Introduction to Stata

Ista Zahn

Harvard-MIT Data Center (HMDC)

dataclass@help.hmdc.harvard.edu

Documents for Today

USERNAME: dataclass

PASSWORD: dataclass

- Find class materials at:

  Scratch > StataIntro

- FIRST THING: copy this folder to your desktop!

Organization

- Please feel free to ask questions at any point if they are relevant to the current topic (or if you are lost!)
- There will be a Q&A after class for more specific, personalized questions
- Collaboration with your neighbors is encouraged
- If you are using a laptop, you will need to adjust paths accordingly

Organization

- Make comments in your Do-file rather than on hand-outs
  - Save on flash drive or email to yourself
- Stata commands will always appear in red
- “Var” simply refers to “variable” (e.g., var1, var2, var3)
- Pathnames should be replace with the path specific to your computer and folders

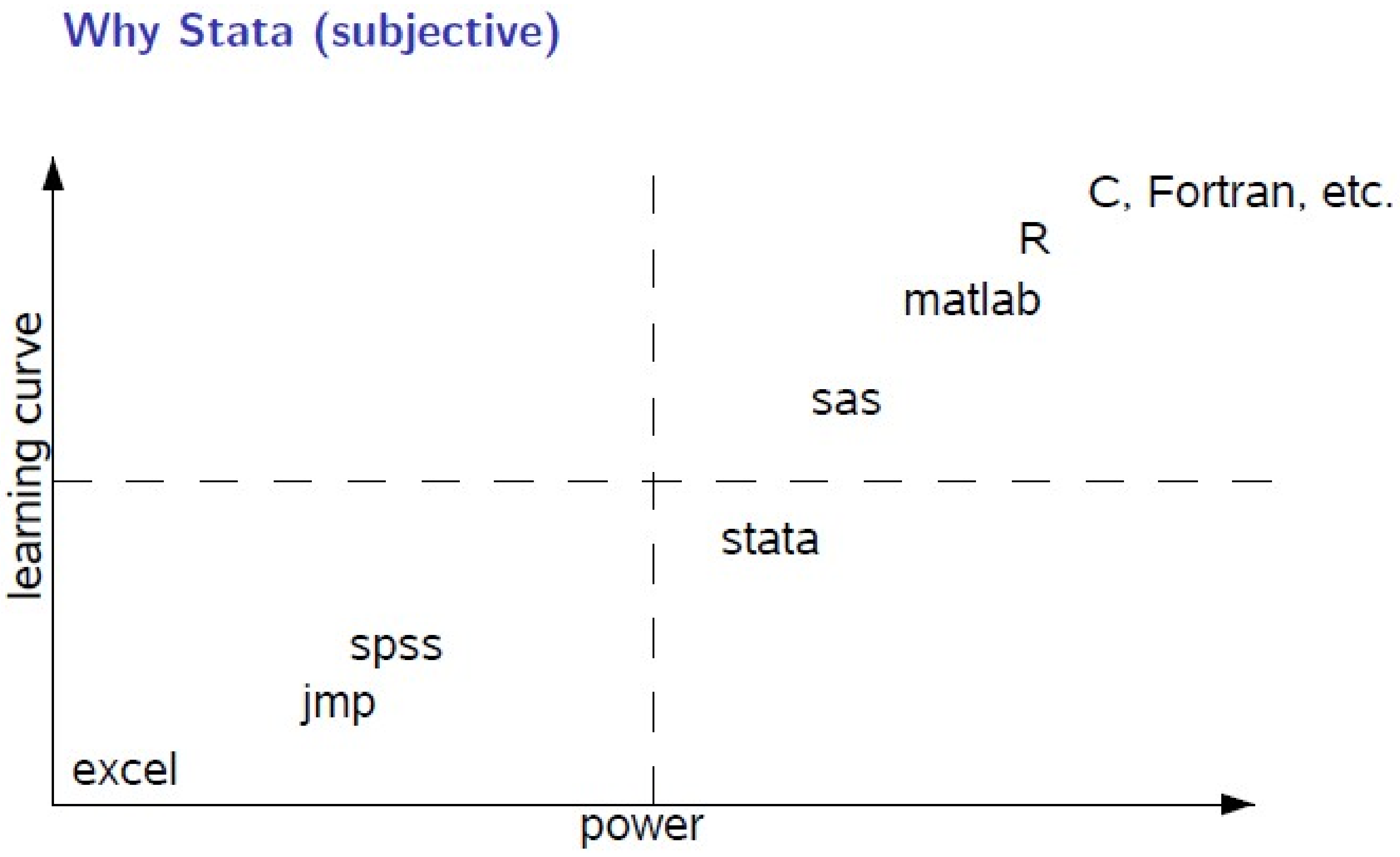
Assumptions and Disclaimers

- This is an **INTRODUCTION** to Stata
- Assumes no/very little knowledge of Stata
- Not appropriate for people already well familiar with Stata
- If you are catching on before the rest of the class, experiment with command features described in help files

Why Stata?

- Used in a variety of disciplines
- User-friendly
- Great guides available on web (as well as in HMDC computer lab library)
- Student and other discount packages available at reasonable cost

Why Stata?



Why Stata (subjective)

Stata Interface

- Comprised of four windows:
  - Results
  - Command
  - Review
  - Variables
- Review and Variable windows can be closed (user preference)
- Command window can be shortened (recommended)

Do-files

- A fifth window, called a “Do-file” is also useful
- Open Do-file via icon or with dropown menu
- You can type all the same commands into the Do-file that you would type into the command window
- BUT…the Do-file allows you to SAVE your commands
- Your Do-file should contain ALL* commands you executed
  - *at least all the “correct” commands!

Command Window vs. Do-file

- I recommend never using the command window or menus to make CHANGES to data
- Saving commands in Do-file allows you to keep a written record of everything you have done to your data
  - Allows easy replication
  - Allows you to go back and re-run commands, analyses and make modifications

Let’s get started

- Copy the IntroStata folder to the desktop!
- Open up a new Do-file
- Set your directory

```
cd “C://Users/dataclass/Desktop/StataIntro"
```

- Start a log file to record your stata session
  - To create a log file:
  - log using logname [, append replace]
  - Pause / resume logging with log off and log on

Data File Commands

- Next, we want to open your data file
- Retrieving your data file:
  - use datasetname.dta
- Saving your data file:
  - save datasetname.dta
  - This command should be followed by ", replace" if you're writing over an existing file

How to Start Every Do-file

1. Describe what the file does
2. Change directory
3. Begin log file
4. Call up data
5. Save data under new name (if making changes to dataset)

```
/*DESCRIPTION OF FILE*/
cd " ~/StataIntro"
log using logname
use datasetname.dta
save newdata.dta
```

A Note About Path Names

- If your path has no spaces in the name (that means all directories, folders, file names, etc. can have no spaces), you can write the path as is
- If there are spaces, you need to put your pathname in quotes
- Best to get in the habit of quoting paths

Where’s my data?

- Data editor (browse)
- Data editor (edit)
  - Using the data editor is discouraged (why?)
- Always keep any changes to your data in your Do-file
- Avoid temptation of making manual changes by viewing data via the browser rather than editor

Stata Help

- Easiest way to get help in Stata – just type “help” followed by topic or command
  - help regress
- Falls back to “search” if command not found
- Generally, if you google “Stata [topic],” you’ll get some helpful hits
- UCLA website: http://www.ats.ucla.edu/stat/Stata/

General Stata Command Syntax

- Most Stata commands follow the same underlying principles
- Command variable(s), options
  - sum var1 var2, detail
  - CAUTION – in some cases, if you type a command and don't specify a variable, Stata will perform the command on all variables in your dataset
- You can find command-specific syntax in the help files

Commenting and Formatting Syntax

- Start with comment describing your Do-file
- Use comments throughout
  - Stata needs to be told what is a comment and what is a command:
    - *comment
    - describe var
    - /*comment block comment block comment block comment block comment block comment block */
- Use /// to break varlists over multiple lines:

```
describe var1 var2 var2 ///

var4  var5 var6
```

What if my data is not a Stata file?

- Delimited, ASCII (text file)
  - insheet using gss.csv, clear
  - outsheet using gss_new.csv, replace comma
- Stata will open SAS transport files
  - fdause gss.xpt

What if my data is from another statistical software program?

- SPSS/PASW will allow you to save your data as a Stata file
  - Go to: file > save as > Stata (use most recent version available)
  - Then you can just go into Stata and open it
- StatTransfer

What if my data is in excel?

- You can copy and paste your excel file directly into Stata’s data editor
- You need to make sure that all of your columns have labels
- After you paste, you will see a prompt asking, “Is the first row data or variable names?”
  - Select “treat first row as variable names”
- Or, if you save as .xml use syntax:

```
xmluse use gss.xml, doctype(excel) firstrow
```

Exercise 1: Importing Data

1. Close down Stata and open a new session
2. Go through the three steps for starting each Stata session that we reviewed
   - Begin a log file
   - Open your Stata dataset (gss.dta)
   - Save your Stata dataset using a different name
3. Try opening the following files:
   - A comma separated value file: gss.csv
   - A SPSS file: gss.sav
   - A SAS transport file: gss.xpt

Descriptive Statistics

- Review your data carefully
  - describe
  - sum
  - codebook
  - list
  - tab
- Remember, if you run these commands without specifying variables, Stata will produce output for every variable

Basic Graphing Commands

- View data visually with a histogram
  - hist varname
  - Interested in normality of your data? You can tell Stata to draw the normal curve over your histogram
  - hist varname, normal
- View bivariate distributions with scatterplots
  - twoway (scatter var1 var2)
  - graph matrix var1 var2 var3

Variable and Value Labels

- You never know why and when your data may be reviewed
- ALWAYS label every variable no matter how insignificant it may seem
- Stata uses two sets of label commands
  - 1. variable labels
  - 2. value labels

Variable Names and Labels

- Label variable inc “household income”

```
la var inc “household income”
```

- Want to change the name of your variable?

```
rename oldvarname newvarname
```

Value Labels

- Value labels are labels you put on the values that variables take on (e.g., “yes,” “no,” “1,” “2,” “3”)
- Value labels are a two step process:
  - 1. “define” a value label
  - 2. Assign defined label to variable(s)

Variable and Value Labels

- Let’s define a value label for yes/no responses

```
la define example 1 “Yes” 0 “No”
```

- Stata knows what our label means, but now we need to assign it to variable(s)

```
la val var1 var2 var3 example
```

- Label define particularly useful when you have multiple variables with the same value structure
- If you have many variables, you can search labels using: lookfor

```
lookfor income
```

Exercise 2: Variable Labels and Value Labels

1. Open the data set, gss.csv
2. Take a look at your data using one of the data review commands we discussed.
3. Rename your variables and add variable names using the following codebook:
   - v1, marital, marital status
   - v2, age, age of respondent
   - v3, educ, education
   - v4, sex, respondent’s sex
   - v5, inc, household income
   - v6, happy, general happiness
   - v7, region, region of interview
4. Add value labels to your “marital” variable using the following codebook:
   - 1 “married”
   - 2 “widowed”
   - 3 “divorced”
   - 4 “separated”
   - 5 “never married”

The Cardinal Rule of Data Manipuation

- After ensuring variables were correctly imported you may wish to create new variables or modify existing variables
- NEVER save over an original data file (why?)

Useful Data Manipulation Commands

- == equal to (status quo)
- = used in assigning values
- != not equal to
- > greater than
- >= greater than or equal to
- & and
- | or

Data Manipulation Commands

- Create new variables using “gen”

  ```
  gen newvar1 = var1^2
  gen newvar2_1 = var2 – var1
  ```

- Sometimes useful to start with blank values and fill them in based on values of existing variables
  - Start by generating a column of missings
    - `gen newvar = .`
  - Next, start adding your qualifications
  - `replace newvar=1 if var1==2`
  - `replace newvar=2 if var1==2 & var2==2`
  - `replace newvar=3 if var1==2 | var2==2`

Data Manipulation Commands

- Recoding variables

```
recode varname (1=2) (2=3)
```

- Deleting variables

```
drop varname
```

- Keeping a subset of variables

```
keep var1-varn
```

The “By” Command

- Sometimes, you’d like to generate output based on different categories of a single variable
  - For example, say you want to look at happiness based on whether an individual is male or female
- The “by” command does just this

```
bysort sex: tab happy

hist happy, by(sex)
```

Exercise 3: Manipulating Variables

- Use the dataset, gss.dta
  1. Generate a new variable, age2
  2. Generate a new “high income” variable that will take on a value of “1” if a person has an income value greater than “15” and “0” otherwise
  3. Generate a new divorced/separated dummy variable that will take on a value of “1” if a person is either divorced or separated and “0” otherwise

Exercise 4: Descriptive statistics

1. Use the dataset, gss.dta
2. Examine a few selected variables using the describe, sum and codebook commands
3. Tabulate the variable, “marital,” with and without labels
4. Cross-tabulate marital with region and show gender percent by region
5. Summarize the variable, “income” separately participants based on marital status
6. Summarize the variable, “happy” for married individuals only
7. Generate a histogram of income
8. Generate a second histogram of income, but this time, split income based on participants’ sex and ask Stata to print the normal curve on your histograms

The RCE

- Research Computing Enviroment (RCE) service available to Harvard & MIT users
  - www.iq.harvard.edu/research_computing
- Wonderful resource for organizing data, running analyses efficiently
- Creates a centralized place to store data and run analysis
- Supplies persistent desktop environment accessible from **any** computer with an internet connection

Other Services Available

- Institute for Quantitative Social Science
  - www.iq.harvard.edu
- Computer labs
  - www.iq.harvard.edu/facilities
- Training
  - www.iq.harvard.edu/training

Thank you!

Institute for Quantitative Social Science (IQSS) offers statistical workshops in Stata, SAS and R throughout the semester.

The R Series

- Introduction to R
- R and Statistics
- R Programming

Stata and SAS Courses

- Introduction to Stata
- Data Management in Stata
- Regression in Stata
- Graphics in Stata
- Introduction to SAS

For more information, visit:
http://support.hmdc.harvard.edu/kb-20/statistical_support
Sign up anytime by emailing:
dataclass@help.hmdc.harvard.edu

Help Us Improve This Course!

- Please take a minute to tell us how we did
- **http://tinyurl.com/6h3cxnz**