

STATA Workshop

In addition to the online help menu in Stata, there are numerous resources on the web to help you learn stata. I've posted links to a few good sites under the module titled "Stata Resources" in Canvas.

The stata windows.

- command (for interactive submission of commands)
- results
- variables
- review

Stata icons

- log (to create a file containing a log of session)
- do-editor (to create a "*.do" file containing list of commands to be executed)
- data editor (to view and/or edit stata data as spreadsheet)
- data browser (to browse, but not edit, stata data files)
- break (to stop execution of Stata commands)
- Stata viewer (to view log files, etc.)

Types of stata files.

- dta (Stata data sets)
- do (Stata files containing commands)
- smcl/log (log files containing Stata commands and results from execution)

1. Different ways to create a Stata data set

- a. Use data editor and manually enter
- b. Use data editor and cut/paste from excel
- c. Import using file/import/...
- d. set obs=100 to create empty data set with 100 observations and then create own variables.

2. Opening an existing Stata data set

- a. Interactive: file/open/...
- b. Command line: use "G:\ECO\evenwe\eco311\data\cps2016.dta", clear

After importing the data, creating new variables, and/or dropping observations, you may want to save the new data set as a Stata data set. You can save a Stata set interactively with "file/save as/...". Alternatively, you could add code in your program so that the Stata data set is saved every time you run the program. The following would create a stata data set name "eco671".

3. Saving a Stata data set.

- a. Interactive: file/save/....
- b. Command line: save "m:\cps2016.dta", replace

4. The command line versus do-files

You should practice using do-files to accumulate commands for your data creation and analysis. This makes it possible to accumulate all of your commands and make necessary changes as you work through your project. You can create the relevant commands using the drop down menus, but you should save the commands necessary for your analysis in a do file. You should save your results in a log file.

5. Creating and replacing variables.

Many mathematical functions are available. Type “help functions” on Stata command line for assistance.

To generate a new variable, use the gen command. Note: Stata is case sensitive. All commands are in lower case. I recommend making all your variables in lower case as well. If you wish to update a variable that has already been created, use the replace command.

Examples:

```
gen school2=school^2    /*create the square of school years*/
```

```
gen age50_=1 if age>=50 /*create a dummy variable for workers over age 50*/  
replace age50_=0 if age<50
```

```
gen age50_=(age>=50) /*creates a dummy equal to one whenever the condition in  
parentheses is true*/
```

```
drop if age>50          /*warning: missing values are treated as infinite*/  
drop if age==.          /*drop observations with missing values for age*/  
keep if age<=50
```

(Note: for comparison operators, must use double equal sign for equality: e.g.

```
drop if age==50 /*(NOT drop if age=50) */
```

```
drop if age==. /*drops all observations with “missing values” for age*/
```

```
drop if age~=44 /*drops all observations with age not equal to 44*/
```

(other logical operators >= and <=)

```
drop if age<=16 | age>=65 /*the “or” operator is | */  
drop if age>=55 & female==1 /*the “and” operator is “&” */
```

6. Viewing the data

- a. list
- b. list in 1/10
- c. list if age>50

7. Create sample statistics

- a. summarize
- b. tabstat
- c. corr
- d. table

8. Create statistics by groups

- a. bysort female: summarize age
- b. table female, c(m age p50 age)

9. Weights

- a. Many Stata commands can be adjusted for weights
 - i. pw=probability weights
 - ii. aw=analytical weights
 - iii. fw=final weights
- b. examples
 - i. summarize age [pw=finalwt]

10. Create graphs

- a. interactive
- b. create interactive and copy command to do-file for later use.

Exercise 1.

- a. Open the Stata data set at g:\eco\evenwe\eco311\ cps2015.dta
- b. Create a histogram of age
- c. Create hrwage (wlearn/wkhours)
- d. Compute the mean and median of age and hrwage
 - a. Without weights
 - b. With probability weights [aw=earnwt]
- e. Compute the number of workers in the U.S. [tabstat command]
- f. Create a variable representing the log(hrwage)
- g. Compute the mean of hrwage for workers under age 25
- h. Compute the mean of hrwage for workers between age 25 and 35

5. OLS REGRESSIONS.

To estimate an OLS regression of hourly earnings on years of schooling, we use the procedure called **regress**.

regress hrwage school

An intercept is automatically included (though it can be dropped as an option).
The regression could be restricted to people aged 55 or less with:

regress hrwage school if age<=55

or the regression could be done separately by sex with:

bysort female: regress hrwage _school

To generate a variable called *yhat* with predictions from the regression, and *uhat* with the predicted residuals:

predict yhat, xb

predict uhat, residual

Predict can also be used to generate residuals, standard errors of the prediction, etc. See help for regress.

Exercise 2.

- a. Estimate a simple linear regression of the hourly wage on years of education.
- b. Compute predicted values of hourly wage
- c. Compute predicted residuals
- d. Show that the sum of the residuals equals zero (use `tabstat` command)
- e. Show that the covariance between school and the residuals is zero (`corr` command)
- f. Show that the regression line “passes through the mean”
 - a. Note: you can do calculations in Stata by creating and displaying a scalar.
For example,

scalar x=13+14

display x

will create and display the value of (13+14)