D0N90A: Empirical Evaluation of Economic Policy: Introduction to Stata

Iris Kesternich

University of Leuven

Spring semester 2021

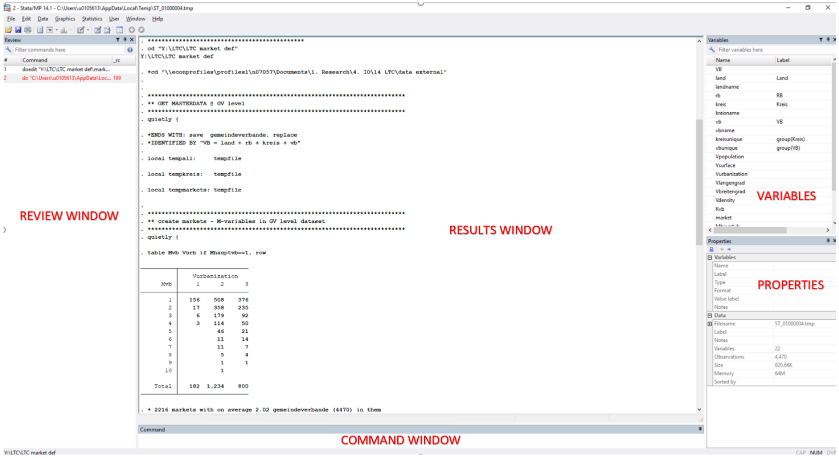
Access

There are different possibilities to access Stata

- On the computers in KU Leuven PC-pools
- On your own computer via a remote desktop connection using VPN to KU Leuven computers `https://feb.kuleuven.be/public/u0017833/courses/timeseries/Connecting%20to%20the%20student%20Remote%20Desktop%20Server%20from%20a%20public%20network.pdf`
- Buy a licence (very costly)

In case you do not know yet which access to use: send today's materials (dofiles) to your email to save them!

User-interface



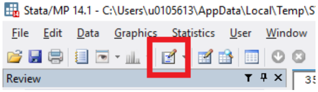
#	Command	_rc
1	doedit "Y:\LTC\LTC market def\mark...	
2	do "C:\Users\u0105613\AppData\Loc...	199

```
. *******************************************
. cd "Y:\LTC\LTC market def"
Y:\LTC\LTC market def

. *cd "\\econprofiles\profiles1\n07057\Documents\1. Research\4. IO\14 LTC\data external"
.
.
. ***********************************************************************
. ** GET MASTERDATA @ GV level
. ***********************************************************************
. quietly {

. *ENDS WITH: save  gemeindeverbande, replace
. *IDENTIFIED BY "VB = land + rb + kreis + vb"
.
. local tempall:     tempfile

. local tempkreis:   tempfile

. local tempmarkets: tempfile


.
. ***********************************************************************
. ** create markets - M-variables in GV level dataset
. ***********************************************************************
. quietly {

. table Mvb Vurb if Mhauptvb==1, row

------------------------------
          |    Vurbanization
      Mvb |     1      2      3
----------+-------------------
        1 |   156    508    376
        2 |    17    358    235
        3 |     6    179     92
        4 |     3    114     50
        5 |           46     21
        6 |           11     14
        7 |           11      7
        8 |            5      4
        9 |            1      1
       10 |            1
          |
    Total |   182  1,234    800
------------------------------

. * 2216 markets with on average 2.02 gemeindeverbande (4470) in them
```

Name	Label
VB	
land	Land
landname	
rb	RB
kreis	Kreis
kreisname	
vb	VB
vbname	
kreisunique	group(Kreis)
vbunique	group(VB)
Vpopulation	
Vsurface	
Vurbanization	
Vlangengrad	
Vbreitengrad	
Vdensity	
Kvb	
market	

Variables	
Name	
Label	
Type	
Format	
Value label	
Notes	
Data	
Filename	ST_01000004.tmp
Label	
Notes	
Variables	22
Observations	4,470
Size	820.66K
Memory	64M
Sorted by	

Do-files


Stata/MP 14.1 - C:\Users\u0105613\AppData\Local\Temp\S
File Edit Data Graphics Statistics User Window
Review

A do-file is a script of commands. The commands tell Stata which tasks it has to perform such as loading data, running statistical analyses,... When the do-file is executed, it runs through all the subsequent commands that are written down.

Always work from a do-file when using Stata:

- ⇒ Results can be replicated at a later date
- ⇒ Mistakes are not permanent. You can simply fix the mistake in the do-file and run it from the beginning.

Folder management

Very import to create a link between Stata and the place where you store your files:

- Whenever you command Stata to use a file, e.g. a data file, Stata needs to know where it has to look for this file.
- Similarly, whenever you want to save output created in Stata, e.g. a graph or a table, Stata needs to know where it has to save this output.

Best practice:

- Keep all files necessary for one project in the same folder
- Refer to that folder in your do-file by copying in its address (between quotation marks!!) after the `cd` command, e.g.:
  `cd "X:\CES\empirical_eval\Stata"`

First commands

Always start your do-file with the following commands:

```
clear
set more off
cd "X:\CES\emprical_eval\Stata"
```

- `clear`: clears everything from memory, it allows you to start with a clean slate whenever the do-file is executed
- `set more off`: no breaks when running the do-file
- `cd + folder address` creates link between Stata and the working directory (fill in your own folder address!!)

First commands

Very important: use comments to make your dofile more readable - for yourself in the future and for your teacher!

- Use * to comment an entire command line from beginning onwards, e.g `*Exercise 1`
- Use // to comment the remainder of a command line, e.g
  ```
  clear // clears everything from memory
  ```
- Use /* ... */ to comment an entire block of text, e.g
  ```
  clear
  /*clears everything from memory, it allows you to
  start with a clean slate whenever the do-file is
  executed*/
  ```

Finding help

If you have problem or need to find more information on what a certain command does:

- Google: type in your question in google, you can most likely find an answer on an internet forum or find a link to the official documentation of the Stata manual.
- From within Stata: type `help command` into the command window to read the command information. For example:
  `help regress`
  Downside: you need to know the exact name of the command

Importing data

Stata can import (and export) different types of datasets:

- Stata format: .dta extension
  ```
  use dataset_name.dta
  ```
- Excel format: .xls or .xlsx extension
  ```
  import excel using dataset_name.xlsx, firstrow
  ```
  - Make sure that your excel file has the correct structure: rows denote observations and columns denote variables
- Delimited text file: .csv or .txt extension
  ```
  import delimited using dataset_name.txt,
  delimiters(",")
  ```
  - Delimited files are files where each element of data is separated by a “delimiter”, this can be a tab, comma, semicolon,...

Inspecting data

Once you have loaded in the dataset, you can start exploring it:

- `browse`: Look at your data in the browse window
- `describe`: Produces a list of all the variables, their data-type and label
- `sum variable_names`: Gives the number of observations, mean, min and max of the variables specified after `sum`
- `sum variable_name, detail`: Gives more detailed summary statistics of the specified variable
- `tab variable_name`: Produces a frequency table: it gives the number of occurrences for each value in the variable

Inspecting data

Types of variables

- Two important distinctions: Numerical and text (string) variables
- Different sub-categories for numerical and string variables
- Example: country or name = string variable, age of respondent = numerical
- You will need to modify commands depending on the type of variable you are working with

Making changes to existing data

You can edit the data in several ways. Operations can affect more than one variable and all observations or a subset of observations of a variable.

- `drop variable_name(s)`: Deletes the variable(s) from the dataset
- `keep variable_name(s)`: Deletes the variable(s) that are not specified after `keep`
- `rename variable_name new_variable_name`: Renames the variable
- `label var variable_name "new_label"`: Attaches a label to the variable

Generating new variables

- Generate a new variable by giving it a name and by defining the values of that variable in an expression.
  `gen new_variable_name =` *expression*
- The expression can be a number of things. For example:
  - A mathematical expression:
    `gen variable1 = 200` or `gen variable2 = ln(2)`
  - A piece of text (called "string" in Stata):
    `gen variable3 = "abc"`
  - A function of existing variables
    `gen variable4 = variable1 + variable2`

Replacing values of a variable (I)

- The syntax of a replace command is very similar to that of the generate command:
  `replace variable_name =` *expression*
- It replaces the values of an existing variable with the values defined by the expression after the equality sign.
- `replace` is usually accompanied by an `if` statement. The replace command is then applied on observations that satisfy the `if` condition. For example:
  `replace age_category = "old" if age>70`

Replacing values of a variable (II)

The logical expression that can be used in an `if` statement:

- "==": equal to
- "!=": not equal to
- "<=": smaller than or equal to
- ">=": greater than or equal to
- "<": smaller than
- ">": greater than

Including multiple conditions in one line: separate conditions with "&" if they hold at the same time, with "|" if only one of the conditions must hold.

Data visualization

For an overview of possible graphical representations of your data, type `help twoway` into Stata.

- A scatter plot shows how two variables correlate:
  ```
  twoway scatter variable_1 variable_2
  ```
- A histogram and kernel density plot show the distribution of a variable:
  ```
  histogram variable_name
  kdensity variable_name
  ```
- A bar chart displays variable. This can be done over different groups of observations:
  ```
  graph bar (mean) variable_name,
  over(group_variable)
  ```