Exercise No. 6: Working with Time Series Data

New Stata Commands		**Old Commands Reviewed**	
append	smooth *sequence,twice ... gen(...)*	describe	label variable *var1* ...
%d *{date format}*	infile	sort *var*	graph twoway scatter
tsset		save	generate *variable* =...
gen *date=monthly(...)*		list in #/#	graph twoway line
format *date %tmCy/l*		use *"/path/dataset"*	
merge *var* using */path.../datasetname*		tabulate	
egen *var1=ma(var2)...*		drop *var1 var2 ...*	

This is a particularly demanding exercise, so get started soon. If you wait to the last moment you'll crash & burn. Any time I ask a question, provide an answer and paste those items into your exercise that are requested.

Data to download from the course website: **tler1.dta**, **tler2.dta, uerate.dta** and **pttwork1.dta**

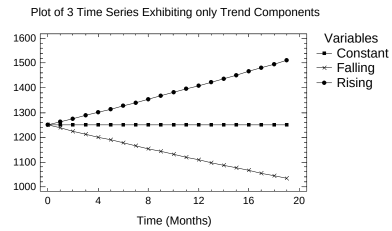
{Be sure to open a log and keep it for your records}

1 Introduction

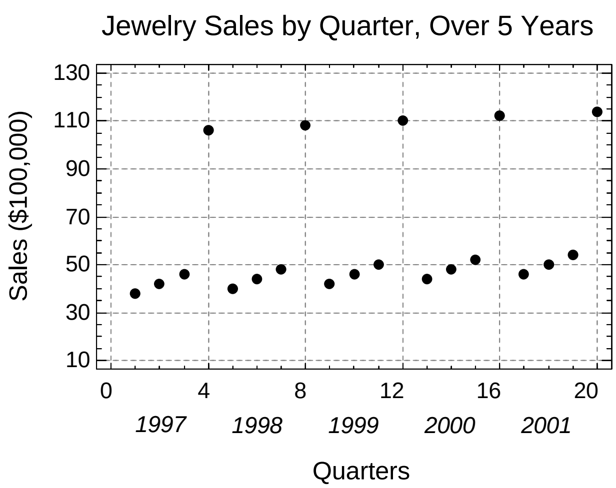
Economists use two major categories of statistical information: cross section and time series data. For example, the Bureau of Labor Statistics (BLS) conducts the ongoing *Consumer Expenditure Survey* (CES) in which approximately 5,000 households are interviewed four times during the year. Each of these quarterly interviews gathers data on the consumption expenditures of the 5,000 households *during a particular three-month period of time.* These data, relating to the consumption behavior of a large number of households at (roughly) the same time are called **cross section data.** On the other hand, the National Income and Product Accounts also measure consumption, but they do it repeatedly for the entire country *over an extended period of time.* Because these data relate to the time at which they are collected, they are called **time series data.** Many time series data are quite important in economic analysis: the unemployment rate, the inflation rate, the Standard & Poors 500 Stock Index, etc. This exercise will acquaint you with the rudiments of time series analysis, so you will at least be able to do descriptive analyses of time series data. Our focus will be on "smoothing" the time series data so that we can, in the words of John Tukey, obtain " ... [a] clearer view of the general, once it is unencumbered by detail."

1.1. Components of Time Series Data

- ***Trend Component***. Time series often move fairly smoothly in one direction over an extended period of time. If they do, we say that the data exhibit a **trend.** The figure below shows three time series exhibiting a rising, constant, and falling trend respectively.

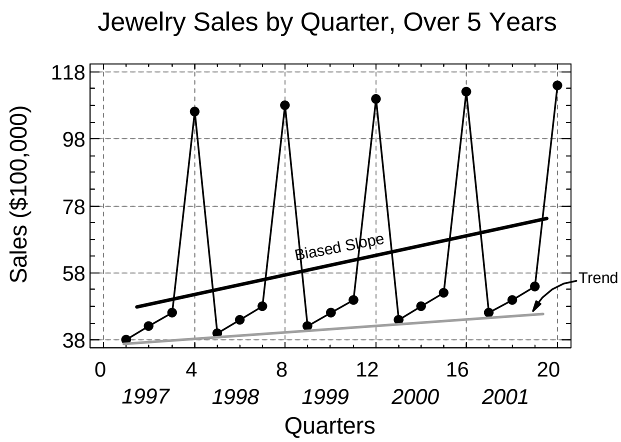


- ***Seasonal Component.*** Many time series show special behavior at specific seasons of the year. Look at the following chart showing the value of jewelry sales in specific quarters of the year over a five year period:



1. Why do you think that sales jump so much in the last quarter of each year? That is, what accounts for the seasonal pattern in jewelry sales?

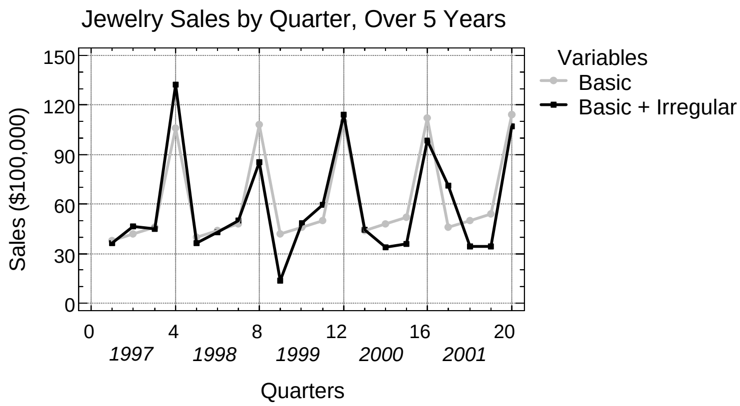
If you were to draw a trend line through these data without taking special account of the seasonal jump in sales, you might draw the heavy black line in the graph below:



This line would have a biased slope and would be too steep; the true trend would be better approximated by the gray "Trend" line in the figure. Adjustment of time series data to remove seasonality is done for a number of reasons:

1. Seasonal adjustment allows reliable comparison of observations at different points in time.
2. The task of understanding the relationships among economic variables is made easier once the complicating factor of seasonality is purged from the data.
3. Seasonal adjustment may be a useful element in an approach to the production of short-term forecasts of future values of a time series.

♦ **Irregular component.** Unfortunately, time series data are generally never as regular as the jewelry series sales shown above. Other factors -- some totally random, others not -- intervene to make the series much more jagged looking and less interpretable. These factors make up the **irregular component** of the time series shown incorporated into the graph below. The time series incorporating the irregular component (black line & squares) is much less easy to interpret than the time series that incorporates only the trend and seasonal components (gray line & circles)



So, in studying a time series we often are interested in differentiating among its components so that we can identify the trend, the seasonal and the irregular components. It's often convenient to think of any particular time series as being composed of these three components. Of course, how these components are related is another difficult issue. Some possibilities include (T_t = trend, S_t = seasonal, I_t = irregular):

1. **additive components model**

$$X_t = T_t + S_t + I_t$$

2. **multiplicative components model**

$$X_t = T_t S_t I_t$$

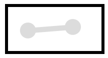
3. **additive/multiplicative components model**

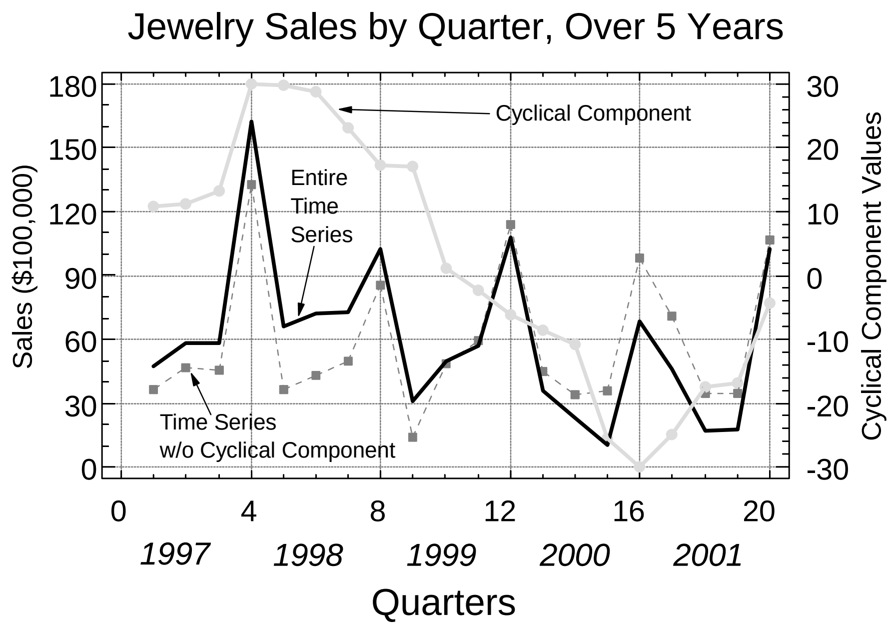
$$X_t = (T_t + I_t)S_t$$

Such models are often called **unobserved components models,** because in practice, although we observe the actual values, X_t, the individual components themselves will not be observed. Much of your concern with these time series data will be to learn what you can about each of the components, and to predict future values based on what you've learned about past behavior of components of the time series.

- **cyclical components.** Finally, just to insure that you don't think this topic is *too* simple, a fourth component, the cyclical component, may be present in a time series. Suppose, for example, that jewelry sales are also affected by business cycles, rising when times are good (full employment) and falling when times are bad (recession). The cyclical component is intended to capture periodic oscillatory behavior, other than that associated with seasonality. It is natural to require that anything characterized as a cycle exhibit some regularity and be of constant duration, like the seasonal component. Certainly this would be a great asset in attempting to estimate such an entity. Unfortunately, our experience with historical business cycles suggests that both their lengths and magnitudes, as measured by differences between peaks and troughs, are far from constant. Consequently, given data series of just a few periods length, it will often be very difficult to distinguish between a smoothly evolving trend and smooth business cycles. For this reason, it is often useful to view the component T_t as a combination of trend and cycle. We might call this combination the **trend-cycle component.**

So, the last chart shows the entire time series for our hypothetical jewelry sales example. It includes a major business cycle that peaked in the fourth quarter of 1997 (Quarter 4) and bottomed out in the fourth quarter of 2000 (Quarter 16). This cyclical component is shown as

 while the time series with and without the cyclical component are shown in the chart also. You'll notice that, in this instance, sales of jewelry are enhanced in 1997-98 because of the economic boom and depressed in 2000 and 2001 because of the recession. Because of the business cycle, the variability of quarterly sales over the five year period rises from $11.87 million to $15.21 million.



2 Managing Time Series Data

This exercise makes use of two kinds of time series data: (1) daily exchange rate data on the dollar value of the Turkish Lira from 24 October 1993 to 20 June 2002 (2,687 observations), and (2) data from the U.S. Department of Labor Bureau of Labor Statistics on the aggregate unemployment rate and the number of people working part-time for "economic" reasons (i.e., working part-time because they can't get full-time jobs). These data are monthly and cover the period from January, 1960 to June, 2002 (510 observations).

2.1. Appending Data Sets (see Hamilton "Combining Two or More Stata Files" pp. 43-48)

Often, for one reason or the other, we need to combine two or more data sets containing the same variables into one large data set, perhaps because the data come from two different volumes of a statistical handbook. The Turkish exchange rate data are stored in two separate data sets. You should download the following data sets from the course web site now.

Download:

Turkish Exchange Rate Data	
	Use these two data sets:
Intercooled Stata	tler1.dta (the first 1,500 observations) tler2.dta (the remaining 1,187 observations)

Data sets "tler1.dta" and "tler2.dta" have exactly the same variables, but "tler1.dta" contains observations before 3 December 1997 and "tler2.dta" contains observations after 2 December 1997. We need to join these two data sets together to get the entire series extending from 24 October 1993 to 20 June 2002.

We can do this two ways:

1. Load "tler1.dta" into Stata:

`use "\your path\tler1.dta", clear`

`describe`

```
Contains data from \your path\tler1.dta
  obs:         1,500
 vars:             3                          24 Jul 2002 20:25
 size:        36,000 (96.1% of memory free)
-------------------------------------------------------------------------------
              storage  display     value
variable name   type   format      label      variable label
-------------------------------------------------------------------------------
date            long   %d
tl              double %10.0g
tenmilli        double %10.0g
-------------------------------------------------------------------------------
Sorted by: date
```

(Always make sure that the data are sorted by "date").

`sort date`

then, execute the following command to add the later observations to "tler1.dta":

`append using "\your path\tler2.dta"`

Now, save the combined data set under a *new name:*

`save d:\your path\tler`

2. "Describe" your combined data set and paste the results into your exercise.

`describe`

```
Contains data from D:\your path\tler.dta
  obs:         2,687
 vars:             3                          24 Jul 2002 20:25
```

```
 size:        64,488 (93.3% of memory free)
-------------------------------------------------------------------------------
              storage  display     value
variable name   type   format      label      variable label
-------------------------------------------------------------------------------
date            long   %d
tl              double %10.0g
tenmilli        double %10.0g
-------------------------------------------------------------------------------
Sorted by:  date
```

You can "list" the data (or call up the Browse Window) to see what you've got, but if you appended "tler2.dta" to "tler1.dta" you should have the variables in the right order.

Now, you've got the complete time series in one data set. We'll do some analysis in a minute.

Of course, you could "append" the data sets in reverse order. Then you must be sure to sort the combined data on the variable "date", otherwise your time series analysis will look pretty weird (See, Hamilton, "Combining Two or More Stata Files").

2.2. Merging two Stata Data Sets (see Hamilton "Combining Two or More Stata Files")

The employment and unemployment data have a different problem: They come from different sources and, although they cover the same time period, they're in different data sets. Consequently, in order to get the part-time employment data and the aggregate unemployment data into the same data set, we have to "merge" the two data sets. Here we're combining data observation by observation; i.e., we want to put all the data from January 1960 in the same observation so that each observation of the new data set will contain all the variables for that month/year.

Download from the course web site the data sets: "uerate.dta" and "pttwork1.dta".

```
-------------------------------------------------------------------------------
. use "D:\your path\uerate.dta", clear

. describe

Contains data from D:\your path\uerate.dta
  obs:           510 (max=       1,130)
 vars:             3 (max=          99)
width:            28 (max=         200)
-------------------------------------------------------------------------------
              storage  display     value
variable name   type   format      label      variable label
-------------------------------------------------------------------------------
yrmo            str12  %12s
ueru            double %12.0g
uersa           double %12.0g
-------------------------------------------------------------------------------
Sorted by:

. list in 1/5

             yrmo          ueru         uersa
  1.       1960 1           6.1           5.2
  2.       1960 2           5.7           4.8
  3.       1960 3           6.1           5.4
  4.       1960 4           5.2           5.2
  5.       1960 5           4.8           5.1
```

Note that file "uerate.dta" contains three variables: yrmo (a string variable containing year-month), ueru (a floating point variable containing the percentage unemployment rate, *not* seasonally adjusted), and uersa (the official seasonally adjusted unemployment rate). Now, load the data set containing the monthly number of part-time workers (for economic reasons).

```
. use "D:\your path\pttwork1.dta", clear

. describe

Contains data from D:\your path\pttwork1.dta
  obs:           510 (max=       1,130)
 vars:             2 (max=          99)
width:            20 (max=         200)
---------------------------------------------------------------------------
              storage  display     value
variable name   type   format      label      variable label
---------------------------------------------------------------------------
yrmo            str12  %12s                   Year Month
pttwork         double %12.0g
---------------------------------------------------------------------------
Sorted by:

. list in 1/5

           yrmo        pttwork
  1.      1960 1          2522
  2.      1960 2          2530
  3.      1960 3          2304
  4.      1960 4          2711
  5.      1960 5          2579
```

Note that "pttwork1.dta" contains two variables "yrmo" (a string variable containing year-month), and "pttwork" (a numerical variable containing the monthly number (in thousands) of part-time workers for economic reasons).

Both data sets contain 510 observations beginning in January, 1960 and ending in June, 2002 (verify this to make sure it's true) and we want to merge the two data sets so that the resulting data set would contain all the variables from both data sets, organized by year-month. That is, the observation would look like this:

```
             yrmo          ueru         uersa      pttwork
  1.      1960 1           6.1           5.2         2522
 etc.
```

So, we've got to find a way to combine both data sets on an observation-by-observation basis. It's not hard to do, but, first, we have to solve a problem relating to the computer representation of dates in time series analysis.

2.3. Dates (Digression)

Time series data relate to specific points in time. The exchange rate data are *daily* data, being presented for each day of the week, while the employment and unemployment data are *monthly* data. This presents problems as Stata needs to work with dates as *numeric* variables, but numeric variables are very inconvenient to work with on a graph. Stata has extensive facilities to deal with dates and, specifically, dates in a time series context.[1] We'll deal with just a few of them to give you an idea of the power of Stata's date facility.

[1] See, L.C. Hamilton, 2004, *Statistics with Stata 8,* Belmont, CA: Duxbury, pp 339-346, and Stata Corp., 2001, *Stata Statistical Software, Release 7.0: User's Guide,* Chapter 27: Commands for Dealing with Dates.

2.3.1 Stata's Basic Date Style

Raw data can contain dates in all sorts of formats; however, there is one format that Stata understands, called elapsed dates or **%d** dates. A %d date is the number of days from January 1, 1960. In this format,

0 corresponds to 01jan1960	-1 corresponds to 31dec1959
1 corresponds to 02jan 1960	-2 corresponds to 30dec 1959
15,000 corresponds to 25jan2001	-15,000 corresponds to 01dec1918

etc.

So, no matter how a date appears in your raw data, it needs to be converted to this numerical format to be useful in time series analysis. Once in this format, the date can be formatted for visual reference (e.g., 2002/jan) while retaining its basic underlying value. This means that dates can be manipulated as numbers even though they might appear to be string values like "2002/jan."

2.3.2 Turkish Data

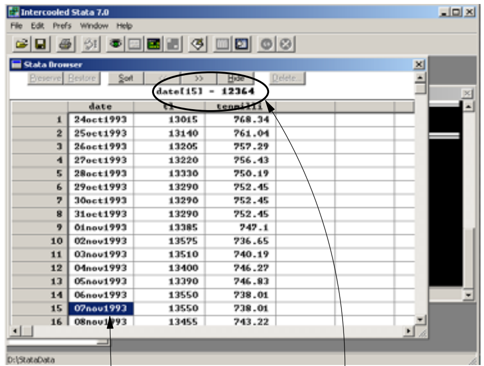
- Load the Turkish Lira times series data into Stata and describe and list the first 15 observations of the data:

```
. use "D:\amifiles\Econ70\exercises\exer_x\turkey\tler.dta", clear

. describe

Contains data from D:\amifiles\Econ70\exercises\exer_x\turkey\tler.dta
  obs:         2,687
 vars:             3                          25 Jul 2002 15:13
 size:        64,488 (93.6% of memory free)
---------------------------------------------------------------------------
> ----
              storage  display     value
variable name   type   format      label      variable label
---------------------------------------------------------------------------
> ----
date            long   %d
tl              double %10.0g
tenmilli        double %10.0g
---------------------------------------------------------------------------
> ----
Sorted by:  date

. list in 1/15

          date         tl    tenmilli
  1. 24oct1993      13015      768.34
  2. 25oct1993      13140      761.04
  3. 26oct1993      13205      757.29
  4. 27oct1993      13220      756.43
  5. 28oct1993      13330      750.19
  6. 29oct1993      13290      752.45
  7. 30oct1993      13290      752.45
  8. 31oct1993      13290      752.45
  9. 01nov1993      13385       747.1
 10. 02nov1993      13575      736.65
 11. 03nov1993      13510      740.19
 12. 04nov1993      13400      746.27
 13. 05nov1993      13390      746.83
 14. 06nov1993      13550      738.01
 15. 07nov1993      13550      738.01
```

"Date" is stored as an integer.

Its display format is much easier to read!

If you want to see the actual number underlying "07nov1993", just *Browse* the data and find that observation:



Here's the formatted date

Here's the underlying raw integer

The Turkish Lira data have a date variable, "date" that is already in the appropriate format for working with time series data, so we could just continue with our analysis; however, Stata has a feature that is particularly useful when working with time series data. We can declare the entire data set a time series data set by using the command:

```
tsset date,daily
```

This command declares the data a time series, designates that "date" is the variable in the data set representing time and specifies that these data are *daily* data, as opposed to annual, monthly or quarterly data. Issuing this command also sorts the data by "date" so that subsequent time series operations on the data will function properly.

- So, now issue the command

```
tsset date,daily
```

and save the "tler" data.

2.3.3 Employment/Unemployment Data

Now, back to the two data sets containing part-time employment and unemployment respectively. Load the "pttwork1.dta" data set.

- "describe" the data (paste into your exercise)
- "list" the first 15 cases in the data set (paste into your exercise)
- create a new "date" variable that is numeric using:

```
gen date=monthly(yrmo, "ym")
```

- change the data set to a time series data set using:

```
tsset date, monthly
```

- give the date variable a more attractive format using:

`format date %tmCy/l` *(note this last "l" is a lower case "L")*

- "describe" and "list" the first 15 cases and paste into your exercise

- save the data set as "pttwork1a.dta"

Now, go through the exact same procedure with the "uerate" data, pasting into your exercise as above. Save the altered data set as "uerate1a.dta"

2.4. Completing Merge of the part-time and unemployment data sets

Now our data are ready to merge into one data set.

- Load "uerate1a" and "describe" it to make sure the data are sorted by "date."

- Merge the data sets by using the following command: (see, Hamilton, pp. 46- 49)

```
merge 1:1 date using "D:\your path\pttwork1a".
```

- Save the combined data set with a new name: "ptt_uer.dta"

- "describe" and "list" the first 15 observations. Your output should look something like this. Note the following features of this combined data set:

```
. describe

Contains data from D:\your path\ptt_uer.dta
  obs:           510
 vars:             4                          11 Apr 2014 22:43
 size:        16,320
-------------------------------------------------------------------------------
              storage   display    value
variable name   type    format     label      variable label
-------------------------------------------------------------------------------
yrmo            str12   %12s                  "yrmo" no longer needed
ueru            double  %12.0g
uersa           double  %12.0g         The variables don't have informative labels
date            float   %tmCy/l
_merge          byte    %8.0g
-------------------------------------------------------------------------------
Sorted by:  date              What is this "_merge" variable?

. list in 1/15

     +---------------------------------------------------------------+
     |    yrmo   ueru   uersa       date   pttwork          _merge |
     |---------------------------------------------------------------|
  1. |  1960 1    6.1     5.2   1960/jan      2522     matched (3) |
  2. |  1960 2    5.7     4.8   1960/feb      2530     matched (3) |
  3. |  1960 3    6.1     5.4   1960/mar      2304     matched (3) |
  4. |  1960 4    5.2     5.2   1960/apr      2711     matched (3) |
  5. |  1960 5    4.8     5.1   1960/may      2579     matched (3) |
     |---------------------------------------------------------------|
  6. |  1960 6    5.8     5.4   1960/jun      3304     matched (3) |
  7. |  1960 7    5.5     5.5   1960/jul      3215     matched (3) |
  8. |  1960 8    5.2     5.6   1960/aug      3360     matched (3) |
  9. |  1960 9    4.7     5.5   1960/sep      2825     matched (3) |
 10. | 1960 10      5     6.1   1960/oct      2751     matched (3) |
     |---------------------------------------------------------------|
 11. | 1960 11    5.6     6.1   1960/nov      3013     matched (3) |
 12. | 1960 12    6.4     6.6   1960/dec      3148     matched (3) |
 13. |  1961 1    7.7     6.6   1961/jan      3406     matched (3) |
 14. |  1961 2    8.1     6.9   1961/feb      3477     matched (3) |
 15. |  1961 3    7.7     6.9   1961/mar      3159     matched (3) |
     +---------------------------------------------------------------+
```

- Let's redefine the combined data set as a time series data set sorted on "date":

```
tsset date, monthly
```

- The "_merge" variable is provided by Stata to help you insure that each observation was merged successfully (See Hamilton, p. 48.). "_merge" takes on the following values:

  1 = observation from the master dataset ("uerate1a", in this case) only.
  2 = observation from the using datset only ("pttwork1a", in this case) only.
  3 = observation from both master and using data.

- Stata actually has put in the log a table showing that all 510 cases contain observations from both data sets (browse the data if you want to convince yourself even further).

```
Result                           # of obs.
 -----------------------------------------
 not matched                             0
 matched                               510  (_merge==3)
 -----------------------------------------
```

- Once satisfied, we can drop the superfluous "yrmo" variable and the "_merge" variable:

```
drop  yrmo _merge
```

- Now, let's add some labels to the variables:

```
label variable pttwork "Part-time workers (‘000s, not SA)"
label variable ueru "Unemp. Rate  (not SA)"
label variable uersa "Unemp. Rate  (SA)"
label variable date  "Year/Month"
```

The combined data set now contains four variables: two unemployment rates, one seasonally adjusted and one not seasonally adjusted; number of part-time workers each month (not seasonally adjusted); and a numerical date variable, formatted to be easily readable in output.

- Save your spruced-up data set, describe it and list the first 15 cases. Paste the "describe" and "list" results into your exercise.

Now, finally, we have two data sets, "ptt_uer.dta" and "tler.dta" on which we can perform some time series analysis.

3 Turkish Lira Exchange Rates

Load the Turkish Lira exchange rate data ("tler.dta") into Stata.

"Describe" the data set. Oops! you'll note that I forgot to label these variables, so let's do it now:

```
label variable  tl "Turkish Lira per U.S. Dollar"
label variable   tenmilli  "U.S. Dollars per TL 10 million"
label variable date "Date"
```

Save the data set.

Notice that the data set contains two versions of the Turkish Lira/U.S. Dollar exchange rate: TL/$ (tl) and $/TL10 million (tenmilli). We use TL10 million because of the vast differences in accounting magnitudes for the two currencies. (Actually, I carry a TL10 million note in my wallet just to get a sense of what being *really* rich might feel like!)[2]

- Graph "tenmilli" vs. "date" just to get an idea of how the exchange rate has varied between fall, 1993 and summer 2002:

```
graph twoway line tenmilli date
```

- Paste the graph into your exercise and write a sentence or two describing what's happened to the dollar value of TL10 million since October, 1993.

- Graphing "tl" versus "date" shows you what's happened to the number of Turkish lira needed to buy one dollar since 1993 (paste the following graph into your exercise):

```
graph twoway line  tl date
```

- On 25 October 1993 it required TL13,015 to buy one Dollar; On 20 June 2002 it required TL1,560,002.

- Since "tl" and "tenmilli" are roughly reciprocals of each other, they're measuring the same thing. We can graph both variables together; however, we'll have to measure them on different axes since their magnitudes are so different. We'll add a right vertical scale to the graph and measure the value of "tl" on that, rescaling the right axis[3] so all the values of both variables will fit on the same graph. In addition, we'll spiff up the graph a bit with some techniques you've learned earlier:

```
graph twoway line tenmilli date, yaxis(1) ylabel(0 (200) 800, angle(horizontal) nogrid axis(1))
ytitle("U.S. Dollars per 10 million TL" " ", axis(1)) || line tl date, yaxis(2) ytitle("TL per
One U.S. Dollar", axis(2)) ylabel(0 (500000) 1500000, angle(horizontal) axis(2)) || ,
title("Turkish Lira vs. U.S. Dollar: 1993-2002") xtitle(" " "Date") xlabel(12350 (750) 15511,
grid)
```

---

[2] The Turks revalued the Turkish Lira in 2004, lopping many zeros off the Lira and making my TL 10 thousand bill a historical curiosity.

[3] See Hamilton, "Further Time Plot Examples" p. 348 and the Stata Graphics Manual for information for information on how to plot two time series using two different y-axis scales.

- Paste this graph into your exercise.

- We're looking at a long time period here. The graphs show fairly obvious and smooth trends; however, the day-to-day fluctuations are not necessarily minor. To see this, graph just a small segment of the data so you can see the fluctuations more clearly. Use the "in" qualifier (see Hamilton "Specifying Subsets of the Data: in and if Qualifiers", pp. 20-24) to choose a subset of observations. We'll use the time period 01June1995 to 05August1995. Look up the observation numbers by "browsing" the data. You would execute this command:

```
graph twoway line  tenmilli date in 586/651
```

Now the graph shows the day-to-day fluctuations more clearly.

The exchange rate data don't appear to have any obvious seasonality; there doesn't appear to be any cyclical behavior going on, either. So, let's see if we can't smooth out the daily fluctuations by isolating the trend-cycle. We'll use two different Stata commands to smooth the data.

3.1. *Moving Average* *(Hamilton "Smoothing" pp. 341- 345)*

Remember, the purpose of smoothing data is to eliminate details that obscure underlying trends and/or cycles. A simple and time-honored method of smoothing a time series is to create a new version of the series using a **moving average.** Instead of plotting a single value of the series for each date, we average the series values around the date in question and plot their *average* rather than simply the single variable.

- We could do this by "generating" a new time series by using the implicit subscripts of the series in question: Try this (Note: "_n" is the current observation number):

```
generate tenmilli2 = (tenmilli[_n-1] + tenmilli[_n] + tenmilli[_n+1])/3
```

- This gives us a smoothed series for "tenmilli", which is the average of the three exchange rate values around the target date for each observation. Plot the two time series as follows to see what you get:

```
graph twoway line  tenmilli date, || line  tenmilli2 date
```

Note two things: (1) the smoothed series "tenmilli2" has two missing values (Browse the data) because we couldn't compute values for the two endpoints of the series; (2) because the series is so long and the daily fluctuations are minor, the graph doesn't show much of a difference between the original and the smoothed series. However, if you repeat the "graph" command for the s*ubset* of data we just looked at, you see something more informative:

```
graph twoway line  tenmilli date in 586/651, || line  tenmilli2
date in 586/651
```

- Let's smooth the series a little bit more by taking a moving average of 5 data points. However, this time, we'll use the "extensions to generate" (egen) command to compute the moving average:

```
egen tenmilli3 = ma(tenmilli), nomiss t(5)
```

- Also, let's label the two new time series:

```
label variable  tenmilli2 "TL10 million moving avg. (t=3)"
label variable  tenmilli3 "TL10 million moving avg. (t=5)"
```

- Now graph the original and two smoothed series for the same restricted time period that you used above. Paste the graph into your exercise. Notice that the larger the number of values you include in the moving average, the smoother the resulting smoothed time series. (Stata plots the 3 curves in different colors, which is fine for your screen or a color inkjet printer, but would have to be modified if you wanted to stick to black and white printing.)

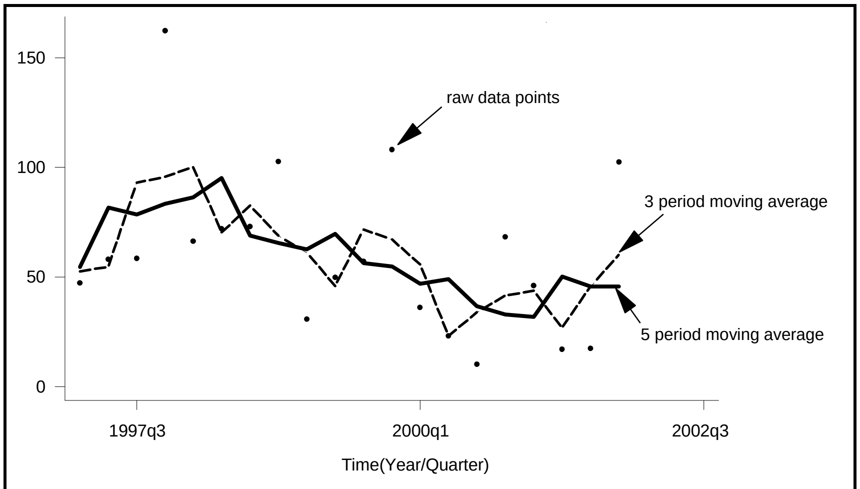
```
graph twoway line  tenmilli date in 586/651, || line  tenmilli2
date in 586/651|| line  tenmilli3 date in 586/651
```

The Turkish Lira exchange rate data, while showing a very strong downward trend in the dollar value of TL10 million (or upward trend in the number of TL required to buy one U.S. dollar), don't have a very large irregular component, nor do they exhibit much in the way of seasonality or cyclicity. So let's turn next to the part-time employment and aggregate unemployment data.

4 Part-time Employment & Unemployment Data

4.1. "Smooth": Outlier-resistant nonlinear smoothing

Moving averages share a drawback with other mean-based statistics in that they have little resistance to outliers. Take, for example, the jewelry sales example at the beginning of this exercise. We saw that the fourth quarter of each year shows a major jump over surrounding quarters. Let's see what impact this has on attempts to smooth the data using moving average smoothers of 3 and 5 periods:



3&5 period moving average of jewelry sales data

Again, the larger number of terms in the moving average, the smoother the smoothed series; however, note that the presence of those large fourth quart sales spurts still makes the smoothed series rather jerky. Let's try another method that is not as affected by large outliers as is the moving average method. This method, due to statistician John W. Tukey,[4] computes *medians* of blocks of data very similarly to the way moving averages are computed:

> For each observation, *t*, of the time series (except for the beginning and ending points), take (e.g., for a three observation rolling median) the three values of the series, x_{t-1}, x_t, x_{t+1} , rearrange the three observations from lowest to highest and pick the middle observation as the median value. Notice that large outliers like those in the jewelry data cease to have any input in the calculation of the smoothed series.
>
> This produces a new time series made up of median values. You can do this again and again until the series stops changing (as it eventually will since we're computing medians). Plot this series as the smoothed series.[5]

This method of **robust nonlinear smoothing** can be used over and over to produce a smoothed series. Stata makes this easy by providing a command "smooth" that allows repeated smoothing with just one command. Here's the command:

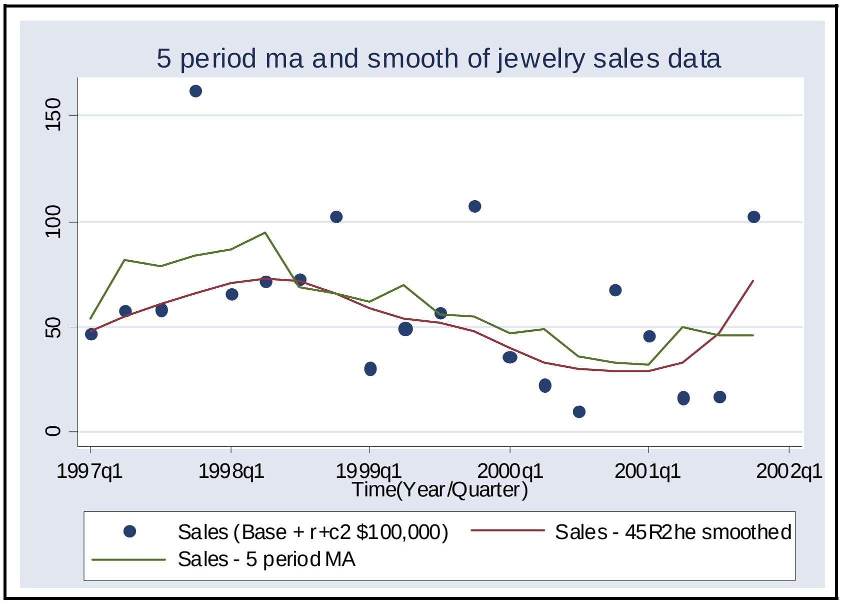
```
smooth 45R2eh,twice  jrc2, gen(jrc2sm2)
```

and here's the smoothed series:

```
graph twoway scatter  jrc2 quarters ||  line  jrc2sm2 quarters ||
line  jrc2ma5 quarters || , ti("5 period ma and smooth of jewelry
```

[4] John W. Tukey, *Exploratory Data Analysis*, Reading, MA: Addison-Wesley, 1977: chapter 7.
[5] Hamilton gives a *very* brief discussion of this procedure on p. 281. See also *Stata Reference Manual:* "smooth" and Tukey, *op. cit.*

```
sales data")
```



Note how the robust smoother completely eliminates the seasonality of this time series, as compared to the 5-period moving average. Note also, that this smoother captures the business cycle (last graph in Section 1, above) which dominates the mildly rising trend seen in the third graph of Section 1. Now, let's try smoothing the employment data.

4.2. Smoothing the employment and unemployment data

4.2.1 The Unemployment Rate

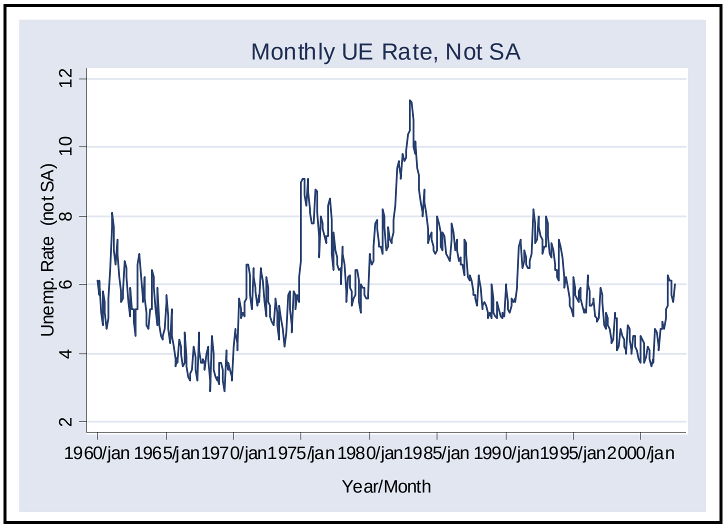
- Load the "ptt_uer.dta" data set.

- Graph the non-seasonally adjusted unemployment data.

```
graph twoway line  ueru date, xtitle(" " "Year/Month") xlabel(0
60 to 504) ti("Monthly UE Rate, Not SA")
```

This command specifies labels for the x-axis by starting at "0" (Jan/1960) and then providing a label for January every 5 years. (Note that we use the underlying numerical values for "date", not the formatted 'year/month' dates shown on the x-axis.)

- Paste this graph into your exercise. It should look like the following:



These unemployment data suggest some long-term cycles as well as some seasonal variation in unemployment each year. Let's see if we can get rid of the seasonal data and view how the seasonally adjusted unemployment rate has changed over time.

- Try the following command to generate a smoothed unemployment rate time series:

```
smooth 9R7r5r3ssrhe,twice ueru,generate(smoo3)
```

Let's interpret this command to see what you've done: The string '9R7r5r3ssrhe' performs the following series of smoothings:

- ✓ '9R ...' Starting at the beginning of the series take 9 observations at a time, sort them from smallest to largest and pick the median (middle) value to create a new value in the smoothed series, phase 1. The 'R' means "keep doing this until the series doesn't change any more." I chose a block of 9 observations because the seasonality in unemployment likely extends over a whole year and 9 months is the longest group that "smooth" allows.
- ✓ '7r ...' Take the partially smoothed series from phase 1 and smooth it again using 7 observations at a time, as above. 'r' (you can use either lower case or capital letter, it doesn't matter) means keep refining the series using 7 observations until it has no more effect
- ✓ '5r ...' Now, do the same thing with a set of 5 observations
- ✓ '3 ...' Now, do it one last time with a set of three observations
- ✓ 'ssr ...' Apply the 'ss' smoother 'r' times to smooth out the peaks and valleys of the smoothed series
- ✓ 'h' is the Hanning linear smoother. It's used to help the end points of the series approximate the original data better
- ✓ 'e' the "end-point rule" is another correction to the end-points of the smoothed series so they better approximate the end points of the original time series.
- ✓ ',twice' (Note that there's no space between the comma and 'twice') A smoother divides the data into a smooth and a rough: $data = smooth + rough$. If the smoothing is successful, the rough should exhibit no pattern. Twicing refers to applying the smoother to the observed, calculating the rough and then applying the smoother to the rough. The resulting "smoothed rough" is then added

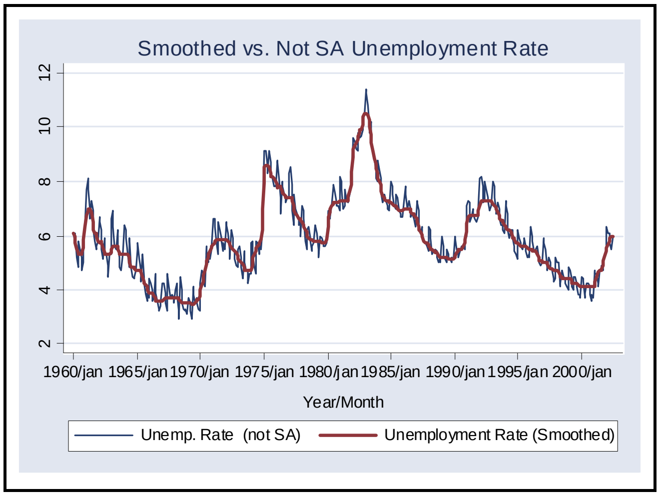
back to the smooth from the first step.

Let's stop for a moment: How much of this stuff can I honestly expect you to remember? Just enough for you to be able to look up this command and play around with it to get a smoothed series that meets your needs. Smoothing a time series involves a lot of trial and error; there's nothing like experimentation.

- So, let's compare the smoothed series with the non-seasonally adjusted series. Run this command:

```
graph twoway line  ueru date || line  smoo3 date, clwidth(thick)
|| ,  xtitle(" " "Year/Month") xlabel(0 60 to 504) ti("Smoothed
vs. Not SA Unemployment Rate")
```

You should get a graph that looks like this:



- Paste the graph into your exercise. How well does the smoothed series seasonally adjust the unemployment rate series?

- Now, graph your smoothed series against the Bureau of Labor Statistics' seasonally adjusted series ("uersa" represented by dots) using the following command:

```
graph twoway scatter  uersa date, msymbol(oh) || line  smoo3
date, clwidth(medium) || , ti("BLS SA UE Rate vs. Our Smoothed
Rate")  xtitle( " " "Year/Month")
```

- Paste the resulting graph into your exercise and write a short paragraph telling me what this graph tells you about how well the two series match. (Note: The BLS computes its version of the seasonally adjusted unemployment rate by seasonally adjusting twelve different unemployment series from different sectors of the economy and adding the twelve seasonally adjusted series together to achieve the seasonally adjusted aggregate series you've just plotted.)

4.2.2 Part-time workers for "economic" reasons

The time series "pttwork" shows the number of people (in '000s) in each month from 1960/jan to 2002/jun who are working part-time for "economic" reasons. By this, the BLS means workers who would just as soon be working full time but cannot because they can't find full time jobs. Below, see a page from the *Statistical Abstract, 2000* relating to this time series.

No. 665. Part-Time Workers by Reason: 1999

[In thousands (30,913 represents 30,913,000), except hours. For persons working 1 to 34 hours per per week. For civilian noninstitutional population 16 years old and over. Annual average of monthly figures. Based on the Current Population Survey and subject to sampling error; see text, Section 1, Population, and Appendix III]

Reason	All industries			Nonagricultural industries		
		Usually work—			Usually work—	
	Total	Full time	Part time	Total	Full time	Part time
Total working fewer than 35 hours	**30,913**	**10,079**	**20,834**	**30,000**	**9,807**	**20,193**
Economic reasons	3,357	1,281	2,076	3,189	1,193	1,996
Slack work or business conditions	1,968	1,021	947	1,861	962	899
Could find only part-time work	1,079	-	1,079	1,056	-	1,056
Seasonal work	147	97	50	115	74	41
Job started or ended during the week	162	162	-	157	157	-
Noneconomic reasons	27,556	8,798	18,758	26,811	8,614	18,197
Child-care problems	856	86	770	843	84	759
Other family or personal obligations	5,629	746	4,882	5,476	727	4,749
Health or medical limitations	712	-	712	674	-	674
In school or training	6,463	100	6,363	6,320	97	6,223
Retired or Social Security limit on earnings	1,984	-	1,984	1,863	-	1,863
Vacation or personal day	3,239	3,239	-	3,188	3,188	-
Holiday, legal, or religious	966	966	-	956	956	-
Weather related curtailment	824	824	-	781	781	-
Other	6,884	2,837	4,047	6,710	2,781	3,929
Average hours per week:						
Economic reasons	23.1	24.0	22.5	23.2	24.1	22.6
Noneconomic reasons	21.5	25.7	19.6	21.6	25.8	19.6

\- Represents or rounds to zero.

Source: U.S. Bureau of Labor Statistics, *Employment and Earnings,* monthly, January 2000 issue.

414 Labor Force, Employment, and Earnings

U.S. Census Bureau, Statistical Abstract of the United States: 2000

- Compute the average number of part-time workers during 1999 from the "pttwork" data series. (Do this in Stata and paste the command(s) you used and the results into your exercise.) Does your result equal the published result in the table above? Why/Why not? (It's OK to speculate here!) In percentage terms by how much do the two numbers differ?

- Graph the "pttwork" series and paste it into your exercise. Pretty up the axes and add a title. Examine the graph of the time series. Does it appear that there's seasonality in the series? There may be too much data to see evidence of seasonality clearly, so,

- Graph the "pttwork" series from 1999jan to 2002jun, adjusting the axes so that you can get a more detailed picture of the monthly ups and downs. Paste the graph into your exercise. Does it look like we've got regular seasonal rises and falls in the number of economically-part-time employed? If so, identify the months when the number of economic-part-time workers is highest & lowest and explain why economic-part-time is lowest or highest in each of these months.

- For the entire period, produce a smoothed time series for pttwork that seasonally adjusts the data and create a new variable “pttwsmoo” that contains the smoothed data. Paste the Stata command that you used to smooth the data into your exercise. Graph the smoothed time series and paste it into your exercise.

- Write a short paragraph after the graph in which you answer these questions: (1) Were you successful in getting rid of seasonality? If not, why do you think not? (2) Does there look to be a steady long-term trend in the smoothed part-time work data? (3) Does there appear to be cyclical movement in the smoothed data? If so, what do you think could be causing it?

- Now, graph the two smoothed series in your data, “smoo3” (The smoothed unemployment rate) and “pttwsmoo” (The smoothed part-time work [economic reasons] series) on the same graph. Note, these two series have widely differing magnitudes so you’ll have to use two vertical axes and rescale one in order to successfully graph the two series together. Look back in the exercise for ideas on how to do this. Paste both the command you used to produce the graph and the graph itself into your exercise.

- What does the juxtaposition of the smoothed aggregate unemployment data and the smoothed part-time [economic] tell you about the reason that the part-time employment data seem to be following a cyclical path?

5 Last Things

- Save your modified data set, close down your log, and save your exercise.

- Check over your exercise to be sure that every thing’s there that I requested.

- Go treat yourself to a beverage of your choice.