STATA Commands in s:\703 Kaufman\TS_Example_sil.do
s:\fac\rlk\703\TS_Example_srl.do

COMMENTS AND ANNOTATIONS

Remember that you must have loaded (automatically by myprofile.do or otherwise) the programs from ***703 do-files.do*** to be able to use the *DURBWATS* and *BUSETIME* programs written for class.

```
**  COMMENT  Time Series Example: ALL Variables are Natural Logs
log using x:\TS_example_out.log, replace
```

```
infile year density unemp young burgper burghld durexp autocap  ///
    using "s:\703 Kaufman\tsexmp.dat"
```

This **infile** shows how to read in data that are in free-format: each value for a case is separated by one or more spaces. You always need a variable indexing time in your data set for a time series analysis. In this case we naturally use a variable named ***year***, but it could have any name.

```
tsset year, yearly
global ivars = "young density"
```

You must **tsset** your data which reformats it as a times series structure in STATA. Specify the name of the variable indexing time (**year** in this case) and use options to indicate the periodicity of the data (**, yearly** in this case). **gloval ivars** sets up a nickname for your independent variables.

```
**  COMMENT Example 1: Autos/capita on age and density
regress autocap $ivars
predict resols, res
```

This is an OLS regression of ***autocap*** on ***young*** and ***density***. The **predict** command specifies ***resols*** to name the variable containing the residuals or choose your own name and change below.

```
scatter resols year, connect(l) yline(0)  name(Autos_per_capita_residuals)
corrgram resols, lags(5)
estat dwatson
```

scatter graphs residuals from ***resols*** against the variable ***year***. **connect(1)** connects the points on the plot and **yline(0)** draws a horizontal bar at zero on the y-axis.
corrgram is used to get the ACF and PACF based on the values in ***resols***, the OLS residuals.
lags(#) specifies how many lags to calcualte for the ACF and PACF, in this case **5**.
estat is used to get the Durbin-Watson statistics for the OLS model

```
**  COMMENT  Estimate as AR(1)
prais autocap $ivars, rho(dw)
glsbeta autocap
busetime autocap year 1946
```

prais specifies an AR(1) time-series model estimated using the Prais-Winsten method.
glsbeta calculates the standardized coefficients for the preceding model.
busetime calculates Buse's R^2 for the last model. You must specify the name of your outcome variable (**autocap**), the name of your time index variable (**year**) and the value of the time index variable for the first time point in the sample you are analyzing (**1946**).

```
**  COMMENT Example 2: Burglary rate on durable expenditure  age  & density
global ivars = "durexp young unemp"
regress burghld  $ivars
drop resols
predict resols, res
```

For this second example, reset **ivar**, the nickname for independent variables, to contain ***durexp***, ***young***, and ***unemp*** .

The OLS residuals are saved in ***resols*** to be used in **scatter** and **corrgram** but you must first **drop** this variable from the data set before it is recalculated.

```
scatter resols  year, connect(l) yline(0)  name(Burglary_per_household_residuals)
corrgram resols, lags(5)
estat dwatson
```

scatter plots residuals from ***resols*** against the variable ***year***
corrgram is used to get the ACF and PACF based on the values in ***resols***, the OLS residuals.
estat again used to get the Durbin-Watson statistics for the OLS model

```
** COMMENT Estimate as AR(1). Get ACF/PACF for transformed residuals
prais burghld $ivars, rho(dw)
durbwats 1.650 year
predict resar1, res
```

Again use **prais** to estimate an AR(1) model using the Prais-Winsten method. Note that the results suggest continued problems with autocorrelation.
durbwats estimates the exact Durbin-Watson critical value. ***1.650*** should be replaced with D_u the upper limit critical value for your sample size and number of independent variables. This is D_u for n=30 and k'=3. Also specify the name of your time index variable (**year**).

```
gen res2=resar1-e(rho)*resar1[_n-1]
replace res2=(1-e(rho)^2)^.5*resar1 if year == 1946

corrgram res2, lags(5)
```

gen calculates the transformed residuals as **res2**. **e(rho)** is the value of rho from the prior AR(1) model. Note the transformation is different for the first year (**1946**) than other years. The function **resar1[_n-1]** specifies the value from **resar1** lagged by one year.
corrgram is used to get the ACF and PACF based on the values in **res2** (EGLS transformed residuals). This can help show if the autocorrelation has been properly modeled.

```
** COMMENT Try lagging all the X's (Excludes 1st year of data)
gen unemp1=unemp[_n-1]
gen young1=young[_n-1]
gen durexp1=durexp[_n-1]
drop if year==1946
```

The **gen** commands calculates lagged independent variables and saves them into new variables.

drop eliminates the first case from the analysis. **Caution:** *create the lagged variables before excluding cases with the* **drop** *command or you will get missing values for your new first case*.

```
global ivars = "durexp1 young1 unemp1"
prais burghld $ivars, rho(dw)
durbwats 1.650 year
drop resar1 res2
predict resar1, res
```

Note the **ivar** nickname is reset to specify lagged variables: ***durexp1***, ***young1*** and ***unemp1***
prais re-estimates the AR(1) model with the lagged predictors
durbwats and then **corrgram** are used for diagnostics. On the **durbwats** command ***1.650*** is D_u for n=29 and k'=3. This is not an error; it is the same as for n=30 and k'=3.

```
gen res2=resar1-e(rho)*resar1[_n-1]
replace res2=(1-e(rho)^2)^.5*resar1 if year == 1947
corrgram res2, lags(5)
```

Note the first year in the analysis is now ***1947*** and its residual has a different transformation than that applied to other years.

```
** COMMENT Try lagging just Durexp
global ivars = "durexp1 young unemp"
prais burghld $ivars, rho(dw)
durbwats 1.650 year
drop resar1 res2
predict resar1, res
```

Now only one lagged predictor is used so the contents of the **ivar** nickname are redefined. Prais-Winsten estimation is then done
durbwats again gets the Durbin-Watson approximated exact critical value.

```
gen res2=resar1-e(rho)*resar1[_n-1]
replace res2=(1-e(rho)^2)^.5*resar1 if year == 1947
corrgram res2, lags(5)
```

Then **corrgram** is applied to the transformed EGLS residuals. Again note that the first year (***1947***) is transformed differently.
All the results suggest success in modeling the autocorrelation now.

```
** COMMENT Calculate R square Y & Y_hat, Buse's R square, Standardized coefficients
busetime burghld year 1947
glsbeta burghld
```

bustime getsthe Buse's R^2 for the last model. Note that the first year in the sample is now ***1947*** because we excluded 1946 when we lagged durexp.
glsbeta calculates the standardized coefficients for the preceding model.
From this point down the commands illustrate a few points ands would not typically be run in a

```
**  COMMENT Calculate Corrected OLS Var(B) & Display Corrected results
sca  rho=1-e(dw_0)/2
sca  rhosqt=(1-rho^2)^.5
sca  nr1=e(N)-1
mat  t1=I(e(N))
mat  t2=rho*I(nr1)
mat  t3=[t2, J(nr1,1,0)]
mat  t4=[J(1,e(N),0)\t3]
mat  omesq=t1-t4
mat  omesq[1,1]=rhosqt
mat  omeinv=omesq'*omesq
mat  ome=invsym(omeinv)
quiet  regress  burghld  $ivars
drop  resols
quiet  predict  resols, res
mat  b=e(b)
mkmat  $ivars, mat(holdiv)
mat  iv=[holdiv,J(e(N),1,1)]
matr ix xpx= iv'*iv
matr ix xpxinv=invsym(xpx)
matr ix xpixomx = xpxinv*iv'*ome*iv
scalar  trce= trace(ome) - trace(xpixomx)
matrix  accum ssres= resols
scalar  sighat= ssres[1,1]/trce
matrix  varols= sighat*xpixomx*xpxinv
matrix  colnames varols = $ivars _cons
matrix  rownames varols = $ivars _cons
ereturn  post b varols
display  "OLS coefficients and Omega Corrected OLS Var(b)"
ereturn  display
```

real analysis.

This set of commands first runs OLS to get the coefficients and then constructs and displays the Ω matrix from the estimate of ρ. It then calculates the corrected variance for the OLS coefficients and presents the OLS coefficients, their corrected standard errors and t-test results.

```
**  COMMENT  Calculate OLS Results using Newey-West Correction
newey burghld $ivars, lag (2)
```

This demonstrates the Newey-West correction. Note that you must specify how many lags are relevant for the autocorrelation using **lags(#)**. In this case I specified **lags(2)** given the initial diagnostics suggesting the possibility of a second order process.

```
**  COMMENT Try AR(2) model with lagging durexp
arima burghld $ivars, ar(1 2)
```

An alternative to lagging any of the predictors would be to try an AR(2) model. These estimate an AR(2) model with and without a lagged effect for doorbell expenditures (durbexp) and suggest that an AR(2) model without lagged predictors is a statistically viable alternative.

```
**  COMMENT Try AR(2) model without lagging durexp -- read data in again to get all cases
clear
infile year density unemp young burgper burghld durexp autocap ///
    using "s:\703 Kaufman\tsexmp.dat"
tsset year, yearly
arima burghld $durexp young unemp, ar(1 2)
```

Note the use of **clear** to clear the current data (which excludes 1947) and re-read in the full data.

This documentation file and the syntax/do-file are available from the Examples section of the class Web Page.