STATA II

Sara Nadel

sara_nadel@hksphd.harvard.edu

A note on log commands

- Always use *log close* and NOT *log off.*
  - Log close indicates to turn off the log entirely, whereas log off pauses it.
- About *capture:* try the following code

  cd "*directory name*"

  capture log close

  log using *filename.log,* replace

  [code of do-file]

  log close

Getting Started

- *(students who missed STATA I only)*: Copy and paste the MPAID_STATA file from the shared folder to your personal folder
- Open STATA on your computer
  - Start → All Programs → STATA
- Open a *do-file*
  - Window → Do File Editor → New Do-File
  - (*also: CTRL-8)*
- Check the auto-save on do/run box
  - On do-file page: edit→ preferences → Auto-Save on Do/Run
- Open data
  - Type the following into do file:

    cd "*path to the MPAID_STATA*"

    insheet using tz_data.csv, comma clear

  Do the do-file by clicking CTRL-D

The **set mem** command

- Necessary for large databases
- Use the **set mem XXXM** command to tell STATA to allow a larger database *for this time only*
  - Example: *set mem 50M*
- The **set mem** command can only be used when no data is loaded. Type *clear* before using the **set mem** command.

Opening Data

- **use**: for data that is in a STATA format.
- **insheet**: for data that is in a non-STATA format.(e.g. CSV – comma separated values)
  - Type *he insheet* into your command window
  - Key characteristics of insheet:
    - The word *using* is always included in the command
    - I recommend including the type of file (in this case, *.csv*). Otherwise, STATA assumes *.raw*, which is unusual.
    - I also recommend including the word *comma*

STATA Cardinal Rule # 2

NEVER save over original data

Save data under a different name

Saving data

- **save**: saves the data in STATA format
  - Type *save stata_2.dta*, *replace* into your do-file and do it by pressing CTRL-D
  - **replace**: this tells STATA that if there is already a file with the same name, that file should be saved over with the new file.
- **outsheet**: saves data in a .csv file format
  - Type *outsheet using stata2.csv, comma replace* into your do-file and do it by pressing CTRL-D
  - Note that *using*, *.csv*, and *comma* are used here
  - Outsheet is useful if you want to make a graph that you are more comfortable making in excel

The **gen** command

- **gen:** generates new variables
- See *help gen* for a description of the options
- Examples:
  - *gen totalfees = schoolfees + uniform + schoolsupplies*
  - *gen schoolage = 0*

    replace schoolage = 1 if age > 6 & age < 15
  - *gen number = _n*
    - *Assigns a number indicating the order of the variables (useful for sorting back to the original order in the future)*

Managing data

- **drop**: drops unwanted observations or variables
  - Examples:
    - *drop if everschool == .*
    - *drop village*
- **rename**: renames variables
  - *rename totalfees totalschoolcost*

The **tab** command

- **tab** creates tables
  - See *help tab (choose one- or two-way)* to see the many options
  - *tab inschool if schoolage == 1*
  - *tab inschool gender if schoolage == 1*
  - *tab inschool gender if schoolage == 1, cell*
  - *tab inschool gender if schoolage == 1, row*
  - *tab inschool gender if schoolage == 1, col*
  - *tab inschool gender if schoolage == 1, missing*
  - *tab gender if schoolage == 1, sum(inschool)*

The **tabstat** command

- **tabstat** can give more complex statistics
  - tabstat is very flexible – see *help tabstat*
  - Use of **by** in tabstat: groups data by a given variable
  - *tabstat inschool everschool agestarted if schoolage == 1*
  - *tabstat inschool everschool agestarted if schoolage== 1, stat(mean count)*
  - *by gender: tabstat inschool if schoolage == 1*
  - *by gender: tabstat inschool if schoolage == 1, stat(mean median max min)*

Labeling Data

- **label data:** labels the dataset overall
  - *label data "educational data from WB TZ dataset"*
    describe
- **label var**: labels a variable
  - *label var schoolage "Between the ages of six and fifteen"*
- **label values and label define**: attach descriptions to categorical operators
  - *label define yesno 1 "yes" 0 "no"*
    label values inschool yesno
    tab inschool if age < 18 & age > 7

Graphs

- Drop-down menu is useful for making graphs
- Histogram: relative frequencies of the ages of people in school

  histogram age if inschool == 1, title(Age of Individuals in School)

  graph save age_inschool.gph, replace
- Bar Graph: means of different groups

  graph bar (median) agestarted, over(gender) title(Age of beginning school by gender)

  graph save gender_age_school.gph, replace

Exercises: Construct a do-file that does the following

- Creates a variable, *adult*, that indicates that the individual is 18 yrs or older. Label both the variable and the categories (1=yes, 0 = no)
- Identify what percentage of the adults can read, by gender
- Find the average, median, and maximum age at which educated adults began school, by gender
- Look at how many people were interviewed in each region. What is notable about this?

Summary

- Opening data: **use** and **insheet**
- Saving data: **save** and **outsheet**; **replace**
- Cardinal Rule # 2: Never save over original data
- Generating variables using **gen**
- Managing data: **drop, rename**
- Tables: **tab** and **tabstat**
- Labeling: **label data, label var, value define** & **label values**
- Graphs**: histogram** and **bar graph**