism is actually used in setting salaries.   When we have all the possible observations, our regression gives us the description we want. Only by dragging in some far-fetched concept of all possible Universities of Maryland can one find any meaning for these variances and covariances. Notice also that if we had all but one or two of the faculty members, our b would be a much better estimate of $β$ than would be indicated by the variances and covariances. Thus we see that as our sample size increases towards the whole population, the regression coefficients become better and better descriptive statistics while their variances and covariances become more and more meaningless.

If we return now to the time-series regression which are our main concern in this book, what can we say about the nature of β? If I estimate a regression for investment over the period 1975 - 2000, I would claim only that it is a *description* of investment behavior in the last quarter of the 20th century. I have *all* the data, the whole population, not a sample cast my way by a cosmic Datamaker. The equation may be a good description or a bad description, depending on how well it conforms to the "good advice" of Chapter 6, Part1. But it is not an *estimate* of an unknown, *true* β. If I use the equation to forecast to 2010, I would only be trying to see what will happen if my description remains valid.

Thus, if we are to be serious, we have to admit that variances and covariances of our regression coefficients and the tests based on them make little or no sense. We must admit that we are simply fitting grossly oversimplified equations to a complex reality. Instead of testing, testing, testing as advised by some, we must ask the plainer but harder questions of the "good advice" in Chapter 6. We must think, compute, and re-think to get as good a description as we can, one that would be workable in the sense that forecasts made with it are helpful and counter-historical simulations contribute to understanding the effects of policy.

In this way, we also avoid the "pretest" trap that plagues those who would rely on testing. Anyone with much experience in building models will admit that when we begin studying a question with regression, we don't know which variables to include among the explanatory set. So we generally include a lot of variables that prove to have little or no explanatory value as shown by t-tests. So we throw them out and present the final equation with nice, "significant" coefficients on all the variables. What is wrong with that? Well, when we threw a variable out, we may have been making a mistake. Maybe it really did have a non-zero coefficient in β. We

really have no idea how likely it was that we made such a mistake. We know that, if we were using a 5 percent t-test, that there was a .05 probability that we would not throw it out even though we should have, but the probability of the other mistake – often called a type II error -- is unknown. But this other mistake can kill us, for if we threw out a variable that belongs in, then we are not estimating the true equation. And if we are not estimating the true equation, all the formulas for variances and covariances are wrong and all the tests invalid.

Thus, while at first it seemed that Datamaker's habit of throwing in to *X* some jokers that were not really used in making *y* was pretty innocuous, on closer inspection it turns out to be a really nasty trick that brings the application of the theory to a most embarrassing impasse. From a practical point of view, we have to experiment to find variables that work. But as soon as we do, any claim that we are making valid tests of hypotheses is untenable.

The same problem does not arise if we admit that we are just looking for plausible though much over-simplified descriptions of behavior. One who has relied on (probably invalid) t-tests may suppose that once one drops t-tests, any old equation that fits the data is acceptable. Actually, nothing could be farther from the truth. The discipline of plausibility along the lines of the "good advice" of Chapter 6 is far stricter than that of "significant" t-tests.

4. Are There Better Alternatives to Least-Squares? The Gauss-Markov Theorem and Generalized Least Squares

One may, however, accept the idea that regression coefficients are descriptions, not estimates of some unknowable, true parameters and still ask whether there might be better descriptors. And here Datamaker's supporters have a fall-back position. They may say, "All right, we will put aside testing hypotheses. But suppose, just for the sake of argument, that the data were created more or less as described by the Datamaker story with exactly the equation you have selected by following all the rules of 'good advice.' Wouldn't you want the fitting process to come up with a good approximation of that true equation?"

If you say, "Not especially. I want nothing to do with that ridiculous Datamaker," you will be following the practice of many builders of applied models, and I'll have no objection. I myself, however, am a little more tolerant of belief in Datamaker. I don't want to be accused of blasphemy against Datamaker only to worship Ordinary Least Squares. So, if assenting to this question leads us to ways that get better descriptions, descriptions that are more plausible and hold up better over time, why not look at them? It is in that spirit that the rest of this chapter looks at some alternatives to ordinary least squares suggested by pursuing this limited Datamaker idea.

First of all, however, we need to recognize that ordinary least-squares (OLS) may have some pretty good properties. There is a remarkable proposition, known as the Gauss-Markov theorem, which establishes conditions in which OLS is hard to improve upon. This theorem states that if the data is generated by a Datamaker process but without necessarily using normal errors, then

least squares will be the minimum-variance unbiased estimators that can be expressed as a linear function of the dependent variable.

More specifically, if $y = X\beta + e$, with $E(e) = 0$ and $E(ee') = \sigma^2 I$ while X is fixed and non-stochastic, then not only is the least squares estimate of ß unbiased, in the sense that $E(b) = \beta$, but it is the "best linear unbiased estimate" in the sense that a property summarized by saying that the estimate is "blue." A "linear" estimate in the sense of this theorem means one that can be calculated as a linear combination of the y, that is by multiplying some constant matrix times y. Note that the least-squares estimate qualifies as linear for it is obtained by premultiplying y by $(X'X)^{-1}X'$. "Best" in the sense of this theorem means having the smallest variance. An estimating method which achieves this smallest variance is said to be *efficient*.

To demonstrate this proposition, let c be another linear, unbiased estimate of ß which we may without any loss of generality suppose to be given by

$$c = ((X'X)^{-1}X' + C)y$$

where C is a constant matrix depending perhaps on X but not on y or β. If this c is to be unbiased, then

$$\beta = E(c) = E((X'X)^{-1}X' + C)(X\beta + e) = \beta + CX\beta.$$

If this equation is to hold for all possible β, $CX = 0$ must hold. Now to find the variances of c, we first note that

$$c - \beta = ((X'X)^{-1}X' + C)(X\beta + e) - \beta$$
$$= ((X'X)^{-1}X' + C)e$$

since $CX = 0$. The matrix of variances and covariances of C is therefore

$$E((c - \beta)(c - \beta)') = ((X'X)^{-1}X' + C)E(ee')((X'X)^{-1}X' + C)'$$
$$= \sigma^2((X'X)^{-1}X' + C)((X'X)^{-1}X' + C)'$$
$$= \sigma^2((X'X)^{-1} + CC')$$

since $CX = 0$. Since the diagonals of CC' are the sums of squares, they must be positive and therefore the variances of c must be greater than those of b, the least squares estimate, which appear as the first term on the right in the last line.

Thus, under all of the assumptions we have made, the least-squares estimates are "blue". Note that for this theorem, we did not need to assume that the e have a normal distribution. But note also that we derived it by arguing that $CX\beta = 0$ for *all* β. If we have reason to believe that β satisfies some constraints then $CX\beta = 0$ would not have to hold for all β but only for those satisfying the constraints. In that case, therefore, more efficient estimates of β may found by imposing the constraints with, for example, G's *con* or *sma* commands.

This theorem has guided the development of methods to deal with cases in which E(ee') is not $\sigma^2 I$. These methods are special cases of Aitchen's Generalized Least Squares (GLS). We will explain the general idea here and two special cases in the following sections.

Let us suppose that the Datamaker assumptions hold except that E(ee') = $\boldsymbol{\Omega}$. The least squares estimates will then still be unbiased and consistent. They may not, however, be efficient. Can we find efficient estimates? If we know $\boldsymbol{\Omega}$, the answer is Yes, by use of what is called *generalized least squares (GLS)*, which we will now explain. To be a variance-covariance matrix, $\boldsymbol{\Omega}$ must be positive semidefinite. The principal axes theorem (see my *Matrix Methods in Economics*, page 117) then guarantees the existence of a matrix $\boldsymbol{V}$ such that $\boldsymbol{V'V} = \boldsymbol{VV'} = \boldsymbol{I}$ and $\boldsymbol{V'\Omega V} = \boldsymbol{D}$, where $\boldsymbol{D}$ is a non-negative diagonal matrix. We can then define another diagonal matrix $\boldsymbol{R}$ with diagonal elements $r_{ii} = 1/\sqrt{d_{ii}}$ where d_{ii} is the ith diagonal element of $\boldsymbol{D}$, so that $\boldsymbol{R'DR} = \boldsymbol{I}$. Let $\boldsymbol{B} = \boldsymbol{VR}$. If we now multiply

$$y = X\beta + e$$

on the left by $\boldsymbol{B'}$, we have

$$B'y = B'X\beta + B'e$$

and E($\boldsymbol{B'ee'B}$) = $\boldsymbol{B'\Omega B} = \boldsymbol{R'V'\Omega VR} = \boldsymbol{R'DR} = \boldsymbol{I}$. Thus, the OLS regression of the transformed y, $\boldsymbol{B'}y$, on the transformed X variables, $\boldsymbol{B'X}$, satisfies the conditions of the Gauss-Markov theorem and produces *efficient* estimates of β. The result of that regression will be

$$b^{GLS} = (X'BB'X)^{-1}X'BB'y = (X'\Omega^{-1}X)^{-1}X'\Omega^{-1}y.$$

The simplification given by the second of the = signs follows from the fact that

$$BB' = \Omega^{-1}$$

which comes from

$$V'\Omega V = D$$

by inverting both sides to get

$$V^{-1}\Omega^{-1}V^{-1} = D^{-1} = RR'$$

and then multiplying both sides on the left by $\boldsymbol{V}$ and on the right by $\boldsymbol{V'}$ to yield

$$\Omega^{-1} = VRR'V' = B'B.$$

Consequently, in computing we never need to apply the principal axes theorem and associated algorithms to find V. We just need the conceptually simpler Ω^{-1}.

The only problem with this panacea is that Ω is really never known. If the method is to be of any use, we have to make some assumptions that allow us to estimate it. In the next two sections, we will look at two such assumptions that may offer useful alternatives to ordinary least squares. The first relates to the case of a time series regression in which the error of one period is correlated with the error of the next. The second relates to systems of equations in which errors in different equations may be related.

Here we should note that the simplest such assumption is to suppose that Ω is diagonal with diagonals that vary in some way that can be estimated from the residuals of the regression. For example, with time series data, their square roots might be a linear function of time that can be estimated by regressing their absolute values on time. In such a case, the GLS estimate is found by simply dividing the dependent and independent variables of each observation by the standard deviation of the error term for that observation and then applying OLS to the resulting observations. In G , such a procedure is given by this series of commands for regression of y on x:

```
r y = x
f srrs = @sqrt(@sq(resid))
r srrs = time
f scale = predic
f yScaled = y/scale
f xScaled = x/scale
r yScaled = xScaled
```

You may ask for an example where this method has made estimates more plausible, and I will have to confess that I could not find one among the equations of QUEST or any other equation I could make up with the time series data that accompany this book. Generally, the procedure made no difference because there was little or no trend in the residuals. I believe that this version of GLS may be more applicable with cross-section data where the differences in size of observation may be much larger than they usually are with economic time series.

5. The Hildreth-Lu Technique for Autocorrelation of Residuals

If the value of RHO on G's regression display indicates that the "true" errors may be autocorrelated, then, as we have just seen, the least-squares estimates are not "efficient." Worse still, if the lagged value of the dependent variable is among the independent variables, then autocorrelation in the error terms means that the errors are correlated with at least one variable in the X matrix, so the least squares are not consistent. The Hildreth-Lu technique may be helpful in the face of such evidence of autocorrelation.

This technique begins from the assumption that the errors are autocorrelated by the first-order autocorrelation scheme

e(t) = ρe(t-1) + u(t)

where the u(t) are not autocorrelated. If we know ρ, there is a simple remedy. Let us write

y(t) = ßx(t) + e(t)

y(t-1) = ßx(t-1) + e(t-1).

and then multiply the second equation by ρ and subtract from the first to get

y(t) - ρy(t-1)   = ß(x(t) - ρx(t-1)) + e(t) - ρe(t-1)

= ß(x(t) - ρx(t-1)) + u(t).

Notice now that the error term is not autocorrelated, so OLS gives us efficient estimates of this equation.

Of course, we do not know ρ. The Cochrane-Orcutt suggestion was to use the ρ estimated from the OLS estimate. It may happen, however, that the very problems we are trying to circumvent cause the OLS estimate of ρ to be poor; then the method may be even worse than OLS. A better procedure was suggested by Hildreth and Lu: try a range of values of ρ and pick the "best" one. This is the method included in G. The general form of the Hildreth-Lu command is

hl <rho1> <rho2> <incr> <y> = <x1>, [x2,] [x3,] ...[xn]

Here rho1 is the starting guess of ρ, incr is the amount by which it is incremented on each iteration and rho2 is an upper limit on the guess. The y and x1, ..., xn are as in the r command. For example,

hl 0 1. .1 cR = gR, vR, feR, -fiR

will regress c$ - ρc$[1] on g$ - ρg$[1] and v$ - ρv$, first with ρ = 0, then with ρ = .1, and so on up to ρ = .9. A maximum of ten values of ρ will be tried on any invocation of the command. The results of each regression are displayed, and the assumed value of ρ is shown as RHO-HL on each display. Once an approximate range of interest for ρ has been identified, the equation can be rerun with a smaller value of incr. No more than 20 variables in all are presently permitted in the hl command in G.

At the end of the process, you will get a table with this heading:

RHO-HL   SEE 1-AHEAD   RHO-EST   SEE LONG-RUN

The RHO-HL shows the assumed ρ, the SEE 1-AHEAD shows the standard error of estimate (SEE) of the estimated equation (without using any further rho adjustment of the forecast), the RHO-EST shows the rho of the estimated equation, and SEE LONG-RUN shows the standard

error of using the fitted equation on the original, undifferenced data, without a knowledge of the true lagged value of the dependent variable, as must be done in forecasts of more than one period ahead.

If the "save" command is on for model building, all of the estimated equations will be placed in the ".sav" file as undifferenced equations suitable for going into a model. You must choose which one you want.

The above example, estimated by ordinary least squares, gives the following results

```
:                              Multiplier Estimates
  SEE   =       68.19 RSQ   = 0.9980 RHO =    0.78 Obser  =  105 from 1975.100
  SEE+1 =       42.31 RBSQ  = 0.9979 DW   =    0.43 DoFree =  100 to   2001.100
  MAPE  =        0.97
    Variable name            Reg-Coef  Mexval  Elas   NorRes     Mean   Beta
  0 gdpR                    - - - - - - - - - - - - - - -    6237.65  - - -
  1 intercept              -457.82310    11.5  -0.07   501.59     1.00
  2 vR                        1.40468    50.9   0.23    67.95  1026.15  0.250
  3 gR                        3.34416   300.2   0.65     7.79  1220.11  0.510
  4 feR                       2.35637    88.3   0.23     1.02   595.92  0.339
  5 -fiR                      0.33296     0.9  -0.04     1.00  -691.98  0.059
```

The hl command in the example gave the output summary table:

HL rho	SEE 1 ahead	Est. rho	SEE long
0.000000	68.185463	0.784687	68.185455
0.100000	62.975750	0.740949	68.189636
0.200000	58.092529	0.682434	68.208275
0.300000	53.619797	0.605829	68.257622
0.400000	49.660053	0.508860	68.372414
0.500000	46.329838	0.392330	68.634872
0.600000	43.744110	0.262790	69.280540
0.700000	41.972450	0.134028	71.159630
0.800000	40.928905	0.024735	78.298286
0.900000	39.859150	-0.023783	129.082184

In choosing which ρ to use, we need to look at everything in this summary table and at the regression coefficients. The first column in the table is simply the assumed value of ρ. Let us look first at the Rho-Est column. If the transformation did not eliminate autocorrelation in the transformed equation -- and sometimes it does not -- then the transformation was based on a false assumption about the structure of the error and may have made matters worse. The value of HL Rho which gives the Rho-Est closest to zero is of special interest; let us call it ρ^*. In our case, it lies in the interval [.8 , .9], and we can pin it down more closely with the command

        hl .8 .9 .01 gdpR = gR, vR, feR, -fiR

with the following results:

HL rho	SEE 1 ahead	Est. rho	SEE long
0.800000	40.928894	0.024736	78.298286
0.810000	40.849819	0.015745	79.840630
0.820000	40.771847	0.007264	81.688118
0.830000	40.693424	-0.000630	83.922455
0.840000	40.612625	-0.007836	86.654854
0.850000	40.526974	-0.014209	90.039986
0.860000	40.433220	-0.019560	94.297501
0.870000	40.326958	-0.023624	99.743355
0.880000	40.202152	-0.026041	106.840851
0.890000	40.050323	-0.026317	116.276627

From these results, we can, with sufficient accuracy, say that ρ^* is .83. As a first guess, it is the ρ we want.

Next, however, we should look at SEE 1 ahead, the standard error of the transformed equation. If this SEE 1 ahead reaches a minimum for ρ below ρ^*, we might prefer that lower ρ. In our example, however, SEE 1 ahead goes right on declining past ρ^*.

But it is important to look also at SEE long-run. It will generally be rising as HL rho is increased. If it rises sharply for values of Rho-HL lower than ρ^*, as it seems to me to be doing in the example, you may want to pick a value before the sharp rise. Otherwise, you would be making a substantial sacrifice of the equation's ability to fit the data when it does not have the actual lagged value of the dependent variable to fall back on.

The usual advice is simply to pick ρ^* as the value of the HL ρ, re-estimate the equation, and be done with it.

Following this advice, we would pick $\rho = .83$. I, however, would be reluctant to see the long-term performance of the equation so much worsened, with the SEE long rising from 68.2 to 83.9. I would be more interested in a value of perhaps $\rho = .6$, which would give some improvement in the one-period-ahead forecast, with a drop from 68.18 to 43.74 in the SEE 1 ahead and a rise of the SEE long run only from 68.18 to 69.28.

But how much better off would I really be in forecasting one period ahead? The one-period ahead forecast of the OLS equation with the usual, automatic rho-adjustment is 42.31 (not visible in the summary table but shown on the full printout). This is only very slightly worse than the 42.28 found for the rho-adjusted forecast of the equation estimated with a Hildreth-Lu ρ of .6 and not much worse than the 40.69 with the usually chosen Hildreth-Lu ρ of .83. Thus, the short-term forecasting ability of the equation has not been noticeably helped by the Hildreth-Lu procedure, while the long-term forecasting ability has been impaired, a little for $\rho = .6$, a lot for $\rho = .83$.

Next, we should look at the regression coefficients and ask if the coefficients have become any more sensible. The usual multiplier analysis gives equal weight to a dollar of any one of these demands. So, theoretically, all of the regression coefficients should be the same. Let us look at them for three values of the Hildreth-Lu ρ. We find:

HL-rho	0	.6	.83
1 intercept	-457.82310	-159.45291	-64.98312
2 vR	1.40468	1.26041	1.17343
3 gR	3.34416	3.37155	3.54343
4 feR	2.35637	2.24681	2.09337
5 -fiR	0.33296	0.15618	0.20951

The largest coefficient was made steadily larger; the three smallest all were even smaller with the Hildreth-Lu estimate. Thus, the coefficients do not become more reasonable with the use of the Hildreth-Lu procedure.

Finally, two plots should be made and studied before deciding to accept a Hildreth-Lu estimate in place of the OLS estimate. One plot shows the errors of the one-period-ahead forecast from both equations with the rho-adjustment technique of Part 1 Chapter 2 applied to the least-squares equation. The second plot shows the errors of the OLS prediction and the prediction with the Hildreth-Lu values of the parameters but without the lagged value of the dependent variable. This comparison shows how the two equations will do in historical simulation or long-term forecasting when the last actual lagged value of the dependent variable has faded into the remote past. The line marked by the + signs shows the OLS errors in both graphs. Here they are for our example, with the Hildreth-Lu lines computed with $\rho = .83$.
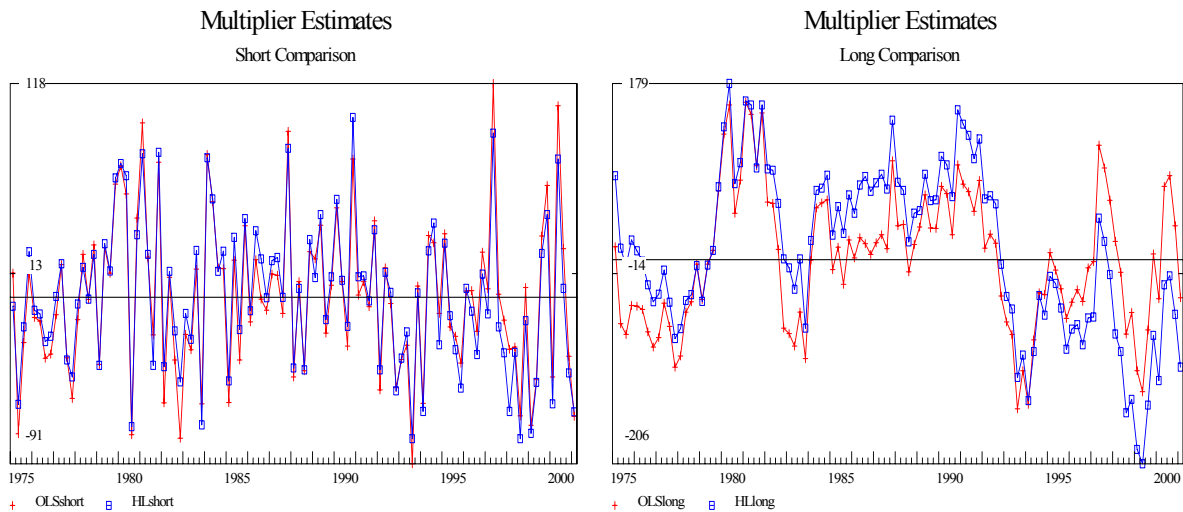
The least-squares fit is always better in the second graph; the question is by how wide a margin. If the margin is wide, and it sometimes is, I lose interest in the Hildreth-Lu estimates. In the present case, I find little difference.

The example is pretty typical of my own experience with the Hildreth-Lu technique. When one goes beyond the usual textbook advice, I have seldom found that I want to use it. My impression is that about ninety percent of the time, it makes little difference; you use it if you believe the Datamake fable and skip it if you don't. It is capable, however, of sometime seriously degrading the long-term forecasting ability of the equation and producing nonsensical regression coefficients. My advice is to never use the technique without examining the results carefully in the way shown in this section. Indiscriminate use is dangerous.

For reference, here is the file used to make all the calculations discussed in this section.

```
title Multiplier Estimates
add lim75
gdates 1975.1 2001.1
r gdpR = vR,gR,feR,-fiR
hl 0 1. .1 gdpR = vR,gR,feR,-fiR
hl .8 .9 .01 gdpR = vR,gR,feR,-fiR
hl .83 .83 .01 gdpR = vR,gR,feR,-fiR
gname hlshort
subti Short Comparison
f OLSshort = predp1 - depvar
f HLshort = hlshort - depvar
gr OLSshort HLshort
gname hllong
subti Long Comparison
f OLSlong = predic - depvar
f HLlong = hllong - depvar
gr OLSlong HLlong
```

EXERCISES

1.   Re-estimate all of the equations of the model in Chapter 8 with the Hildreth-Lu technique. Be sure to examine carefully the two plots for each equation. Which equations, if any, were definitely improved by the method? Were there any where you would definitely prefer the ordinary least squares?

2.   Rebuild and simulate the AMI model with the Hildreth-Lu estimates developed in exercise 1. (If there are some HL estimates that you really do not like, stick to the LS estimate for them.) Is the performance of the model improved? Run some policy experiments and make some forecasts with the two models. What differences do you note?

6. Stacked and Seemingly Unrelated Regression

Stacked regression allows us to impose constraints on regression coefficients across two or more related regressions. We can take as an example the estimation of the demand for food and the demand for gasoline, each as a function of its own price, the price of the other, and a "demand curve shifter" which is disposable income per capita in the case of food and an estimate of the stock of cars per capita in the case of gasoline. A theorem of microeconomics suggests that the

price of food should have the same coefficient in the equation for the demand for gasoline that the price of gasoline has in the equation for the demand for food. We can set up the estimation as follows:

> f lim 1970.1 2001.1
> f ypc$ = pidis$/pop
> f food = cfood$/pop
> f gasoline= cgaso$/pop
> f dc = c/c$
> f pfood = (cfood/cfood$)/dc
> f pgasoline = (cgaso/cgaso$)/dc
> f ub = @cum(ub,1.,.08)
> f cars1 = @cum(cars1,cdmv$,.08)/ub
> f cars2 = @cum(cars2,cars1,.08)/ub
> f carspc = (cars1+cars2)/pop
> title Demand for Food
> r food = ypc$, pfood, pgasoline
> title Demand for Gasoline and Oil
> r gasoline= carspc, pfood, pgasoline

The results are:

```
:                              Demand for Food
  SEE   =       37.03 RSQ   = 0.9718 RHO =   0.88 Obser   =  125 from 1970.100
  SEE+1 =       18.41 RBSQ  = 0.9711 DW  =   0.25 DoFree =  121 to   2001.100
  MAPE  =        1.09
    Variable name             Reg-Coef  Mexval  Elas    NorRes      Mean    Beta
  0 food                      - - - - - - - - - - - - - - -     2719.34  - - -
  1 intercept              1964.62501    59.4   0.72    35.52      1.00
  2 ypc$                      0.06547   318.0   0.43     1.19  18058.19   0.923
  3 pfood                  -374.85603     3.4  -0.14     1.02      1.04  -0.070
  4 pgasoline               -27.93018     1.2  -0.01     1.00      1.29  -0.033

:                         Demand for Gasoline and Oil
  SEE   =       12.13 RSQ   = 0.7194 RHO =   0.84 Obser   =  125 from 1970.100
  SEE+1 =        6.78 RBSQ  = 0.7124 DW  =   0.33 DoFree =  121 to   2001.100
  MAPE  =        2.15
    Variable name             Reg-Coef  Mexval  Elas    NorRes      Mean    Beta
  0 gasoline                  - - - - - - - - - - - - - - -      447.25  - - -
  1 intercept               140.15273     4.0   0.31     3.56      1.00
  2 carspc                    0.06003    58.2   0.20     1.60   1483.05   0.861
  3 pfood                   262.63407    15.4   0.61     1.56      1.04   0.475
  4 pgasoline               -43.56514    24.7  -0.13     1.00      1.29  -0.502
```

Stacked Regression

Clearly, the coefficient on the price of gasoline in the food equation, -27.93, is by no means equal to the coefficient of the price of food in the gasoline equation, 282.63. If we want to *impose* that equality, we "stack" the regressions as follows:

```
stack
  r food = ypc$, pfood, pgasoline
  r gasoline= carspc, pfood, pgasoline
  con 1  0 = a4 - b3
do
```

In the constraint command, an a refers to a coefficient in the first equation, a b refers to a coefficient in the second equation, and so on up to the number of equations in the stack. In the example, we are softly constraining the fourth coefficient in the first equation to be equal to the third coefficient in the second equation. Note that the constraint command must follow all of the "r" commands under the "stack" command. In effect, the "stack" command combines the regressions under it into one big regression and applies the constraint in this combined regression. The combined regression may be thought of as looking something like this:

$$\begin{pmatrix} y1 \\ y2 \end{pmatrix} = \begin{pmatrix} X1 & 0 \\ 0 & X2 \end{pmatrix} \begin{pmatrix} a \\ b \end{pmatrix} + \begin{pmatrix} r1 \\ r2 \end{pmatrix}$$

We have, in effect, "stacked" one regression on top of the other. Now the errors of the first equation may well have a different variance than those of the second. In the present example, the variance of the r1 is about four times as large as the variance of the r2. If the two were combined without taking account of that difference, most of the adjusting to accommodate the restraint would be done by the second equation. We can, however, easily get the variances to be of similar size by first estimating the individual equations separately, calculating the SEE of each equation separately, and then dividing both the independent and dependent variables of each equation by the SEE of that equation. If a regression is then done on these "normalized" variables, the SEE will be 1.0 for both equations.

 This is exactly what the "stack" command does. It first reports the individual equations, which are the same as shown above, and then reports the variances of each equation as the diagonal elements of a "Sigma Matrix", like this in our example:

    The Sigma Matrix
      1371.33198   0.00000
          0.00000 147.16666

    The Sigma Inverse Matrix
        0.0007 0.0000
        0.0000 0.0068

```
                             Demand for Gasoline and Oil
Regression number 1, food
  SEE    =      37.89 RSQ   = 0.9705 RHO =    0.89 Obser   =  250 from 1970.100
  SEE+1  =      18.12 RBSQ  = 0.9698 DW  =    0.22 DoFree  =  242 to   2001.100
  MAPE   =       1.13 SEESUR =  1.08
    Variable name              Reg-Coef  Mexval  Elas   NorRes      Mean    Beta
  0 food                       - - - - - - - - - - - - - - -     2719.34  - - -
  1 intercept               2088.13832    32.4   0.77    1.03        1.00
  2 ypc$                        0.06526   182.2   0.43    1.00    18058.19  0.920
  3 pfood                    -537.15829     3.1  -0.21    1.00        1.04 -0.101
  4 pgasoline                  10.66789     0.1   0.01    1.00        1.29  0.013


:                             Demand for Gasoline and Oil
Regression number 2, gasoline
  SEE    =      13.86 RSQ   = 0.6337 RHO =    0.88 Obser   =  250 from 1970.100
  SEE+1  =       6.92 RBSQ  = 0.6247 DW  =    0.24 DoFree  =  242 to   2001.100
  MAPE   =       2.53 SEESUR =  1.08
    Variable name              Reg-Coef  Mexval  Elas   NorRes      Mean    Beta
  5 gasoline                   - - - - - - - - - - - - - - -      447.25  - - -
  1 intercept                404.65497    64.1   0.90    1.95        1.00
  2 carspc                      0.04439    22.2   0.15    1.12     1483.05  0.637
  3 pfood                      10.66910     0.1   0.02    1.12        1.04  0.019
  4 pgasoline                 -26.65290     5.6  -0.08    1.00        1.29 -0.307
```

We can now see that the equality of a4 and b3 has been assured with little cost to SEE of either equation. Do the magnitudes of the price and "demand shifters" seem reasonable to you? The "SEESUR" measure which appears on these displays is the SEE of the combined, stacked regression. Without the constraint, it would be 1.00 because of the normalization.

Seemingly Unrelated Regression (SUR)

If we now think of the errors in the stacked regression, we realize that -- although the equations are "seemingly unrelated" -- there is one obvious possibility for correlation among the error terms. Namely the error in period t in one equation may be correlated with the error in period t in the other equation. Perhaps, whenever we spend more on gasoline than we "should" according to the equation, simultaneously spend less on food. If that is so, then the least squares estimates of the stacked system is not the "best", that is, it does not have minimum variance. It is, therefore, a candidate for being improved by application of generalized least squares.

To estimate $\mathbf{\Omega}$, we make the hypothesis that all the off-diagonal elements are zero except that $E(e_{it}e_{jt}) = \sigma_{ij}$, where e_{it} and e_{jt} are the errors in the ith and jth equations of the stacked system, that is, that contemporaneous cross correlations are not necessarily zero. The matrix of these contemporaneous cross correlations we will call $\mathbf{\Sigma}$. From its inverse we can easily construct Ω^{-1} and compute the GLS estimate. Because many elements of Ω are zero, there are shortcuts to making the calculations.

In G, the setup for applying GLS to this "seemingly unrelated regression" or SUR problem is as simple as the stack command. Here are the commands to estimate our previous example by SUR.

```
     sur
       r food = ypc$, pfood, pgasoline
```

```
            r gasoline= carspc, pfood, pgasoline
         do
```

G first estimates the equations independently, then prints out the estimate of the Σ matrix and its inverse based on the residuals from the separate regressions, like this:

```
    The Sigma Matrix
      0 1371.33198    65.65089
      1    65.65089  147.16666

    The Sigma Inverse Matrix
      0   0.0007  -0.0003
      1  -0.0003   0.0069

  Seemingly Unrelated Regression of Demand for Food and Gasoline
Regression number 1, food
   SEE   =       37.04 RSQ   = 0.9718 RHO =   0.88 Obser  =   250 from 1970.100
   SEE+1 =       18.34 RBSQ  = 0.9711 DW   =   0.24 DoFree =  242 to    2001.100
   MAPE  =        1.09 SEESUR =       1.00
     Variable name              Reg-Coef  Mexval  Elas   NorRes      Mean   Beta
   0 food                       - - - - - - - - - - - - - - - - -   2719.34  - - -
   1 intercept                1988.32281    33.8   0.73    1.02      1.00
   2 ypc$                        0.06514   202.8   0.43    1.00  18058.19  0.918
   3 pfood                    -392.19094     1.9  -0.15    1.00      1.04 -0.074
   4 pgasoline                 -27.71240     0.6  -0.01    1.00      1.29 -0.033

Regression number 2, gasoline
   SEE   =       12.13 RSQ   = 0.7194 RHO =   0.84 Obser  =   250 from 1970.100
   SEE+1 =        6.78 RBSQ  = 0.7124 DW   =   0.33 DoFree =  242 to    2001.100
   MAPE  =        2.15 SEESUR =       1.00
     Variable name              Reg-Coef  Mexval  Elas   NorRes      Mean   Beta
   5 gasoline                   - - - - - - - - - - - - - - - - -    447.25  - - -
   1 intercept                 141.83479     2.1   0.32    2.28      1.00
   2 carspc                      0.05979    32.2   0.20    1.30   1483.05  0.858
   3 pfood                     261.34901     7.9   0.61    1.28      1.04  0.472
   4 pgasoline                 -43.55446    13.0  -0.13    1.00      1.29 -0.502
```

Comparing these regression coefficients with the original ones shows that the effects of SUR are trifling. This outcome is fairly typical of my experience. In fact, if the independent variables are exactly the same in the two regressions, SUR has no effect. The real reason for using "carspc" instead of "ypc$" in the Gasoline equation was to show at least some slight effect of SUR.

Of course, we can now add the constraint to SUR like this

```
         title SUR for Food and Gasoline-- with cross-equation constraint
         sur
          r food = ypc$, pfood, pgasoline
          r gasoline= carspc, pfood, pgasoline
          con 1  0 = a4 - b3
         do
```

with these results:

```
:          Seemingly Unrelated Regression of Demand for Food and Gasoline
Regression number 1, food
  SEE   =       38.25 RSQ   = 0.9700 RHO =    0.89 Obser  =   250 from 1970.100
  SEE+1 =       17.88 RBSQ  = 0.9692 DW  =    0.22 DoFree =   242 to    2001.100
  MAPE  =        1.14 SEESUR =        1.08
    Variable name          Reg-Coef  Mexval  Elas   NorRes     Mean   Beta
  0 food                   - - - - - - - - - - - - - - -     2719.34  - - -
  1 intercept          2220.35697    38.0   0.82    1.02      1.00
  2 ypc$                   0.06425   180.7   0.43    1.00   18058.19  0.906
  3 pfood               -653.05646     4.9  -0.25    1.00      1.04 -0.122
  4 pgasoline             16.05632     0.2   0.01    1.00      1.29  0.019
:          Seemingly Unrelated Regression of Demand for Food and Gasoline
Regression number 2, gasoline
  SEE   =       13.79 RSQ   = 0.6372 RHO =    0.88 Obser  =   250 from 1970.100
  SEE+1 =        6.92 RBSQ  = 0.6282 DW  =    0.24 DoFree =   242 to    2001.100
  MAPE  =        2.53 SEESUR =        1.08
    Variable name          Reg-Coef  Mexval  Elas   NorRes     Mean   Beta
  5 gasoline              - - - - - - - - - - - - - - -      447.25  - - -
  1 intercept           397.14551    60.7   0.89    1.95      1.00
  2 carspc                0.04495    22.5   0.15    1.12    1483.05  0.645
  3 pfood                16.05747     0.2   0.04    1.12      1.04  0.029
  4 pgasoline           -25.83077     5.7  -0.07    1.00      1.29 -0.298
```

The results for some of the coefficients are noticeably different from the stacked without SUR. The price interaction, in particular, is stronger. Which is more plausible is hard to say.

G will accommodate up to ten regressions under a "stack" or "sur" command.

We notice that the essential idea for practical application of GLS was some notion of the structure of Ω. The assumption that the errors have the sort of autocorrelation for which we applied the Hildreth-Lu method leads to a structure for Ω; and in this case GLS can be shown to be almost exactly the same as Hildreth-Lu.

SUR should be used only if you are a firm believer in the fable about a true equation. Otherwise, it may give you parameters which will suit your purposes far less well than do the parameters given by ordinary least squares. Let me try to explain why that is so. Any generalized least squares method amounts to minimizing not r'r but $r'\Omega^{-1} r$, where r is the vector of residuals. Let us consider a system of two stacked equations. Let us suppose that

$$\Sigma \; = \; (1/3)\begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix} \text{ and } \Sigma^{-1} \; = \; \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix}.$$

Then, for three observations on each equation, the Ω^{-1} matrix is shown in the first six columns of the panel below.

		Ω^{-1}				A	B	C
2	0	0	-1	0	0	2	2	1.2
0	2	0	0	-1	0	-1	-1	-0.6
0	0	2	0	0	-1	-1	-1	-0.6
-1	0	0	2	0	0	2	-2	-1.2
0	-1	0	0	2	0	-1	1	0.6
0	0	-1	0	0	2	-1	1	0.6

Consider now three alternative estimates of the parameters. Estimate A gives the residuals shown in the column labeled A, while estimates B and C give the residuals shown in the columns labeled B and C. The value of r'Ω^{-1}r for estimate A is 12; for estimate B, 36; and for estimate C, 12.96. The SUR criterion will pick estimate A. Are you sure you want that estimate? Estimate C gives residuals which are forty percent lower in absolute value for every observation. Furthermore, the residuals in estimates B and C cancel out in each period; in the first period, for example, the first equation has a miss of +1.2 while the second equation has a miss of -1.2. SUR likes estimate A because the residuals follow the expected pattern of a positive correlation between the errors of the two equations. OLS is indifferent between A and B but strongly prefers C to either of them.

In most examples I can think of, I also would prefer estimate C. Suppose, for example, that the two equations are for (1) investment in equipment and (2) investment in structures. Suppose that I make a forecast using estimate A; and, sure enough, it turns out that both equations overestimate by 2, so that total investment is over-predicted by 4. Am I going to be happier than if one had been over-predicted by 1.2 while the other was under-predicted by 1.2, so that the errors exactly canceled out? Am I going to be consoled by the thought that these equations always tend to make errors that compound rather than cancel? Not I. How about you?

Of course, if you really believe the fable about there being a true equation of exactly the form you are estimating, then you will believe that SUR gives you more efficient estimates of the true parameters. But if you recognize that all you are trying to do is to get an equation which gives you a crude approximation of the way the economy works, then you may very well want to avoid SUR and all GLS procedures.

Comment on Maximum Likelihood Methods

Generalized least squares is a special case of a family of methods known as "maximum likelihood" methods. These methods all amount to expressing, as a function of the parameters of the equation, the probability that the sample should have occurred and then choosing the parameters to maximize that probability. What the method really amounts to depends on the assumption about the form of the probability function, and that often amounts chiefly to assumptions about the Ω matrix. If one assumes that $\Omega = \sigma^2 I$, then the ordinary least squares estimates are maximum likelihood estimates. But if one allows more and more elements of Ω to be unknown and determined by the maximizing process, the results can be very like choosing

alternative A in the above example. In practice, that amounts to tolerating habitual mistakes while avoiding unusual ones. Mathematical statisticians assure us that under fairly general assumptions maximum likelihood estimates have desirable large sample properties, namely, they are both consistent and asymptotically efficient. These properties, however, are only as good as the Datamaker assumption. As soon as we admit that we know perfectly well that we are not estimating the true model and that we don't even know all the relevant variables and are just looking for reasonable, workable approximations, the comfort from these theorems vanishes.

Some people say that they find maximum likelihood "an intuitively appealing" method.   I expect this appeal is because we find it difficult to think about joint probabilities. Our thinking gravitates to the case of independent, identically distributed errors, and there maximum likelihood is the same as ordinary least squares. When one holds clearly in mind what maximum likelihood does when there are significant dependencies among the errors of the equations, the method becomes very unappealing, at least to me.

7. Equations with Moving Average Error Terms

Autocorrelation of the residuals may be caused by other structures than the one used for the Hildreth-Lu technique. For example, we could imagine that the error term is a *moving average* of independent error terms, thus

(1) $e(t) = u(t) + h_1u(t-1) + h_2u(t-1),$

where the u's are independent random variables. Notice that in this case the errors are autocorrelated but not in the way assumed by the Hildreth-Lu procedure. Applying Hildreth-Lu is likely to make the situation worse. This assumption about the error term is fundamental to a technique that was popularized by G. E. P Box and G. M. Jenkins.[2] In this line of literature, the independent variables are often simply the lagged values of the dependent variable, so the method is often referred to as ARMA (AutoRegressive Moving Average) or ARIMA (AutoRegressive Integrated Moving Average) if differencing has been used to produce a stationary series.   The application of the technique to economic data has been nicely discussed in the textbook by Pindyck and Rubinfeld.[3] The general form of the equation used in ARMA analysis is

(2) $y(t) = b_1x_1(t) + b_2x_2(t) + ... + b_px_p(t) + u(t) + h_1u(t-1) + ... + h_qu(t-q)$

---

[2]G. E. P Box and G. M. Jenkins in *Time Series Analysis* (San Francisco, Holden-Day, 1970)

[3] Robert S. Pindyck and Daniel L. Rubinfeld , *Econometric Models and Economic Forecasts*, New York, McGraw-Hill, 1976, 1981,1991.

where the:

    $x(t)$ are observable variables which may or may not be lagged values of the dependent
        variable
    b are matching constants to be estimated.
    $u(t)$ is an unobservable random variable with an unchanging distribution, often assumed
        to be normal, with each observation independent of all previous ones.
    $h_1, ..., h_q$ are constants to be estimated.

If $q = 0$, the b can be estimated by ordinary least squares.   The special problem arises if $q > 0$, for the $u(t)$ are unobservable. We will see how that inconvenience is overcome.

The technique became enormously popular in the 1970's and 1980's because almost any time series could be forecasted one or two periods ahead with some accuracy and almost no work by using only lagged values of the dependent variable as x variables. Thousands of papers were written and probably millions of forecasts made with the technique. Conferences on forecasting were completely dominated by its practitioners. The main questions were how to decide how many lagged values of the dependent variable should be used and how many lags of $u(t)$ were needed. Needless to say, answers to these questions added little to our understanding of how the economy works.

Nonetheless, these techniques have a place in the toolkit of a structural model builder. In the first place, in any forecast with a structural model the errors of the equations must also be forecast, either explicitly or implicitly. If nothing is done about them, the implicit forecast is that they are zero. The use of rho adjustment in models built with G makes a simple autoregressive forecast of the errors. But the question naturally arises as to whether better forecasts of the errors could be made if more care were devoted to them. This is a natural problem for ARMA techniques.

Another use arises in updating data. In the U.S. National Income and Product Accounts, the first release for each new quarter has no data for corporate profits or other series dependent on them. Yet QUEST must have data on all these series in order to start from the new quarter. Making up these one-quarter-ahead forecasts is a possible application of ARMA techniques. Annual models with industry detail are often used in November or December for forecasting the year ahead. At that season, perhaps nine months of data has been accumulated on the current year. To get the annual number for the current year in such a series, we need a forecast of just the next three months. ARMA methods can be usefully applied to this problem.

Because of these ancillary uses with structural models, as well as to understand what is in the snake oil bottles sold by many forecasters, you need ARMA in your bag of tricks. We will explain how the estimation is done in G and, of course, the commands for using it.

As already observed, if $q = 0$, the b can be estimated by ordinary least squares. The special problem arises if $q > 0$, for the $u(t)$ are unobservable. If we make a guess of the b and h vectors,

and assume that the u(t) were zero before the beginning of the period of fitting, then we can recursively *calculate* the u(t). The idea of the fitting process is then to choose the b and h vectors to minimize the sum of squares of these calculated *u(t)*. The problem in doing so, however, is that these *u(t)* are highly non-linear functions of the *h* and *b* vectors. Many books about time series leave the matter there with the comment that the programs take care of the minor detail of how the equation is estimated.

As with least squares, however, I want you to understand what the computer is doing when using G, so that you realize how utterly mechanical the process is and do not suppose that any special trust should be placed in the results. In the case of moving average errors, different programs may give different results, so it may be important to know the limits of the method used.

The process used in G is iterative. It begins by assuming that the *h* vector is zero and uses ordinary least squares to compute an initial estimate of the *b* vector. Then it computes approximate values of the partial derivatives of the predicted values with respect to each element of *b* and *h* and regresses the current estimate of *u(t)* on these partial derivatives. (We'll see how these partials are computed in a moment; therein lies the trick.) The resulting regression coefficients are added to the current estimates of *b* and *h* and, if all goes well, the process is repeated until convergence.   "If all goes well" is said for good reason. There is no guarantee that the process will converge or even that each step will reduce the sum of squared errors.   In G, however, if the full step does not reduce the sum of squared errors, a step of half that size is tried, and if that does not work, a step of one quarter the size is tried, and so on down to one 64th. If even that tiny step does not help, the process stops and prints the message "Iterations XX. Cornered." where XX is the number of iterations completed, counting the one that could not be completed. If, on the other hand, the convergence criterion (that no element of *h* should change by more than .001) is met, the message is "Iterations XX. Converged."

This process cannot be guaranteed to produce the global minimum sum of squares, but it produces a value that is no worse than the initial estimate and sometimes much better. Also, without elaborate checking, it avoids explosive equations.

It remains to explain how to approximate the partial derivatives of the predicted values with respect to each element of *b* and *h*. To motivate the method, it is convenient to use the lag operator, L, defined by the equation

$$Lz(t) = z(t-1)$$

for any time-series variable, z(t). Powers of L work as expected:

$$L^2z(t) = L(Lz(t)) = L(z(t-1)) = z(t-2)$$

and so on for higher powers. In this notation, the general form of the equation we are trying to estimate is

(3) $y(t) = b_1 x_1(t) + ... + b_p x_p(t) + (1 + h_1 L + h_2 L^2 + ... + h_q L^q) u(t)$

Since $u(t) = y(t) - p(t)$, where $p(t)$ is the predicted value, the partial derivatives of the predicted values are the *negatives* of the partial derivatives of u(t) with respect to *b* and *h*. We can write *u(t)* as

(4) $u(t) = (1 + h_1 L + h_2 L^2 + ... + h_q L^q)^{-1}(y(t) - (b_1 x_1(t) + ... + b_p x_p(t)))$.

The negative of the partial derivative of *u(t)* with respect to b_j is then

(5) $z_j(t) = (1 + h_1 L + h_2 L^2 + ... + h_q L^q)^{-1} x_j(t)$

and the negative of the partial of u(t) with respect to h_j

(6) $z_{q+j}(t) = (1 + h_1 L + h_2 L^2 + ... + h_q L^q)^{-2} L^j (y(t) - (b_1 x_1(t) + ... + b_p x_p(t)))$.

To solve (5) for $z_j(t)$, we rewrite it as

(7) $(1 + h_1 L + h_2 L^2 + ... + h_q L^q) z_j(t) = x_j(t)$

or

(8) $z_j(t) = - h_1 z_j(t-1) - h_2 z_j(t-2) - ... - h_q z_j(t-q) + x_j(t)$

We start off with the approximation that $z_j(t) = 0$ for $t < 0$. Then we can recursively compute the values for all the more recent values of *t*. Similarly, for the partials with respect to the *h*'s,

(9) $(1 + h_1 L + h_2 L^2 + ... + h_q L^q)^2 z_{q+j}(t) = L^j (y(t) - (b_1 x_1(t) + ... + b_p x_p(t)))$.

Let us define the elements of a vector *g* by

(10) $(1 + h_1 L + h_2 L^2 + ... + h_q L^q)^2 = 1 + g_1 L + g_2 L^2 + ... + g_{2q} L^{2q}$

Then (9) can be written as

(11) $z_{q+j}(t) = - g_1 z_{q+j}(t-1) - g_2 z_{q+j}(t-2) - ... - g_{2q} z_j(t-2q) + e(t-j)$,

where *e(t)* is the residual using only the *x* variables with the current values of the *b* parameters. If we begin with the approximation that the values of all variables in equation (11) are zero before $t = 0$, we can then solve the equation recursively for the values of $z_{q+j}(t)$.

Such is the theory of the estimation. The practice is much easier. The command is just

      bj <q> <y> = <x1>, [x2,] ..., [xn]

where q is the order of the moving average error. For example,

```
bj 3 d = d[1],d[2],d[3]
```

The ! to suppress the intercept also works, thus

```
bj 3 d = ! d[1],d[2],d[3]
```

The command takes its name from Box and Jenkins, authors of the book cited above. Here is an almost classical example applied to annualized quarter-to-quarter rates of growth of U.S. real GDP. Because the method is so frequently used with lagged values of the dependent variable as the only independent variable, we will take first such an example. The G commands to set up the problem are just

```
ti BJ Demo: Real GDP Growth
lim 1970.1 2001.3 2005.4
mode f
f lgdpR = @log(gdpR)
f d = 400.*(lgdpR - lgdpR[1])
bj 3 d = d[1],d[2],d[3]
vr -12 -8 -4 0 4 8 12 16
gname bj1
gr *
```

And here are the results.

BJ Demo: Real GDP Growth
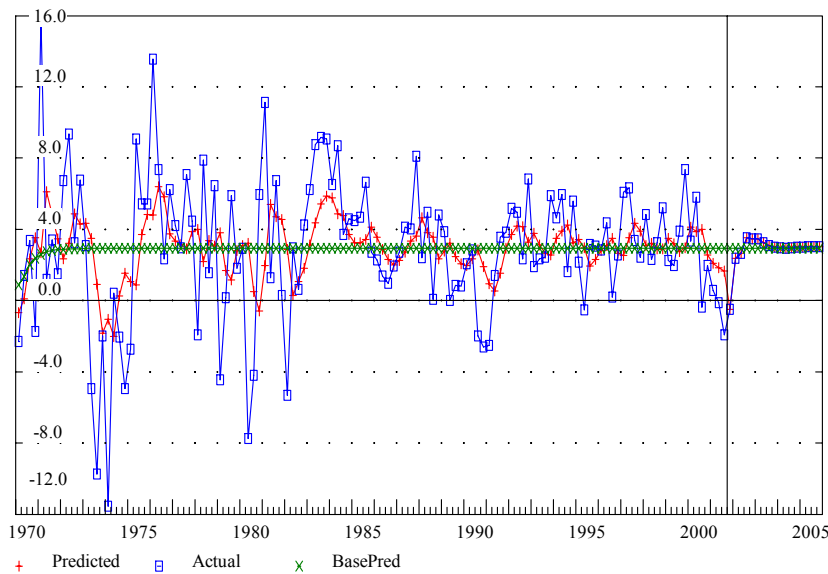
```
:                              BJ Demo: Real GDP Growth
  SEE   =        3.83 RSQ   = 0.1266 RHO =   -0.00 Obser  =  127 from 1970.100
  SEE+1 =        3.83 RBSQ  = 0.1053 DurH = 999.00 DoFree =  123 to   2001.300
  MAPE  =     1930.84
     Variable name          Reg-Coef  Mexval  Elas   NorRes    Mean   Beta
  0 d                       - - - - - - - - - - - - - - - -   2.92 - - -
  1 intercept                1.62239    4.5   0.56    1.14     1.00
  2 d[1]                     0.25565    3.2   0.25    1.03     2.91  0.256
  3 d[2]                     0.16613    1.3   0.17    1.00     2.92  0.167
  4 d[3]                     0.02402    0.0   0.02    1.00     2.91  0.024

Iterations 3. Cornered.

:                              BJ Demo: Real GDP Growth
  SEE   =        3.80 RSQ   = 0.1896 RHO =   -0.00 Obser  =  127 from 1970.100
  SEE+1 =        0.00 RBSQ  = 0.1491 DurH = 999.00 DoFree =  120 to   2001.300
  MAPE  =      153.64
     Variable name          Reg-Coef  Mexval  Elas   NorRes    Mean   Beta
  0 d                       - - - - - - - - - - - - - - - -   3.93 - - -
  1 intercept                1.38956    0.5   0.47    1.24     1.34
  2 d[1]                     0.66263    0.6   0.66    1.10     3.89  0.809
  3 d[2]                     0.10811    0.0   0.11    1.09     3.90  0.132
  4 d[3]                    -0.23370    0.2  -0.23    1.08     3.90 -0.285
  5 d_mu[1]                 -0.41293    0.3   0.00    1.02    -0.03 -0.548
  6 d_mu[2]                 -0.03871    0.0  -0.00    1.01     0.00 -0.051
  7 d_mu[3]                  0.20081    0.1   0.00    1.00     0.05  0.264
```

The first table shows the regression without the moving average errors. The second shows the results with them. The calculated *u(t)* variable is given the name of the dependent variable plus the suffix _mu. It is entered into the G workspace bank with this name, and the *h* parameters are shown as the regression coefficients of its lagged values in the second table, while the estimates of the *b* are shown as the regression coefficients of the usual independent variables in this second table. Other statistics for particular variables in the second table are derived from the last regression and may not have much meaning. In the graph, G detects as usual the presence of lagged dependent variables and calculates a third line, BasePred, the predictions the equation would have made using as lagged values of the dependent variable the equation's own prediction.

These results are fairly typical of my experience with the method. The SEE dropped ever so slightly from 3.83 to 3.80. The regression coefficients jumped around gaily; BasePred very quickly goes to the average value; the forecast also very quickly converges to the historical average.

Here is another example with errors from the equipment investment equation in QUEST. For a comparison, we take the automatic rho-adjustment forecast. The commands used were

```
add vfnreR.reg
f  e = resid
ti Error from Gross Equipment Investment
lim 1980.1 2001.3 2005.4
mode f
bj 2 e = ! e[1],e[2],e[3]
f  fancy = depvar
```

```
# For comparison
r e = ! e[1]
f  plain = predic
ti Error Forecasts for Equipment Investment
vr -10 -5 0
gr plain fancy 2001.4 2005.4
```

The original investment regression began in 1975, so the error series begins in that year. To have historical data on lagged values of e, the regression period was then shortened. Here are the results for the bj command and the comparison of the forecasts.

```
:                      Error from Gross Equipment Investment
  SEE    =       9.70 RSQ   = 0.6496 RHO =   0.01 Obser  =   87 from 1980.100
  SEE+1  =       9.71 RBSQ  = 0.6413 DurH = 999.0 DoFree =   84 to    2001.300
  MAPE   =     181.74
    Variable name             Reg-Coef  Mexval  Elas   NorRes     Mean   Beta
  0 e                         - - - - - - - - - - - - - - - -    -1.31  - - -
  1 e[1]                        0.64455   18.8   0.49    1.05    -0.99
  2 e[2]                        0.25956    2.5   0.13    1.01    -0.68  0.264
  3 e[3]                       -0.08432    0.4  -0.04    1.00    -0.63 -0.086

Iterations = 3. Cornered.

:                      Error from Gross Equipment Investment
  SEE    =       9.70 RSQ   = 0.7247 RHO =   0.01 Obser  =   87 from 1980.100
  SEE+1  =       0.00 RBSQ  = 0.7113 DurH = 999.0 DoFree =   82 to    2001.300
  MAPE   =     326.63
    Variable name             Reg-Coef  Mexval  Elas   NorRes     Mean   Beta
  0 e                         - - - - - - - - - - - - - - - -    -1.49  - - -
  1 e[1]                        0.78320    0.0   0.66    1.05    -1.26
  2 e[2]                        0.22574    0.0   0.14    1.01    -0.90  0.245
  3 e[3]                       -0.15725    0.0  -0.09    1.01    -0.84 -0.171
  4 e_mu[1]                    -0.13894    0.0  -0.06    1.01    -0.65 -0.076
  5 e_mu[2]                    -0.06913    0.0  -0.03    1.00    -0.55 -0.038

:                      Gross Equipment Investment and Replacement
  SEE    =       9.96 RSQ   = 0.6311 RHO =  -0.15 Obser  =   87 from 1980.100
  SEE+1  =       9.83 RBSQ  = 0.6311 DurH = -1.74 DoFree =   86 to    2001.300
  MAPE   =     194.89
    Variable name             Reg-Coef  Mexval  Elas   NorRes     Mean   Beta
  0 e                         - - - - - - - - - - - - - - - -    -1.31  - - -
  1 e[1]                        0.79307   65.2   0.60    1.00    -0.99
```

Error Forecasts for Equipment Investment

plain   fancy2
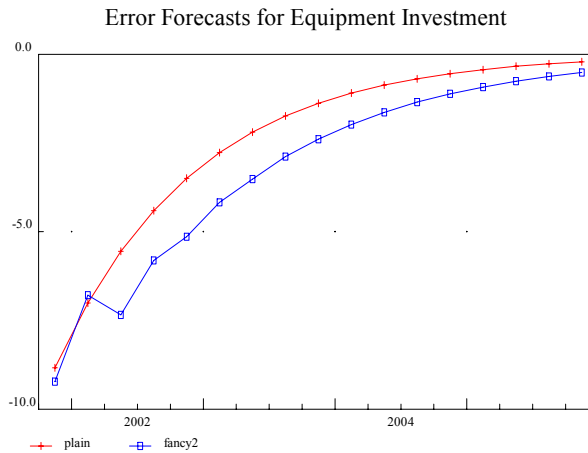
When tested on real data, it is always possible, of course, that a lack-luster performance of the method is due simply to the fact that it is inappropriate for the problem at hand. It is therefore interesting to fabricate an example where we know that the model is appropriate and see if the estimating method gets the right answer. Here is such a test.

```
ti Fabricated MA
lim 1970.1 2001.3
f one = 1
f ep = @normal()
f ep2 = ep + 2.*ep[1] + ep[2]
f dep = rtb + ep2
bj 2 dep = ! rtb
gr *
```

The @normal() function returns a random normal deviate with mean 0 and variance 1. Given the way the dependent variable is made, one would hope for $h_1 = 2$ and $h_2 = 1$. Of course, every time you run this command file, you get a different answer because of the random nature of the *ep* variable.

96

Fabricated MA

+ Predicted   □ Actual

```
:                     Fabricated MA Residual and Original Error
  SEE   =        2.50 RSQ   = 0.5971 RHO =    0.61 Obser   =   127 from 1970.100
  SEE+1 =        2.00 RBSQ  = 0.5971 DW   =    0.78 DoFree =   126 to    2001.300
  MAPE  =       83.99
    Variable name              Reg-Coef  Mexval  Elas   NorRes      Mean    Beta
  0 dep                        - - - - - - - - - - - - - - - - -    6.54  - - -
  1 rtb                         1.02251   205.4   1.03    1.00      6.56

Interations = 51. Converged.

:                     Fabricated MA Residual and Original Error
  SEE   =        1.18 RSQ   = 1.0000 RHO =    0.61 Obser   =   127 from 1970.100
  SEE+1 =        0.00 RBSQ  = 1.0000 DW   =    2.14 DoFree =   124 to    2001.300
  MAPE  =       27.66
    Variable name              Reg-Coef  Mexval  Elas   NorRes      Mean    Beta
  0 dep                        - - - - - - - - - - - - - - - - -   -1.90  - - -
  1 rtb                         1.06490   398.1  -0.96 9999.99      1.71
  2 dep_mu[1]                   1.91765 20063.6   2.97 9999.99     -2.94  1.922
  3 dep_mu[2]                   0.94776  9875.5  -1.08    1.00      2.17  0.947
```

The method ran to convergence, the estimates of the *h* are in the right general neighborhood but are both a little low, while the SEE is a little high. In repeated runs of this test, each time with different random errors, the coefficient on rtb fluctuates a little around the "true" value of 1.0; but the estimates of *h* are consistently below their "true" values of 2 and 1.   In this special, fabricated example, the reduction in the SEE is quite substantial. Moreover, as shown in the graph below of the original errors, *ep*, and the computed *u(t)*, the process has been fairly successful in figuring out what the *ep* were. In repeated runs of this example, it always runs to convergence and the estimates of the elements of *h* are *always* low. Though I am convinced that the method is working as intended, the estimates of the *h* appear to me to be biased a bit towards zero. Tightening the convergence test reduced this problem, so it may have something to do with starting from *h* = 0.

97

Fabricated MA Residual and Original Error

The Autocorrelation Function

Regression with moving average error terms is in its element when the observable variables are all lagged values of the dependent variable. The sources cited above give much attention to how to choose p and q in this case. A key tool in this connection is the autocorrelation function, which simply shows the correlation between the current and earlier values of a stationary variable. Once it has been computed, it is easy to compute, via relations known as the Yule-Walker equations, the approximations of the regression coefficients when regressing the current value on one lagged value, on two lagged values, on three lagged values, and so on.

In G the command is

    ac <series> [n]

where n is the number of correlations to be computed. The computation is done on the series between the first and second dates on the last *limits* command. The default value of n is 11. The series should be stationary. For example:

    ac viR 12

This command gives these results

```
Autocorrelation function
 1.0000  0.5250  0.3137  0.2170 -0.0048 -0.0909 -0.0375 -0.0911 -0.1610 -0.0329  0.0138  0.0055
Partial Autocorrelation
 0.525
 0.497  0.053
 0.495  0.030  0.046
 0.504  0.036  0.146 -0.202
 0.494  0.043  0.148 -0.176 -0.052
 0.498  0.058  0.136 -0.180 -0.093  0.084
 0.503  0.052  0.125 -0.171 -0.090  0.115 -0.062
```

98

```
0.495  0.067  0.113 -0.193 -0.074  0.121  0.002 -0.127
0.510  0.067  0.099 -0.184 -0.051  0.108 -0.006 -0.185  0.118
0.504  0.077  0.099 -0.190 -0.048  0.118 -0.011 -0.189  0.089  0.056
0.505  0.078  0.097 -0.191 -0.047  0.118 -0.013 -0.188  0.090  0.062 -0.012
0.503  0.084  0.106 -0.210 -0.048  0.130 -0.018 -0.208  0.100  0.070  0.041 -0.104
```

The "triangle" shows the regression coefficients as calculated from the autocorrelation function via the Yule-Walker equations. The first line shows (approximately) the regression coefficients on one lagged value; the second, on two; and so on. (Actually doing the regressions will give slightly different results because, in any finite series, the correlation between, say, x(t-3) and x(t-4) -- calculated by the regression program -- will be slightly different from that between x(t) and x(t-1) -- which is used in its place by the Yule-Walker equations.)

The numbers down the diagonal of the triangle are know as the partial autocorrelation function. The command also puts the autocorrelation function into the work space with a name given by the variable with _ac suffixed. It can then be graphed. Since dates have no meaning for it, the absolute positions in the series (indicated by numbers following a colon) are used to specify the range to be graphed. In our case, the command would be:

gr viR_ac : 1 11

If a process is a pure autoregression, with no moving average terms, the autocorrelation function is a declining exponential. If, on the other hand, it is a pure moving average process of order q, its autocorrelation function will fall to zero after q terms. In real data, of course, either of these extreme outcomes is very rare. In practice, you will probably just try a number of alternative specifications. You can easily write a G command file with an argument (the variable name) which will compute a handful of equations for you; you have only to pick the best. Some programs will do even that for you, but it seems to me important to look over the candidates.

8. The Classical Econometrics of Simultaneous Equations

Identification

In the early days of econometrics, before the computer made computation of regressions with more than two or three independent variables feasible, models were small; but a lot of thought went into theoretical problems of estimation. One of these was *identification,* a problem most simply illustrated by estimating the supply and demand curves for a product. The quantity demanded is a function of price, so we might suppose that if we estimated

(1) $q_t = a_1 + a_2 p_t$

we would have a demand function. But the quantity supplied is also a function of the price, so maybe what we got when we estimated (1) was the supply curve! In fact, what we estimate might be any combination of the demand and supply curve. If we wish to *identify* which curve we are estimating, to find the economic structure, we need more information. For example, if we know

that the supply of the product we are studying is a function of rainfall, r_t , so that the quantity supplied is

(2) $\qquad q_t = b_1 + b_2 p_t + b_3 r_t$

then we have a chance of identifying a demand curve of the form of equation (1). Graphically, we can imagine the demand curve fixed and stable while the supply curve jumps about depending on rainfall. The observed price-quantity combinations thus all fall on the stable demand curve, which becomes identified.

Note that the identification was possible because of the *exclusion* of the rainfall variable from the demand equation. Careful analysis of this situation led to the result that an equation in a system of N equations is identified if it excludes N-1 or more of the variables in the system. [4] If exactly N-1 variables are excluded, the equation is said to be *exactly identified*; if more than N-1 are excluded, the equation is said to be *over identified*. In these counts, N is the number of regression equations; each lagged value of an endogenous variable and all exogenous variables count as exogenous. Since most of these variables are excluded from any one regression, current models of the economy such as Quest and other models you are likely to build are vastly over-identified. Furthermore, a known value of a parameter in a regression equation is as good as an exclusion for the counts. The subject has therefore become one of more pedagogical and historical interest than of practical importance.

Estimation

We have already noted that if one of the independent variables in a regression actually depends, through other equations on the dependent one, least squares estimates may be inconsistent. For example, if in one equation consumption depends upon income but via another equation income is consumption plus investment and government expenditures, then there is danger of inconsistency, which may be called *simultaneous equation bias*.   In the early days of econometrics, the 1940's and 1950's, this problem was considered central, and a number of techniques were developed. All of them are, in my opinion, vastly inferior to the dynamic optimization which we have already studied and which solves simultaneous equation bias as a sort of minor side benefit. Nevertheless, a few words about these older techniques are perhaps in order just so you will know what they are.

In estimating QUEST, we used an *instrumental variable* approach to this problem for estimating the consumption function. We wanted consumption to depend on disposable income, but disposable income depends, via the accounting identities, on consumption. So we regressed

---

[4]See G. S. Maddala, *Econometrics* (McGraw-Hill, New York, 1977) pp. 471 - 477. It is indicative of the decline in the importance of the topic that this clearly written appendix was dropped from the second edition of the book.

disposable income on its own lagged values and used the predicted value as current period disposable income in the consumption function.

Following this approach to its logical conclusion leads to the method of two-stage least squares, 2SLS for short. In the first stage, each endogenous independent variable is regressed on all of the exogenous variables in the model. The predicted values are then used as the simultaneous values for all endogenous variables in a second "stage" regression. The predicted values, depending only on exogenous variables, certainly do not depend on the error in the equation being estimated. Hence, the cause of simultaneous equation bias has been removed.

This method can be applied in G. If the variable only occurs without lag or if you want to use the first stage also for lagged values, the procedure is simple. Recall that after each regression the predicted values are in the workspace under the name "predic". After each first-stage regression, use an f command to copy this "predic" to a variable having the name of the dependent variable of the preceding regression. Thus if "yRpc" were the dependent variable in the first stage regression, then we should follow the equation with

        f yRpc = predic

We then just re-estimate the equation.

If, however, we have both yRpc and yRpc[1] in the equation and we want to use the first stage estimate for the first but the actual value in the lagged position, then we have to go to a little more trouble. When all of these first stage regressions have been done, we copy the workspace to a new bank and then assign this bank as B. The commands are
        dos copy ws.* first.*
        bank first B
        zap

The *zap* gives us a clean workspace. Then in the regression commands where we want to use the first stage estimate, we prefix a "b." to the name of variable. (The b. will not appear in the .sav file, so it will work right in building a model.)

There are several problems with this procedure. The first is that in models of any size there are enough exogenous variables to give an almost perfect fit in the first stage so that the second stage differs insignificantly from the OLS estimate. It is not unusual for the number of exogenous variables to equal or exceed the number of observations used in fitting the equations. The first stage fit is then perfect and the second stage is identical to OLS. Various arbitrary rules are used to cut off the number of regressors in the first stage to get some difference between OLS and 2SLS, but these differences are then just as arbitrary as the cutoff rules.

A second problem with textbook 2SLS is that it assumes linearity in the model. Without linearity, it is not correct to suppose that the endogenous variables are linear functions of the exogenous ones. The suggestion is then sometimes made to use squares and cross products of all of the

exogenous variables. This procedure, however, will exacerbate the first problem of too many exogenous variables. It also does not insure that the right kind of non-linear functional relation has been approximated.

Three-stage least squares (3SLS) amounts to applying SUR to the second stage equations. Like SUR, GLS, and maximum likelihood methods in general, it rests upon the assumption that errors that recur in certain patterns are more palatable than "erratic" errors of the same size. Given that rather strange assumption, it is hardly surprising that its use has not, so far as I am aware, improved the performance of any model.

The combination of G and Build makes possible another approach to the problem of simultaneous equation bias which avoids both of the difficulties with 2SLS.   It may be called Systemic Two-stage Least Squares, S2SLS, for it makes use of the whole model or system of which the equation is a part. I have to tell you at the outset that it sounds good in theory but does not work well. It goes as follows.

1. Use OLS to estimate the equations of the model.

2. Put the model together and run it in historical simulation. This can be a "static" simulation which uses the historical values for all lagged values. (Just give the command *static* at the] prompt before giving the command *run.*)

3. Use the predicted values from the model as the values of the simultaneous independent variables and re-estimate the equations.

It is clear that the estimates from the third step will not suffer from simultaneous equation inconsistency, for the independent variables are computed without any knowledge of the errors in the equations. There is also no particular problem about nonlinearities; the nonlinearities in the model are fully incorporated in the calculation of the historical simulation values. Nor is there any problem about perfect fit on the first stage, unless, of course, the model is perfect -- a situation we need not worry about.

After step 2, change the names of original .sav files and of the bws and histsim banks by these G commands
>   dos ren *.sav   *.sv1
>   dos ren   histsim.*   histsim1.*
>   dos ren   bws.*   bws1.*

Then , in preparation for step 3, edit the .reg files where you want to apply the technique and put a "b." in front of each variable for which you want the program to use the value from the first step. Save the file with a different name; for example, save the changed cR.reg as cR2.reg. Do *not*, however, change the name of file saved in the *save* commands.

Then do

bank histsim1   b

and *add* the newly edited files which do the regressions. Build and run the model as usual.

My experience with the method has been no better than with the others, which is to say, not good. You may certainly try it, but it has never given results that I wanted to use.

If you have studied econometrics, you have perhaps learned that the supposedly ultimate method in the area of estimation of simultaneous equation models is something known as Full Information Maximum Likelihood or just FIML. Its theoretical statistical properties are about the same as those of 3SLS, so there is little reason to prefer it.

Does G offer FIML? No, but I am glad you asked, for this very FIML offers one the clearest examples of the way that maximum likelihood estimates prefer large, systematic errors to small erratic ones. To explain the example requires the notion of a *recursive* system. A simultaneous system is recursive if it is possible to write the equations in an order so that the first variable depends only on predetermined variables (exogenous and lagged values of endogenous), the second variable depends only on the first and predetermined variables, the third depends only on the first two and predetermined variables, and so on.

When applied to a recursive system, FIML leads -- via a long derivation which need not detain us [5] -- to minimizing the determinant of the matrix of sums of squares and cross products of the residuals. To be specific, let us suppose that we have a system of two equations and the dependent variable of equation 2 does not appear in equation 1 -- the condition that makes the system recursive. Let the misses from the first equation form the first column of a matrix R while the misses of the second equation form the second column. FIML then minimizes the determinant of R'R. Consider two estimates of the parameters. One gives the R'R matrix on the left below; the other gives the R'R on the right. Which estimate would you prefer:

$$R'R = \begin{pmatrix} 4 & 0 \\ 0 & 4 \end{pmatrix} \quad \text{or} \quad R'R = \begin{pmatrix} 8 & 7 \\ 7 & 8 \end{pmatrix} \quad ?$$

I have no hesitancy in saying that, other things equal, I would prefer the one on the left. The criterion used in FIML, however, chooses the estimate on the right, for its determinant is 15 while the determinant of the matrix on the left is 16. In this example, it is clear how maximum likelihood methods tolerate large errors in one place if they are correlated with large errors in another, but are strongly averse to erratic errors. If I, however, have one equation for residential construction and another for non-residential, and the first over-predicted last quarter, I am not at all consoled to discover that the other also over-predicted, even if they have often done that

---

[5] See G.S. Maddala *Econometrics* (1977, McGraw-Hill) page 487, the formula between C-50 and C-51. For a recursive system |**B**| = 1 in Maddala's notation. This useful appendix was omitted from the second edition of the book.

before. To use FIML without being fully aware of this tendency is naive, more naive than using plain OLS with full consciousness of its problems.

9. Vector Autoregression

Following the discussion of autoregression is a natural place to say a few necessary words about vector autoregression (VAR), which has been a "hot" topic in recent years. The idea is simplicity itself. Let us consider a system described by the equations

$$(1) \qquad x_t = A_0 x_t + A_1 x_{t-1} + \ldots + A_p x_{t-p} + f(t) + \varepsilon_t$$

where x is a vector of stationary variables, the A's are constant matrices, f(t) is a vector of exogenous variables, and ε_t is a vector of random, exogenous variables. The classical school of econometrics investigated the conditions under which the A matrices, especially A_0, could be identified. These conditions involved some sort of prior knowledge, usually that some of the elements of A_0 were zero. The VAR school [See Sims 1980] rejected the notion of prior knowledge and also of the division of the variables between endogenous and exogenous. They therefore dropped the *f(t)* term of (1), used only stationary variables, and moved the first term to the left, so that (1) became

$$(2) \qquad (I - A_0)x_t = A_1 x_{t-1} + \ldots + A_p x_{t-p} + \varepsilon_t.$$

On pre-multiplying both sides of (2) by the $(I - A_0)^{-1}$ we get an equation of the form

$$(3) \qquad x_t = B_1 x_{t-1} + \ldots + B_p x_{t-p} + \eta_t$$

where

$$(4) \qquad B_i = (I - A_0)^{-1} A_i$$

Clearly nothing can be said about the B_i matrices, except that if A_i is all zero, so is B_i.

Christopher Sims's initial experiments with the VAR approach simply regressed each variable on the lagged values of all the others, and made a model out of the results. By careful selection of the variables – and thus a lot of implicit theorizing – he was able to get two simple models that made the approach look promising.

Soon, however, it turned out that successful unconstrained VAR models were uncommon. Soft constraints were then introduced to softly require that the diagonal elements of B_1 should be 1.0 and that all other elements of the B's should be zero. In other words, it was assumed that each equation consisted principally of regression on the lagged value of the dependent variable. The regressions with soft constraints were referred to as *Bayesian* regression because of the thought processes used in picking the strength of the constraints. The result was therefore referred to as Bayesian vector autoregression, or BVAR.

The BVAR's have proven much more useful than the VAR's. One should not miss the irony in this outcome. The VAR school began with total agnosticism; it denied all *a-priori* knowledge of the values of parameters. The BVAR school then proceeds to assume *a-priori* values for all parameters!

I believe that you can see why one who hopes, as I do, to use models to express and test our understanding of the economy will not be very interested in the a-theoretic VAR or BVAR approach. It seems to have rejected propositions like "Personal consumption expenditure is more likely to depend on after-tax income than before-tax income," as unfounded assumptions and then to have embraced the assumption that all variables are determined mainly by their lagged values. Such an apotheosis of the lagged value of the dependent variable is not likely to appeal to one who has seen the dangers of the lagged values of the dependent variable, as shown in Chapter 6.

On the other hand, as a purely mechanical, mindless way to forecast several variables one or possibly two periods ahead, the BVAR method is reported to be moderately successful.

10. Cointegration, Unit Roots

In section 7, we have looked at the estimation of equations with moving average errors. If, in equation (2) of that section, all the *x* variables are just lagged values of the dependent variable, the equation become the autoregressive moving average (ARMA) equation

(1) $y_t = b_0 + b_1 y_{t-1} + b_2 y_{t-2} \dots + b_p y_{t-p} + \varepsilon_t + h_1 \varepsilon_{t-1} \dots + h_q \varepsilon_{t-q},$

where ε_t is white noise. We will find it useful to write (1) with the lag operator L, thus:

(2) $(1 - b_0 + b_1 L - b_2 L^2 \dots - b_p L^p) y_t = (1 + h_1 L \dots + h_q L^q) \varepsilon_t$

or, for short,

(3) $B(L) y_t = H(L) \varepsilon_t$

where *B(L)* and *H(L)* are polynomials in *L*.

In these equations, y_t is white noise transformed by an ARMA process. We have previously viewed (1) as a rough-and-ready way to forecast a variable for which we can think of no better explanation. Beginning around 1970, however, this sort of equation came to be used in ever wider circles to *define* what was meant by the expression *time series*. The term *time series analysis* is now, sadly, often used to refer exclusively to the study of these things which might better be called "ARMA-ed random variables" or ARVs for short.

Now I must say plainly that I do not think that any of the series in the national accounts of the United States or any other country, or any other major economic series is an ARV. Their changes over time are profoundly influenced by tax rates, government spending, money supply and many

other variables that are the product of thoughtful human decisions. History matters, not just random variables transformed by a constant ARMA process. To limit the term "time series" to mean "ARV" is therefore a pretty strange use of words, since most time series in the broad and natural sense of the words cannot be "time series" in the narrow sense. Consequently, I will call an ARV an ARV.

In the limited world of ARV's, however, it is possible to give precise meaning to some terms we have used in broader senses. "Stationary" is a good example. If $y(t)$ is an ARV and $E(y(t)) = \mu$ for all t and $E(y(t) - \mu)(y(t-j) - \mu) = \gamma_j$ for all t and any j, then $y(t)$ is said to be *covariance stationary* or *weakly stationary*. Although there is a concept of *strictly stationary*, it is a common practice that I will follow to use simply "stationary" to mean "covariance stationary".

ARV's are, of course, special cases of the solutions of systems of linear difference equations studied in Chapter 7, namely, the special case in which the input function is a weighted average of q values of a white noise variable. An ARV can be stationary only if all the roots of the homogeneous linear difference equation – that is, of the polynomial $B(L)$ – are inside the unit circle in the complex plane. Otherwise, it will, as we have seen, be explosive and not have a constant mean, as required by the definition of stationarity.

Clearly, economic series characterized by growth cannot even seem to be a stationary ARV. Consequently, one may want to investigate a class of ARV's whose first differences are stationary ARV's. If y_t is stationary, we can write

(4) $y_t = \dfrac{H(L)}{B(L)} \varepsilon_t,$

and if x_t is a variable whose first difference is equal to y_t , that is, $(1 - L)x_t = y_t,$ then

(5) $(1 - L)x_t = \dfrac{H(L)}{B(L)} \varepsilon_t$

or

(6) $(1 - L)B(L)x_t = H(L)\varepsilon_t.$

Thus, it is clear that the characteristic polynomial of x_t has the same roots as does that of y_t plus one more real root equal to exactly 1, that is to say, a *unit root*. Because x_t is created by summing successive values o f y_t , a process that would correspond to integration if we were working with continuous time, x_t is said to be *integrated of order 1* or *I(1)* for short. A stationary ARV is correspondingly said to *integrated of order 0,* or *I(0)* for short.

We can now at last say what is meant by *cointegration*[6]. If x_t and y_t are two I(1) ARV's and if there exits a number β such that $y_t - \beta x_t$ is I(0), that is, stationary, then x_t and y_t are said to be *cointegrated.*

Intuitively, if x_t and y_t are cointegrated, it makes sense to regress one on the other; the residual will not grow ever larger and larger. There is thus a sort of equilibrium relation between x_t and y_t . On the other hand, if they are not cointegrated, they may drift apart over time without any persistent relation between them.

Cointegration is definitely a good thing to have in your regression. For one thing, it can be shown that the ordinary least squares estimate is "superconsistent" in the sense that it converges to the true values at a rate of T instead of $\sqrt{T}$. Moreover, cointegration sometimes can resolve identification in simultaneous equation systems that are not identified by the classical rules. For example, if we regress the price (p) on the quantity demanded (q) with data from a market described by the following two equations,

$$\text{demand curve:} \quad p_t + 1q_t = u_{d\,t} \qquad u_{d\,t} = u_{d\,t-1} + \varepsilon_{d\,t}$$
$$\text{supply curve:} \quad p_t - 1q_t = u_{s\,t} \qquad u_{s\,t} = \rho u_{s\,t-1} + \varepsilon_{s\,t} \qquad |\rho| < 1$$

we will get a consistent estimate of the supply curve! Why? Note that u_d is I(1), so p and q , which are both linear combinations of u_d and u_s , is also I(1). Moreover, the supply is a cointegrating relation between them. Ordinary least squares will pick it out because it will have a finite variance, while the variance in the demand curve goes to infinity as T does. You can see this phenomenon by running a number of times this regression file:

```
fdates 1960.1 2010.4
f ed = @normal()
f es = @normal()
f ud = @cum(ud,ed,0)
f us = @cum(us,es,.8)
f p = .5*ud + .5*us
f q = +.5*ud - .5*us
lim 1965.1 2010.4
r p = ! q
```

You will almost certainly get a coefficient on q close to 1.0, that is to say, an estimate of the supply curve. (How to know in any real situation which curve has I(0) residuals is, of course, another matter.)

Clearly, if you are trying to explain by regression an I(1) ARV, y_t , you want to have a cointegrated x_t among the independent variables. The regression on only this variable, however, may not be very good. The residuals may be an ARV with a rich structure which could be

---

[6] The classic reference from which the concept of cointegration caught on in the economics profession is R.F. Engle and C.W.J. Granger, "Cointegration and Error Correction: Representation, Estimation and Testing," *Econometrica*, Vol 55, No. 2, 1987, pp 251-276.

exploited for better fit and forecast. This ARV might be a linear combination of various stationary ARV's that you can observe. The original Engle and Granger article suggested estimating first the cointegrating equation and then estimating another, the so-called error correction equation, for dynamic adjustment in the equation. Other studies by random simulation experiments[7] found that it was better to put the dynamic adjustment into the initial estimation. Another result of the theory is that if x_t and y_t are cointegrated, the regression should be done between them, not their first differences, as was previously frequently advised.

Right from the beginning of this book, we have followed methods that seem to be suggested by the theory of cointegration. In the investment equation for the AMI model, the dependent variable clearly has a trend, as does the replacement variable. The first differences in output, on the other hand, might or might not be I(1); yet they clearly add an important element to the regression. If they are I(1), then they become part of the cointegrating vector; if not, they contribute to explaining the residuals; we don't have to decide which they do. But now suppose that we want to add the real interest rate as an explanatory variable. It is clearly not I(1), so let us suppose it is I(0).   Then, as far as the theory of cointegration offers a guide, we could add it into the equation. But here is where this theory is an insufficient guide. If, over time, the effect on investment (measured in constant dollars per year) of a one percentage point change in the interest rate has increased because of the increase in the size of the economy, then the variable we need is not the interest rate itself but its deviation from mean *multiplied* by some measure of the size of the economy. This procedure, of course, has already been advocated. Finally, the cointegration literature advises analyzing the error term and adding a projection of it to the equation in forecasting. That is exactly what our rho adjustment does automatically.

If cointegration is so nice, perhaps you would like *prove* that your equation has it. My advice is "Forget it!". You need to prove that your residuals do *not* have a unit root. You might think that all that you need do is to regress the residual of your regression on its lagged value and test whether or not the regression coefficient could be 1.0. Stochastic experiments (often called Monte Carlo experiments after the famous gambling casino) with made up data have shown that if you used the ordinary *t* or normal tables, you would far too often conclude that you had found cointegration. Tables for this test based on these experiments and published D.A. Dickey and W.A. Fuller should be used for such testing.[8] Using these tables, it is usually impossible to reject the hypothesis that there is a unit root -- and therefore no cointegration. Why it is so hard to reject the unit root hypothesis is clear if we recall the graph from the stochastic simulation of three models from Chapter 7 and reproduced here for ease of reference. The inner of the three lines we know, from the way it was generated, to be I(0), while the middle one we know to be

---

[7]A. Banerjee, J. Dolado, D.F. Hendry, and G. Smith, "Exploring Equilibrium Relationships in Econometrics through Static Models: Some Monte Carlo Evidence," *Oxford Bulletin of Economics and Statistics,* Vol. 48, 1986, pp. 253-277.

[8] They are reprinted in James D. Hamilton, *Time Series Analysis,* Princeton, 1994, Appendix D.

I(1). There is clearly not much difference between the two. If one were given the inner series, it would be hard to prove without an enormous amount of data that it was not I(1). The usual result, therefore, is that the hypothesis of a unit root in the residuals cannot be absolutely ruled out, although they very well may not have one. So we come back to common sense: if the equation makes sense and the value is ρ is modest, use it.   I can only agree with Maddala's summary of the situation, "In a way, in the case of both unit roots and cointegration, there is too much emphasis on testing and too little on estimation."[9] .

Stochastic Simulation
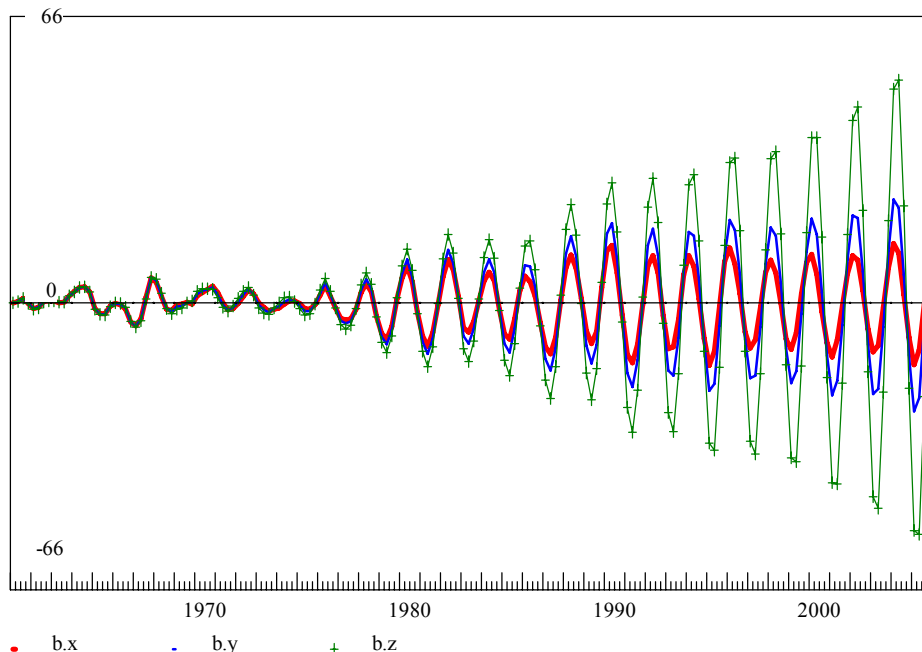
I find cointegration a useful concept to bear in mind in formulating a regression. It also, almost incidentally, gives us one more reason for wanting the value of ρ, the autocorrelation coefficient of the residuals, to be well below 1.0. On the other hand, complicated testing of whether the residuals actually are stationary is so apt to prove indecisive that it is hardly worth bothering with. Economic understanding of the situation we are modeling is much more helpful than mechanistic analysis based, ultimately, on the assumptions that the series involved are ARV's, when they almost certainly are not.

---

[9] G.S. Maddala, *Introduction to Econometrics,* 2nd Edition, New York, Macmillan, 1992, p.601.

Chapter 11. Nonlinear Regression

Occasionally, it is necessary to estimate a function which is not linear in its parameters. Suppose, for example, that we wanted to estimate by least squares a function of the form

$$(1) \qquad y = (1 - (1-x)^{a0})^{a1} .$$

There is no way to make this function linear in the parameters a0 and a1, and estimate them by ordinary least squares. We will have to resort to some variety of non-linear technique. There are many of these, and each has its merits and its problems. None is guaranteed to work on all problems. The one built into G has worked on most problems I have tried it on, but if you find a case where it does not work, please let me know.

Generally, nonlinear methods need to be given starting values of the parameters. The method then varies the parameters to "feel around" in the nearby space to see if it can find a better point. If a better point is found, it then becomes the home base for further "feeling around." The methods differ in the ways they "feel around." While some methods use only the starting values, the one adopted here allows the user to specify also the initial variations. These variations are then also used in terminating the search.

1. Lorenz curves

We will illustrate the method with an example of fitting a Lorenz curve to data on earned income from a sample of 2500 individuals from the 1990 *U. S. Census of Population and Housing*. A Lorenz curve, $y = L(x)$, shows, on the y axis, the fraction of income total received by those persons whose income was in the lowest 100x percent. Thus the point (.50, .21) would indicate that the lowest 50 percent of the population gets 21 percent of the total income. Notice that any Lorenz curve, $L(x)$, must have the properties that $L(0) = 0$, $L(1) = 1$, $L'(x) \geq 0$, and $L''(x) \geq 0$ for $0 \leq x \leq 1$. Any function with these properties we may call a Lorenz function. If $L_1(x)$ and $L_2(x)$ are both Lorenz functions, then $\lambda L_1(x) + (1 - \lambda)L_2(x)$ with $0 \leq \lambda \leq 1$ is also a Lorenz function, as is $L_2(L_1(x))$ – in words, convex combinations of Lorenz functions are Lorenz functions and Lorenz functions of Lorenz functions are Lorenz functions. Here are two examples of Lorenz functions, as may be quickly verified.

$$L_1(x) = x^{\beta} \quad \text{with } \beta \geq 1 \quad \text{and} \quad L_2(x) = 1 - (1-x)^{\alpha} \quad \text{with } 0 \leq \alpha \leq 1.$$

Then, using the fact that Lorenz functions of Lorenz functions are Lorenz functions, we see that equation (1) above is in fact a Lorenz function for $a1 \geq 1$ and $0 \leq a0 \leq 1$. It is this form that we shall fit to our data. (The first use seems to be in R.H. Rasche *et al.*, "Functional forms for estimating the Lorenz Curve," *Econometrica*, vol. 48, no. 4, [1980], pp 1061-1062.)

Because, unlike nearly all other examples so far in this book, this data is not time series, G should be started in a special directory with a G.cfg file including the lines:

Default regression limits; 0 20 20
Default base year of workspace file; 0
First month covered; 1
Default maximum number of observations per series in workspace; 30

You can then introduce the data without the artificiality of using dates for observation numbers that, in fact, do not refer to dates. The part of the command file to read in the data is then

```
ti Lorenz curve for Average Income within Families
fdates 0 30
matdat 0
     x          y
 0.000000  0.000000
 0.050512  0.004259
 0.100082  0.013013
 0.150477  0.025874
 0.200518  0.042053
 0.250206  0.062013
 0.300012  0.085112
 0.350053  0.111390
 0.400212  0.140735
 0.450018  0.173396
 0.500059  0.209550
 0.550453  0.249876
 0.600377  0.294181
 0.650183  0.342411
 0.700224  0.395880
 0.750265  0.455556
 0.800071  0.523243
 0.850230  0.600566
 0.900035  0.690201
 0.950194  0.804458
 1.000000  1.000000;
gdates 0 20
gr x y
lim 1 20
```

Lorenz Curve for Earned Income
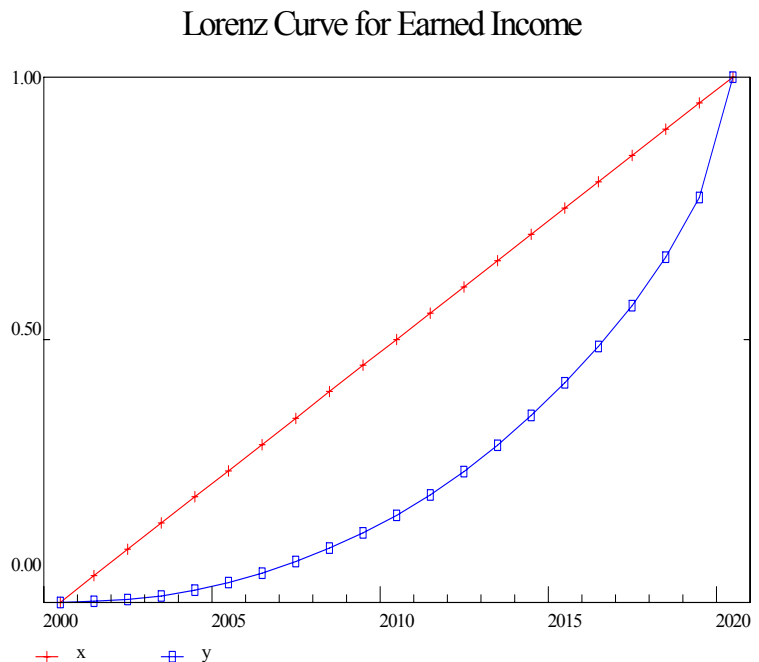
The *matdat* command reads in matrix data, that is, data in which values for different variables in a given period or unit of observation appear across a line. The number following the *matdat* command is the date, or in our case, the observation number of the first observation which follows. (If it is not given, then the date or observation number should appear at the beginning of each line.) Note the *;* at the end of the data. The *gdates* and *gr* commands are for visual checking of the data by the graph shown to the right. The *lim* commands sets the limits -- or range of observations -- for the nonlinear regression command that lies ahead.

The general form for doing nonlinear regression in G is the following:

nl [-] <y> = <non-linear function involving n parameters, a0, a1, ...an-1>
<n, the number of parameters>
<starting values of the parameters>
<initial variations>

111

The optional - following the *nl* will cause printing of intermediate results. Normally it is not necessary. The commands to do the nonlinear regression are then,

```
nl y = @exp(a1*@log(1.-@exp(a0*@log(1.-x))))
2
0.5  2.5
0.05 0.1
gr *
```

The results are:

Lorenz Curve for Earned Income
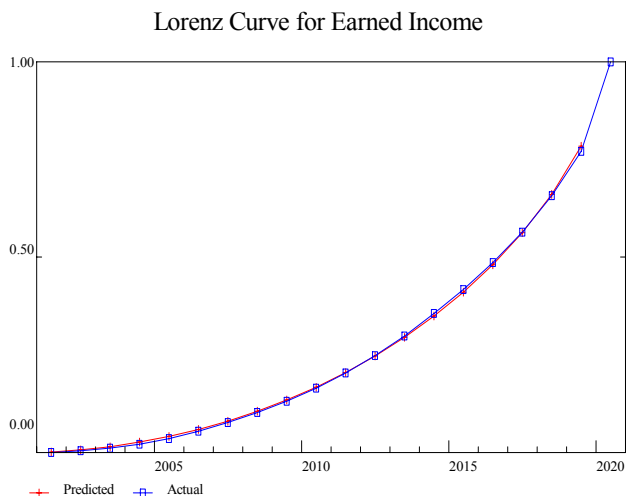
```
Lorenz curve for Earned Income
SEE = 0.223659
Param          Coef      T-value       StdDev
    a0       0.698208     1.81       0.386664
    a1       1.863359     1.53       1.215625

The Variance-Covariance Matrix
  1.4951e-01    4.4554e-01
  4.4554e-01    1.4777e+00
```

A word must be said about t-values and standard deviations in non-linear regression. They are computed by G in the way described in a standard way by linearizing the function around the optimal point. Consequently, they are only as good for movements only within the range of approximate validity of these linearizations. In the present case, it might appear from the standard deviations that a1 could easily be less than 1 and a0 could easily be more than 1. But for such values of a0 and a1, the function is not a Lorenz curve! Thus, utmost caution should be used in interpreting or relying on these statistics.

Now it may appear from the above graph that this form of the Lorenz curve fits this data extremely well. But one of the uses of a Lorenz curve is to calculate the amount of income within

various brackets. We can look at the percentage error in the income for each of the 20 "ventile" brackets of the data by the commands:

```
update predic
0   0.
20 1.000
fdates 1 20
f difp = predic - predic[1]
f difa = y - y[1]
f difrel = 100.*(difp - difa)/difa
```

Alternate Fit Relative Error in Income
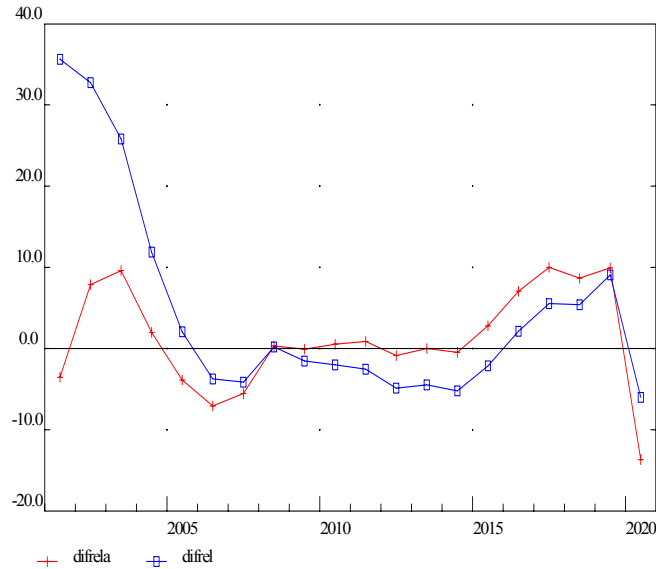
The result is shown by the line marked with squares in the graph above. The errors of over 30 percent in the two lowest brackets are quite likely unacceptably high. Thus, far from fitting virtually perfectly, as one might gather from the first graph, the fit leaves a lot to be desired.

The first step towards a better fit is to fit so as to minimize the sum squares of these percentage errors. That can be done by the following commands.

```
gdates 1 20
fdates 0 20
f ze = 0
f difrela = 0
f z = 0
fdates 1 19
ti Alternate Fit Relative Error in Income
nl f z = @exp(a1*@log(1.-@exp(a0*@log(1.-x))));
    f difp = z - z[1];
    f difrela = (difp - difa)/difa;
    ze = difrela
2
0.5 2.5
0.01 0.01

# Put the 0 and 1 in z at the beginning and end
update z
```

```
0  0
20 1.
fdates 1 20

f difrela = 100.*(z - z[1] - difa)/difa
vr -20 -10 0 10 20 30 40
gr difrela difrel
vr off
```

Here we have employed the capacity of G to use a number of statements in the course of defining the predicted value. The first of these, on the same line with the *nl* command, calculates a variable called *z* from the formula for the Lorenz curve. The second line computes *difp,* the fraction of income in each bracket (except the last).   The third line then calculates *difrela,* the percentage errors in these income fractions. Notice that each of these intermediate lines ends with a *; .* The final line, which does *not* end in a *; ,* has the desired value on the left (in this case, zero) and the predicted value on the right. The remaining lines calculate the values of the difference for the whole range of the function, including the uppermost bracket and produce the graph shown above. The new fit is shown by the curve marked with + signs.

The fit is generally improved but is poor enough to invite us to try a different form of Lorenz curve. As already observed, the product of any two Lorenz curves is also a Lorenz curve, so we could take a product of the Rasche curve we have estimated so far with a simple exponential. The commands for estimating this function are

```
nl f z = @exp(a2*@log(x))*@exp(a1*@log(1.-@exp(a0*@log(1.-x))));
   f difp = z - z[1];
   difrelpa = (difp - difa)/difa;
   ze = difrelpa
3
0.15 .17  2.
0.01  0.01 .01
```

The numerical results are

```
Param           Coef       T-value        StdDev
   a0         0.405709      5.52        0.073498
   a1         0.523675      3.93        0.133357
   a2         1.536559     10.38        0.147986
```

The relative errors are shown in the graph below by the line marked by squares. Comparison with the relative errors of the simple Rasche suggested that a linear combination of the two types of curves might be tried. In the full function, there would have been six parameters to estimate. G's algorithm kept pushing the Rasche parameters that have to be between 0 and 1 out of that range, so they were fixed at values found previously. The first line of the estimation was

```
nl - f z = a0*@exp(a1*@log(x))*@exp(a3*@log(1.-@exp(.15366*@log(1.-x))))
   + (1-a0)*@exp(a2*@log(1.-@exp(.6982*@log(1.-x))));
```

And starting values were:
```
   4
```

```
      0.5 2.0451  1.8633 .17024
      0.01 .01     .01     .01
```

The resulting relative errors are shown by the line marked with + signs in the graph below. The fit is considerably improved for the first four brackets and is about the same for the rest of the curve.

Relative Errors of Composite Lorenz Curves
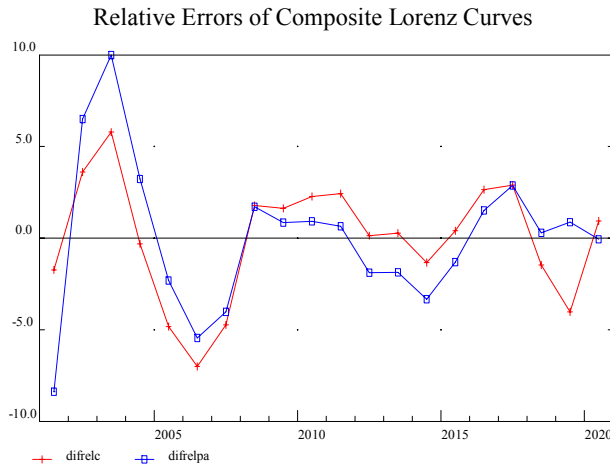
In this function note the -   after the *nl* command; it turns on debugging dumping of the value of the objective function and of the parameters every time an improved solution is reached.

2. The downhill simplex method, Powell's direction set method, and details

We have spoken above rather vaguely of "feeling around" by the nonlinear algorithm. Now we need to describe a bit more precisely what is happening.

Initially, S, the sum of squared errors, is calculated at the initial value of each parameter. Then, one-by-one, the parameters are changed by adding the initial variations, and S is recalculated at each point, thus yielding values at n+1 points (a simplex). Points in the simplex are then replaced by better points generated by the "reflection, expansion, or contraction" operations to be described in a moment or the simplex is shrunk towards its best point. The process continues until no point differs from the best point by more than one-tenth of the initial variation in any parameter.

New points for the simplex are generated and selected in a way best described in a sort of "program in words" as follows:

 Reflect old worst point, W, through mid-point of other points to R(eflected).
 If R is better than the old best, B {
    expand to E by taking another step in the same direction.
    if E is better than R, replace W by E in the simplex.
    else replace W by R.

```
     }
Else{
    contract W half way to mid-point of other points, to C(ontracted)
    if C is better than W, replace W by C.
    Else Shrink all points except B half way towards B.
    }
```

As applied in G, once the algorithm converges, steps of the initial size are made in each direction around the presumed optimum. If any better point is found, the program prints "Fresh start" and starts the process again. It is not unusual to see several "Fresh start" notices.

Though, like all non-linear algorithms, this one is not guaranteed to work on all problems, it has certain advantages. It is easily understood, no derivatives are required, the programming is easy, the process never forgets the best point it has found so far, and the process either converges or goes on improving forever.   While by no means the most "sophisticated" of algorithms, it has a reputation for robustness.

The principal problem that I have with the algorithm is that it sometimes tries to evaluate the function at points that lead to arithmetic errors. For example, it may try to evaluate the logarithm of a negative number.   My advice in such cases is to use the debugging dump option, the -   after the *nl* command. You will often see what parameter is causing the trouble. Use the information from the dump to get a better starting point and use a rather small initial step size.

 The recovery of G in cases of arithmetic error leaves something to be desired. You will get the message that the error has occurred and you click "Cancel" in order not to see the error many more times. Unfortunately that click which stopped the execution of the command file did not close that file. Attempts to save the command file from the editor will be refused. Instead, use File | Save as ..   and save with some other name, like "temp." Exit G, restart it, bring "temp" into the editor and use File | Save as .. to save it with its proper name.

Soft constraints on the parameters can be built into the objective function. For example,

$$nl \; zero = @sq(y - (a0 + a1x1 + a2x2)) + 100*@sq(@pos(-a2))$$

will "softly" require a2 to be positive in the otherwise linear regression of y on x1 and x2. The word "zero" on the left side causes G to minimize the sum of the expression on the right rather than the sum of the squares of the differences between it and the left side. The word "last" on the left causes G to minimize the value of the expression in the last observation of the fit period. This feature can be used in conjunction with the @sum() function -- which puts the sum of its argument from the first to last observation of the fit period into the last observation.

Following the *nl* command, there can be f commands, r commands, and con commands before the non-linear equation itself. These may contain the parameters a1, a2, etc.; they should each be terminated by ';'. The non-linear search then includes the execution of these lines. E.g.:

116

```
nl x1 = @cum(s,v,a0);
y = a1 + a2*x1
```

These special features allow G's non-linear command to handle a wide variety of non-linear problems such as logit analysis, varying parameter estimates, and errors-in-variables techniques. It is beyond our scope here to explain all these possibilities.

Finally, the *save* command creates a file with the nonlinear equation with the estimated parameters substituted for a0, a1, etc. And the *catch* command captures the output to the screen as usual.

Besides the downhill simplex method, G also has available nonlinear regression by *Powell's direction set method*. The format for using it is almost exactly the same except that the command is *nlp*. The line of "step sizes," however, is used only as a convergence criterion. As in the *nl* command, when one iteration of the algorithm does not change any parameter by more than one tenth of its "step size," the process is declared to have converged.

Powell's method uses a sequence of one-dimensional minimizations. For a problem with *n* parameters, the method has at any time a set of *n* directions in which it minimizes. It starts simply with the unit vectors in n-dimensions. It does a one-dimensional minimization first in the first direction, then from the point found in that direction, it does another one-dimensional minimization in the second direction, and so on. When a minimization has been done in each of the *n* directions, the net step, the vector difference between the final point and initial point, usually enters the set of directions in place of the direction in which minimization produced the largest drop. Another step equal to the net step is then tried. The process is then repeated. In some situations, however, it is not desirable to change the set of directions. The exact criterion and details of the algorithm are given in William H. Press, et al. *Numerical Methods in C* (Cambridge University Press, 1986.))

Which method is better? Powell is supposed to be a more "sophisticated" use of the information; and in very limited comparative tests, it has reduced the objective function faster in terms of the number of evaluations of the function. This sort of comparison can be made by running both methods on the same problem and using the ' - ' following the command to show the progress of the algorithm. However, you may find the problems on which the downhill simplex works best. In any event, neither algorithm is perfect, so it is good to have a second in case one fails to work.

3. Logistic functions

Besides Lorenz curves, another common application of nonlinear regression in economics is to the estimation of *logistic* functions. These functions, sometimes called growth curves, often describe fairly well the path of some variable that starts slowly, accelerates, and then slows down as it approaches an asymptotic value. They can also describe a declining process. The general form is

$$y(t) = a_0 + \frac{a_1}{1 + a_2 e^{a_3 t}}$$

Two examples are shown below. The rising curve has a negative a_3 parameter; the falling curve has a positive. Otherwise, the parameters are the same so that the curves look like mirror images of one another around the point where $t = 0$, chosen for this example in 1981. (This origin of time is *not* another parameter, for any change in it can be compensated by a change in the a_2 parameter.) The formulas were

```
f tm21 = time - 21
# the rising curve
f y1 = .05+ .2/(1 + .5*@exp(-.2*tm21))

# the falling curve
f y2 = .05+ .2/(1 + .5*@exp(.2*tm21))
```


Growing and Declining Logistic Curves
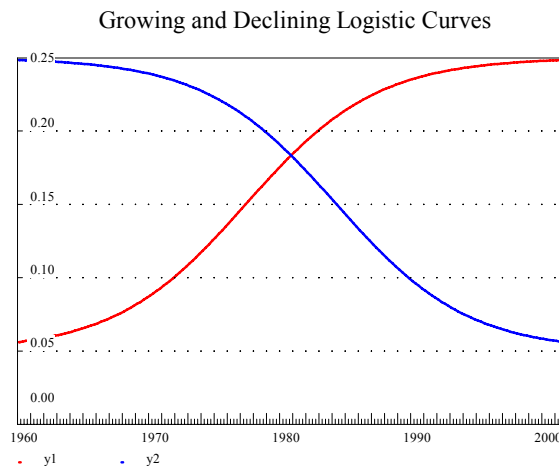
As an application of this family of curves, we may take the ratio of imports to GDP in the US in the period 1960.1 to 2001.4. The historical course of this ratio is shown by the irregular line in the graph below. The logistic fit to it is shown by the smooth line.

Logistic Curve for Imports
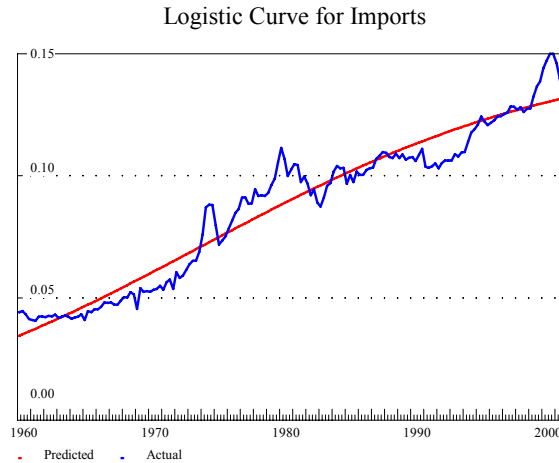
The G command for fitting this line are

```
ti Logistic Curve for Imports
f tm25 = time - 25.
lim 1960.1 2001.4
nlp firat = a0 +a1/(1. + a2*@exp(a3*tm25))
   4
 .04  .20   .02  -.01
 .001 .001  .001  .001
```

The numerical results were:

```
 Logistic Curve for Imports
SEE = 0.007542
Param          Coef     T-value       StdDev
    a0      -0.009051    -0.34       0.026917
    a1       0.160953     4.25       0.037894
    a2       0.468872     7.85       0.059702
    a3      -0.070660    -3.81       0.018550
```

Around the basic framework of the logistic, one can add variations. The asymptotes can be affected by replacing the simple constant a_0 by a linear expression in explanatory variables. The same can be done with the other constants. Indeed, the t variable need not be time but can be a function of other variables. Thus, the form gives rise to a large family of functions; they all require nonlinear estimation.

A final word of warning, however. Many logistic curves have been fit to rising series. Unless the curve has nearly reached its upper asymptote, the estimate of that asymptote has often proven unreliable. The first application of the curve was to automobile ownership in the United States. In about 1920, the researchers predicted that the market would be effectively saturated by 1923. Moreover, the rising logistic provides no information about when the decline will begin.

Chapter 12. Stochastic Simulation

When we estimate the mean household income of a city of 10,000 households on the basis of a random sample of 100 households, we can convey to the users of the estimate some idea of the accuracy of the number by stating also its standard deviation. Can we in some similar way give users of a model an idea of its reliability?

Yes and no. The comparison of the historical simulation of the model with the actual history is already conveys some idea of the accuracy of the model. In this chapter, we will show how to go further and recognize that we know that the regression equations are inexact and that they will almost certainly err in the future just as they have erred in the past. We will make up random additive errors for the equations that have the same standard errors and autocorrelation coefficients as were found for the residuals. We can then run the model with these random errors added to the equations. In fact, we can easily run it a number of times – 50, 100, or more – each time with a different set of random additive errors and calculate the mean and standard errors of each variable in the model.

We can go further and recognize that the regression coefficients are not known with certainty. We can generate random variations in them which have the same variance-covariance matrix as was found in the course of the regression calculation. While these calculations are most easily justified by invoking the Datamaker hypothesis, we can also say that we are interested in the model forecasts that would be generated by random variations in the coefficients that would not reduce the fit of equations by more than a certain amount.

In this chapter, we will see how to make such calculations. But we should be aware of the limits of these calculations. They do not tell us how much error may be introduced into the forecasts by errors in the forecasts of the exogenous variables. If we are willing to specify the extent of those errors, they too can be accounted for. But it is also possible that in the future one or more relation which has held quite dependably in the past may cease to hold. Or, following the line of the Lucas critique, we may by a change in some policy variable push the model into territory in which we have no experience and in which one or more of the equations ceases to work. The techniques explained here cannot be expected to warn us of such problems.

We will first explain simulation with random additive errors and then add the random coefficients as well.

1. Random additive errors

Of the residuals in each equation we know from the regression results the standard error, σ, and the autocorrelation coefficient, ρ. We need to make up random additive errors to the equation which have that same σ and ρ. From *Numerical Recipes in C* (pages 204-217), we borrow a (quite clever) pseudo random number generator that produces "random" independent normal deviates with mean 0 and variance 1. Let us multiply it by a constant, a, to get a variable ε with mean zero and variance a^2. From it, we can make up the variable ζ by the equation

(1) $\quad \zeta_t = \rho\zeta_{t-1} + \varepsilon_t$.

Now since the mean of ε is 0, so is the mean of ζ , while the variance of ζ will be given by

(2) $\quad \sigma_\zeta^2 = E(\zeta_t\zeta_t) = E((\rho\zeta_{t-1} + \varepsilon_t)(\rho\zeta_{t-1} + \varepsilon_t)) = \rho^2\sigma_\zeta^2 + \sigma_\varepsilon^2$

since ε_t is independent of ζ_t by construction. So

(3) $\quad \sigma_\zeta^2 = \sigma_\varepsilon^2/(1 - \rho^2) = a^2/(1 - \rho^2).$

If we now set σ_ζ^2 equal to the variance of the residuals from the regression, and ρ equal to the autocorrelation coefficient from the regression, we can solve this last equation for a, the factor by which the unit random normal deviates must be multiplied so that equation (1) will give a series of random additive error terms, ζ_t, with the required properties.

The application of stochastic simulation in G is extremely simple. First, the necessary information from the regressions must be saved in the .sav files. To do so, just give G the commands
    stochastic yes
    add runall.reg
in the white command box. The first turns on the saving of the necessary information for stochastic simulation; the second -- if you have kept your runall.reg up to date, just re-computes the equations with the extra information being saved. Then build the model as usual with Model | Build. When you do Model | Run, however, click the "stochastic" radio button on the right as shown below. When you do so, the extra stochastic options box appears, and you can specify the number of simulations you want to make and whether you want just the additive error terms, as shown here, or also random error in the regression coefficients. As shown below, the program is set to run 50 complete runs of the model with only additive errors.

When a model is run in stochastic simulation, it produces *two* output banks. One is named, as usual, in the "results bank" field on the "Run Options" form. It will contain the average value for each variable in the model from the simulations. The other is always called "sigma" and gives the standard deviation of each variable as found from the simulations.
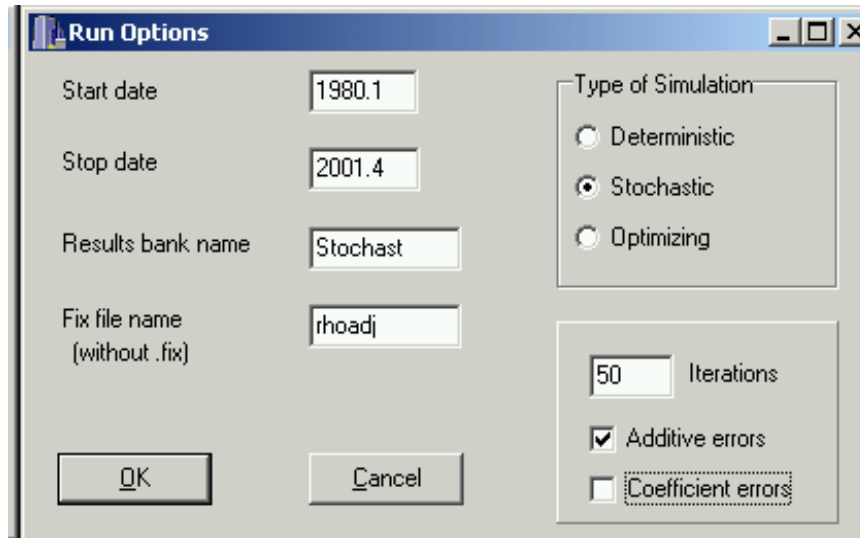
How these two banks can be used in making graphs is illustrated in the "show file" snippet below.

```
bank stochast d
bank sigma c
bank histsim b
bank bws e
gdates 1981.1  2001.4
#Use special graphics settings
add stoch.set

ti gdpD -- GDP Deflator
gname gdpD
f  upper = b.gdpD +1.*c.gdpD
f  lower = b.gdpD -1.*c.gdpD
gr  b.gdpD  upper lower d.gdpD e.gdpD

ti gdpR -- Real Gross Domestic Product
gname gdpR
f  upper = b.gdpR+2.*c.gdpR
f  lower = b.gdpR-2.*c.gdpR
gr b.gdpR upper lower d.gdpR e.gdpR
```

These two graphs for the optimized Quest model are shown below. In each case, there are *two* lines squarely in the middle of the channel or cone marked out by the solid lines on either side. One of these central lines (marked by + signs) is the average of the simulations, the other is the deterministic simulation done with all the error terms zero. Theoretically, in a nonlinear model the deterministic simulation is not necessarily the expected value of the stochastic simulations. In the case of these – and virtually all – variables is Quest, there is very little difference between them. The solid lines at the top and bottom are one standard deviation above and below the average. The line marked by Δ signs is the actual, historical course of the variable. It appears that the historical course generally stayed within the one sigma bounds.

gdpR -- Real Gross Domestic Product

gdpD -- GDP Deflator
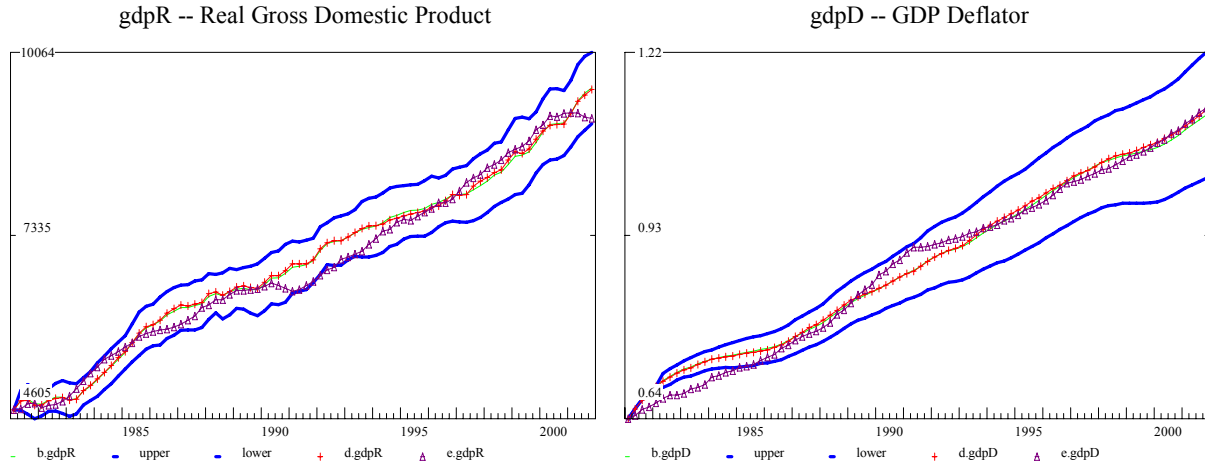
2. Random regression coefficients

In Chapter 9, section 1, we saw that we can, under the Datamaker assumptions, compute the variances and covariances of the regression coefficients by the formula

(1) $V = \mathrm{E}(b - \beta)(b - \beta)' = \mathrm{E}((X'X)^{-1} X'e\, e'X(X'X)^{-1})$
$= (X'X)^{-1} \sigma^2\, I\, X(X'X)^{-1}$
$= \sigma^2 (X'X)^{-1}$

We will see how to generate random error terms in the regression coefficients which will have this same V matrix of variances and covariances. These errors can them be added to the regression coefficients and the model run with the altered coefficients.

To generate random errors with the required variance-covariance matrix, we must compute the characteristic vectors and values (or eigenvectors and eigenvalues) of the V matrix. Since V is symmetric and positive definite, it is known by the principal axes theorem that there exists a matrix, P, of the characteristic vectors such that

(2) $P'P = I$

and

(3) $D = P'VP$

where D is a diagonal matrix with positive elements (the characteristic values of V) on the diagonal.

123

Equation (2) implies that $P' = P^{-1}$, but a left inverse is also a right inverse, so $PP' = I$. Multiplying (3) on the left by P and on the right by P' therefore gives

(4) $PDP' = PP'VPP' = V$

If we let R be the diagonal matrix which has on its diagonal the square roots of the diagonal elements of D, then $RR' = D$ and from (4) we have

(5) $PRR'P' = PP'VPP' = V$.

If ε is a vector of independent random normal variables with zero mean and unit variance, then,

(6) $\eta = PR\varepsilon$

is a vector of random variables that have V as their variance-covariance matrix, for

(7) $E(\eta\eta') = E(PR\varepsilon\varepsilon'R'P') = PRE(\varepsilon\varepsilon')R'P' = PRIR'P' = PRR'P' = V$

where the last equality follows from (5).

Computing the PR is a bit of work, so it is not done by G when the regression is estimated. Instead, when a regression is done by G after a "stochastic yes" command, the variance-covariance matrix of the regression coefficients is put into the .sav file. When the model is built, the PR matrix, called the *principal component matrix*, is computed and put into the heart.dat file. When the model is run in stochastic simulation with the "random coefficients" box checked, the η vector is computed and added to the point estimate of the regression coefficients. The coefficients thus generated are constant through any one run of the model, but many runs of the model may be made.

Computing of PR is done with algorithms from *Numerical Methods in C*; Householder's method is used to get a tridiagonal matrix, and the QL algorithm is used to finish the job.

The two graphs below show the results of fifty stochastic simulations of Quest with *only* random coefficients – no additive error terms. It is readily seen by comparison with the graphs of the additive errors that random coefficients are much less important, at least in Quest, than are the additive errors. It is also clear that the one-sigma range was not large enough to hold the historical series.

gdpR -- Real Gross Domestic Product

gdpD -- GDP Deflator

126