Introduction to Stata

Lecture IV

Tomas R. Martinez

UC3M

September, 2019

Quote of the Day

"99 percent of all statistics only tell 49 percent of the story." -Ron DeLegge II

Data Manipulation

- We know how to import the data
- Now we will learn how to manipulate our data
- Probably the biggest strength of Stata
- **Example:** ECINF data (survey of small and informal firms)

Logical expressions

- Logical expressions
  - & and
  - | or
  - ! not
  - >= greater or equal
  - <= smaller or equal
  - == equal
  - ! = not equal
- Mathematical functions
  - abs(x) absolute value
  - log(x) log e
  - sqrt(x) square root
  - exp(x) exponential
  - help functions for more functions

Generating new variables

- **generate** [type] newvarname = expression [if]
- Where type can be byte, int, float, double, str, str2...
  - Numeric: byte, int, float...
  - String (non numbers): str, str1,...
- The difference between the numeric/string types are basically precision
- The more precise, more memory the variable "costs"
- In practice, Stata optimize a lot behind the scene but you can always use **compress** to store efficiently

Generating new variables

- **generate** [type] newvarname = expression [if]
- The [if] says which condition that the observation has to fulfill in order to get the assigned value for the new variable
- If that's not the case the variable has a missing assigned: .
- **CAREFUL:** Stata considers missing as a infinity positive value!
- Changing some values for a existing variable: **replace**

Drop and keep

- Getting rid of variables and observations
- **drop varname** $\rightarrow$ delete variable varname
- **drop if condition** $\rightarrow$ delete observation that satisfies the condition
- **keep** does the opposite
- **keep varname** $\rightarrow$ delete all the other variables not varname
- **keep if condition** $\rightarrow$ delete all observation that not satisfies the condition
- Do some examples

Grouping observations

- Very often micro data involves "groups" of some observation
- An individual is part of a household, in which is part of municipality, in which is part of a state...
- We want to calculate statistics, variables at the group level
- **egen** is an extension of generate and allows to create new variables using functions
- **Example:** egen mean_income = mean(income)
- This would create a variable equal to the mean of all observations

Grouping observations

- The **by** command
- Stata commands usually allow *by varlist: command*
- Basically, the command applies to different "varlist"' groups
- Your data need to be sorted by the "varlist": **sort**
- Just combine both
- **Example:** bysort sex: egen mean_income=mean(income)
- It generate a variable *mean_income* that is the average income of males for men and the average income of females for women

Collapse

- The **collapse** command converts the entire dataset to some group statistics
- **Example:** Let's say we have a categorical variable *education* with three education groups (less than high school, high school graduates and more than high school)
- **Example:** collapse (mean) income, by(year education)
- Gives a new data set with the average income by year and education group
- **That means you lose your previous data!!!**
- If you want to keep working in your data set write in your do-file: *preserve*, collapse and save, and *restore*