# STATA Tutorial for EE469

January 29, 2015

## 1. Setting & Preparation

- First, start with ".do file" which is STATA syntax. Save all your commands in .do files.
- ***clear*** – remove all existing datasets from active use
- ***set more off*** – This will allow all of your output to scroll down the screen automatically.
- ***set mem 200m*** – Ask for more memory (usually not necessary in later version)
- Write a brief description of the task, date, and data set use. This serves as your own reminder for later work.

## 2. Import and save data

- Read files into stata and save files.
  1. Specify the directory where the data are stored
     Example:
     **cd J:\EE469\stata_workshop\data**
     ***use sample.dta, clear***
     [Note: The ", clear" option is to remove any active dataset that might be running.]

  2. Import the data from specified folder directly.
     Example:
     ***use J:\EE469\stata_workshop\data\sample.dta, clear***

- Import data in other formats (e.g. .xls, .txt, or .sav)
  - Use stat transfer software to convert data in other formats to stata format (.dta)
  - If the data are in excel format, save them as .csv and import the data using the drop-down menu in stata:
    File → import → data → text data created by a spreadsheet

- Data saving
  - Use command "save" by specifying new dataset name.
    Example: ***save sample_edit.dta***
  - If you want to use the same name, put suffix ", replace"
    Example: ***save sample_edit.dta, replace***

  - If you want to use the data with stata10 or older version, use saveold command:
    Example: ***saveold sample_edit.dta***

  **Note: Never overwrite the original dataset unless you have already made a copy!!**

## 3. Basic recode and data transformation commands

- First glance at the variables

  | | |
  |---|---|
  | ***sum*** | See all the variables and their summary |
  | ***codebook, compact*** | One line per variable, min, max, mean, # of unique values |
  | ***tab var1*** | See frequencies for discrete variables |

  Alternatively, you can glance at the data by using ***edit*** command. This will open data browser.
  Example: ***edit var1 var2 var3***

- Check missing data or string variables – the missing data in stata will appear as ".". You can check by using option "***m***" after tab command:
  Example: ***tab var1, m***

- The ***recode*** command:
  - Over-write original variable:
    Example: ***recode var1 (1=1) (2 3=2) (4 5 =3)***
  - Create new variable:
    Example: ***recode sex (1=1) (2 =0), gen(male)***

  - Dealing with ranges
    Example:
    ***recode occ3digit (100/199=1) (200/299=2) (300/399=3), gen(occ)***

- The ***replace*** command:
  - This is an alternative command to recode.
    Example:
    ***replace var1=2 if var1>1 & var1<4***
    ***replace var1=3 if var1>=4***
    This should give the same result as: ***recode var1 (1=1) (2 3=2) (4 5 =3).***

  - ***replace*** command is usually used together with "gen" command, or when dealing with using information from one (or more) variables to supplement another variable.
    Example:
    **replace occ=99 if unemploy==1**

- The ***rename*** command:
  Example:
  ***rename approx monthly_wage***

## 4. Generating new variables:

- Use "gen" command with mathematical operators.
  Example:
  ***gen age2 = age*age***
  ***gen lninc =ln(income)***
  ***gen srtvar1=sqrt(var1)***
  ***gen totinc = inc1+inc2***

- Generate new variables based on the information from other variable. This can be an alternative to ***recode*** command.
  ***gen male = 1***
  ***replace male=0 if sex==2***
  [This should give the same result as ***recode sex (1=1) (2 =0), gen(male).***]

- **Create dummy variables**
  - Use "***gen***" command.
    Example:
    ***gen male=(sex==1)***
    [Note: all other values will be recoded to "0". You need to check for missing.]
  - Use "***tab***" command.
    Example: Suppose we have age group variable (5 groups) and want to create to five dummies.

  ***recode age (11/20=1) (21/30=2) (31/40=3) (41/50=4) (51/60=5) (nonm=.), gen(age_r)***
  ***tab age_r, gen(agegrp)***

  Note: If using dummies in regression, you can also use prefix "xi" to create dummy variables. More details in the regression section.

  **Exercise 1: Create dummy variables from 'empstat'.**

## 5. Using "If" statement

- The operators after "if" statements:

| | | |
|---|---|---|
| == | means | "equal to" |
| != | means | "not equal to" |
| > | means | "greater than" |
| < | means | "less than" |
| >= | means | "greater than or equal to" |
| <= | means | "less than or equal to" |
| & | means | "and" |
| \| | means | "or" |

- Use in order to specify a subset:

Example: ***tab empstat if male==1***

- Use with “keep” and “drop” commands:
  Example:
  ***keep if male ==1***
  ***drop if age<20***

## 6. Variable and value labels

- To give a label to a new variable
  Example:
  ***recode work_7day (2=0), gen(wk7day)***
  ***label var wk7day "Worked in the last 7 days"***

- To give labels to variable values:
  Example:
  ***recode status (. = 99), gen(wkstat)***
  ***label var wkstat "Work Status"***
  ***label define wkstat 1 "Employer" 2 "Own-account worker" 3 "Unpaid family worker" 4 "Employee-Government" 5 "State Enterprise employee" 6 "Employee-private" 7 "Member of Co-operative group" 99 "NIU-age<15 or not working"***
  ***label values wkstat wkstat***

**Exercise 2: Give labels to variable and values according to the following commands:**

*label var dr_seek “During of seeking employment”*

*label define dr_seek 1 "Less than 1 mo" 2 "1-2.9 mo" 3 "3-5.9 mo" 4 "6-8.9 mo" 5 "9-11.9 mo" 6 "12 mo+" 9 "Unknown"*

*label val dr_seek dr_seek*

*label var empstat "Employment status for persons age 15+"*

*label define empstat 1 "Employed" 2 "Unemployed" 3 "Seasonally unemployed" 4 "Not in labor force"*

*label val empstat empstat*

## 7. Basic statistics

- Discrete variables
  Example:
  ***tab wkstat***
  ***tab wkstat, nol*** [option for no label]

***tab age_r wkstat***

- Continuous variables
  Example:
  ***sum age***
  ***sum age, d*** [d= detail]
  ***mean age***

- Get statistics for subgroups:
  Example:
  ***by sex: sum age***

- T-test
  Example:
  ***ttest age, by(sex)***

**8. Basic regressions**

- Basic OLS

  Example:

  ***reg monthly_wage male agegrp1-agegrp5***

- Use prefix "xi":
  Example:
  ***xi: reg monthly_wage male i.age_r i.wkstat***