

Content Outline for Stats II Extra Stata sessions

Max Callaghan

November 2015

1 Reproducible Research and Literate Programming

This first session should give students the tools necessary to produce research output that links data, code and narrative.

1.1 About Reproducible Research and Literate Programming

- Explanation of the principles of Reproducible Research and Literate Programming
 - “The standard of reproducibility calls for the data and the computer code used to analyze the data be made available to others” [1]
 - Literate programming ties together data, code and the actual research output, enhancing reproducibility
- How these can help students avoid errors and unnecessary repetitive tasks

1.2 Tools for Reproducible Research and Literate Programming

- Git and Github
- Rmarkdown

1.2.1 Options for Creating Documents that include Stata Output

- Why is copying, taking screenshots, and pasting into word suboptimal?
- Option 1: Use word’s -link- or -includetext- fields to include log output that can be updated automatically
 - Text fields can include output from log files that automatically updates when you run your do file. However, you would have to run a script to clean these log files [Need to check if this works on school computers. Also, it may be easier to make this a stata function & this needs to be extended to control

formatting. Perhaps it's possible to write a stata function that allows for including text and document writing instructions within a do file].

- Word is easy to use but can be frustrating when what you want to do is very specific. Formatting can be an issue.
- Option 2: Use latex to write your document. Write your output tables into .tex files and graphs into image files and link to them into your document.
 - This way, if you make a change to your do file which changes your output, recompiling your pdf will automatically reflect those changes.
 - [Is it possible to install latex on school computers?]
- Option 3: Use Statweave to create one document containing instructions for writing and formatting your document as well as instructions for stata to run. [As above, is installation on school computers an issue?]

2 Getting Started with Stata

This, and the following sections, should give students a clearer understanding of how stata works, and should equip them with some further strategies and tools for solving problems with stata.

2.1 General Pointers

- Where to look for help
- How to read stata manual pages

2.2 Directory Structure

- File paths and the working directory
- How thinking about directory structure can make things easier

2.3 Reading Data

- Reading data types not formatted for stata: xlsx, csv, etc. Overcoming issues with delimiters and unhelpfully formatted data files
- Reading data from web sources

2.4 Processing Data

- Data types
- Generating new variables

3 Automating Stata

3.1 Macros

- Using macros to store text
 - Storing and evaluating a macro
 - Using a macro to re-use a list of variables without retyping
 - Including a macro within a macro
- Using macros to store results
 - Storing estimation results

3.2 Loops

- Looping through sequences of numbers
- Looping through lists of variables

4 Presenting Results with Stata

4.1 Postestimation

- Accessing and using postestimation results in `e()`
- Using `-predict-` to generate new variables

4.2 The `-estout-` program

- Storing regression output and presenting using `-esttab-`
- Computing additional statistics to add to results tables
- Saving tables to various file formats for use in output

4.3 Producing Graphs

- Different types of graphs using stata
- Combining graphs
- Customising the appearance of graphs in stata
- Saving and presenting graphs in your output

Some Further Resources

- <http://data.princeton.edu/stata/>
- <https://github.com/HertieDataScience/SyllabusAndLectures>

References

- [1] Roger D. Peng. Reproducible research in computational science. *Science*, 334(6060):1226–1227, 2011.