Empirical trade analysis

Introduction to Stata

Cosimo Beverelli

World Trade Organization

Outline

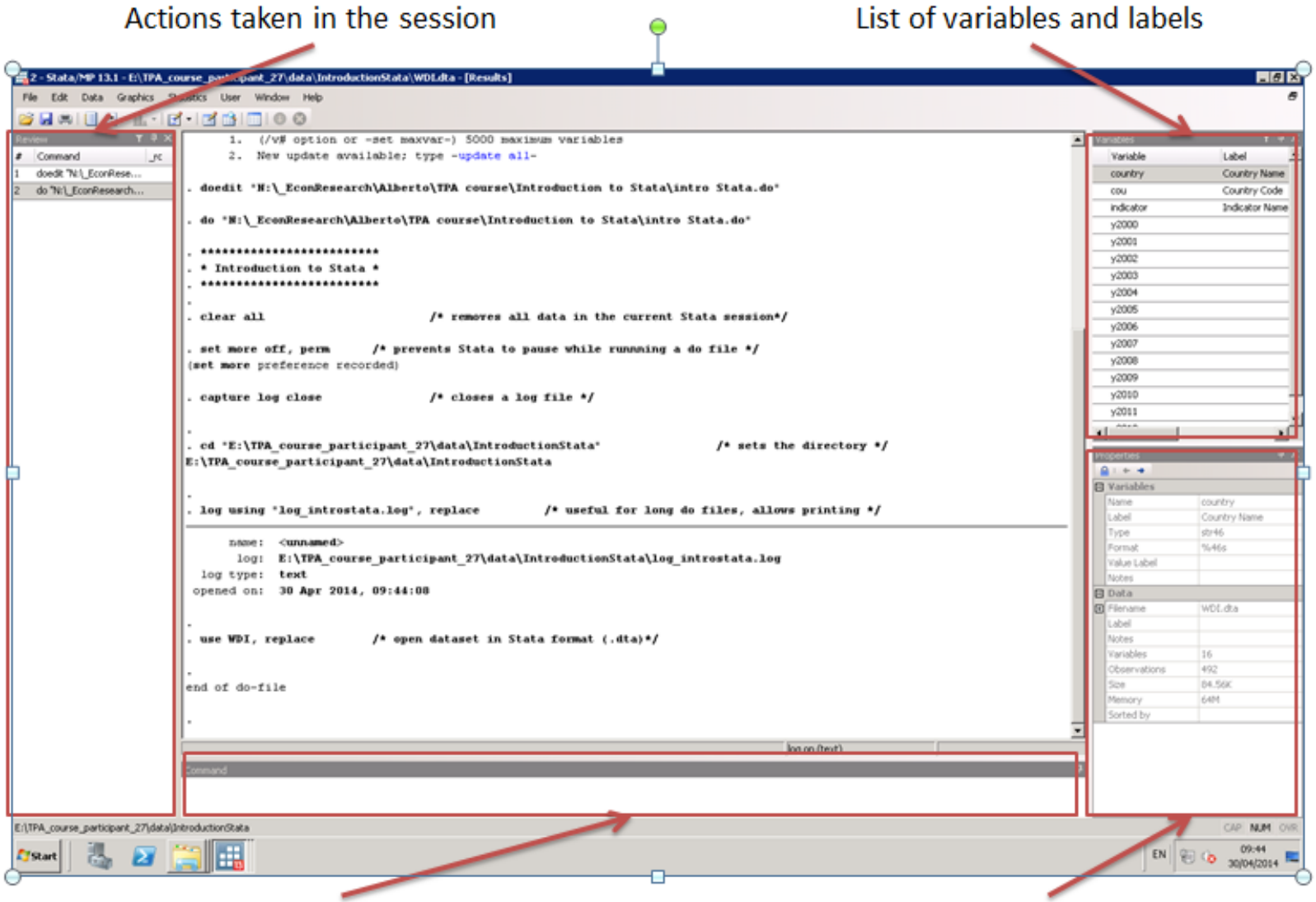
Resources

1. Stata help and Stata manual
2. A variety of books covering Stata exist

- Web resources:

1. Germán Rodríguez's webpage
   - Data management, graphics and programming
2. UCLA IDRES' webpage
   - Very comprehensive, covering all sorts of topics (data management, analysis...) with several examples
   - Includes classes and seminars, learning modules and FAQs
3. Statalist
   - Typically accessed via a Google search

Stata windows



Folders

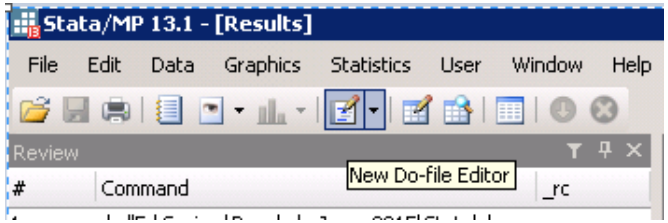
- The "Bangkok_December_2017" folder contains the course material
- You will see a the following three subfolders relevant for Stata analysis:
  - "data"
  - "do_files"
  - "results"
- Creating these subfolders is a way of organizing your work
- The first thing to do is to set a working directory (see slide 6)
- *Hint:* Note that Stata is case-sensitive

Defining the working directory

- Both if you work alone and (especially) if you work in teams, it is important to define a working directory
- This can be do with a "do file" which I normally call "directory_definition.do"
- What is a do file? $\longrightarrow$ See next slide

Stata do files

- A do file is a set of Stata commands typed in a plain text file
- When you work with STATA, always use do files, e.g. one do file for creating your master dataset and one do file for regressions
- Do files can also be used to set globals and directories or to run a series of different do files in progression
- *Hint:*
  - To open a do file, either click on the appropriate icon...



  - ...or use Ctrl+9

Typical commands at beginning of each do file

- `clear all` /* removes all data in the current Stata session */
- `set more off, perm` /* prevents Stata to pause while running a do file */
- `log using "filename", replace` /*opens (replacing it) a log file */
- `capture log close` /* closes a log file (no error if no log file is open) */
- *Notes:*
  - "*" treats everything after it in a line as a comment
  - "/* text */" will make Stata treat "text" as a comment ("text" can span over several lines)
- $\longrightarrow$ Stata demonstration

Datasets and do files used in this introduction to Stata

- To apply some of the Stata commands described in this presentation, we will use two datasets:
  1. WDI_extraction.csv – a very small subset of the World Development indicators
  2. WBES_extraction.xls – a very small subset of the the World Bank Enterprise Surveys
- You can find these datasets in the "data" directory: "data→Stata_introduction"
- There are also 4 do files:
  - 01_data_intro_stata.do
  - 02_descriptive.do
  - 03_reshape_merge.do
  - 04_loops.do
- You can find these do files in the directory: "do_files→Stata_introduction"

Importing and saving data into Stata

- Importing data (main commands):
  - `insheet` for .csv files
  - `import delimited` for .txt files
  - `import excel` for Excel files (can import sheets of a multi-sheet file)
- Saving data
  - `save` *`filename`*`, replace`
- **Exercise**: execute the do file 01_data_intro_stata.do

Basic commands

- Describe the variables
  - `describe` *`varlist`*
- Installing packages
  - `ssc install (or findit)` *`packagename`*
- Identify missing values, which appear as "." (dot) or empty cell
  - `inspect` *`varlist`* `(codebook` *`varlist`*`)`
- Identify duplicate observations
  - `inspect duplicates (report/drop/tag/list)` *`varlist`*
- Identify number of unique values
  - `unique` *`varlist`* `(also, codebook for single variable)`
- Browse the dataset
  - `browse` *`varlist`*

Basic commands (ct'd)

- `generate`
- `destring/tostring`
- `replace`
- `rename/renvars`
- `keep`
- `drop`

- The list can go on...
- ...what is important is to keep in mind that, in case of doubt, you can always use the `help` command

List of useful operators commonly used in expressions

Arithmetic	**Logical**	**Relational**
+ add	! not (also ~)	== equal
- subtract	| or	!= not equal (also ~=)
* multiply	& and	< less than
/ divide		<= less than or equal
^raise to power		> greater than
+ string concatenation		>= greater than or equal

Commands for descriptive statistics

- `summarize` *`varlist`*
- `tabulate` *`var1 var2`*
- `table` *`rowvar (colvar)`*`, content ()`
- `tabstat` *`varlist`*`, statistics() by()`

The egen command

- Used to create new variables
- Commonly used egen functions (refer to WBES_extraction.dta dataset):
  - `bysort cou sector: egen total_sales_sec=total(sales), missing`
  - `bysort cou sector: egen avg_sales_sec=mean(sales)`
  - `egen exp_tot=rowtotal(exp_intermediate exp_final)`
  - `egen id_cluster=group(cou sector)`
  - `gen cou_sec=concat(cou sector)`
  - Further functions include: `max`, `min`, `count`, `tag`...

String functions

- `generate` *`newvar`*`=function (`*`varname`*`)`, where *`varname`* is a string variable
- Some useful functions:
  - `abbrev` $\longrightarrow$ shortens the string the number of indicated characters
  - `length` $\longrightarrow$ returns the number of characters of the string
  - `subinstr` $\longrightarrow$ allows to replace or delete particular substrings
  - `substr` $\longrightarrow$ allows to extract substrings based on its position
  - `upper (lower)` $\longrightarrow$ changes the entire string to uppercase (lowercase) strings
  - `trim` $\longrightarrow$ removes leading and trailing blanks of the string

The `collapse` command

- `collapse (mean)` *`varlist`* `(sum)` *`varlist`*`, by(`*`varlist`*`)`
  - Creates an aggregate dataset by e.g. averaging or summing variables across the dimension identified in `by()`
  - All observations not included in the command are dropped
  - Useful in analysis when moving to a higher level of aggregation, e.g. aggregating trade flows from HS 6-digit to HS 2-digit
  - Useful for calculating descriptive statistics before exporting them to Excel using `export excel`

The `collapse` command (ct'd)

- If you do not want to collapse, `duplicates drop` after `bysort (): egen` gives the same results as collapse
- See lines 144-154 of the do file 02_descriptive.do

- **Exercise**: execute the do file 02_descriptive.do

The `reshape` command

- Reshapes dataset from long to wide format and vice versa
- See `help reshape` for graphical visualization
- In a panel (say country-sector-year), the country, sector and year dimensions are normally in long format

- **Exercise**: Open WDI_extraction.dta and reshape it first long and then wide (see do file 03_reshape_merge.do)

Merging datasets

- To merge datasets you can use `joinby` (which I normally use) or `merge`
- The `joinby` command: `joinby` *`varlist`* `using filename, unmatched(`*`options`*`)`
  - It forms all pairwise combination for *varlist*
  - Unmatched can keep unmatched observations from the master dataset (`master`), the using dataset (`using`) or both (`both`)
  - In these cases, a `_merge` variable has to be created

- **Exercise**: Open BES_extraction.dta and merge it with WDI data (see 03_reshape_merge.do)

Macros

- Macros are names associated with some text
  - The commands `global` and `local` assign strings to global and local macro names
- Globals and locals have a variety of uses:
  - To define the directories
  - They are used in loops (see next slides)
  - A set of explanatory variables can be grouped under one macro name
- Global macros, once defined, are available anywhere in Stata
- Local macros are only available within the selected lines of a do file

Loops

- See Stata help and Germán Rodríguez's webpage
- Two main commands: `foreach` and `forvalues`
- `foreach` loops through strings of text, `foreach` loops through numbers
- `foreach` can also be used to loop over variables and numbers
- Examples: See lines 8-47 of 04_loops.do

Graphics

- Useful links: official Stata and UCLA IDRES
- Main commands include `histogram`, `graph bar`, `twoway`, and `graph twoway`
- Examples: See lines 54-130 of 04_loops.do

- **Exercise**: execute the do file 04_loops.do