Introduction to Stata – Session 1

Tarjei Havnes

[1] ESOP and Department of Economics
University of Oslo

[2] Research department
Statistics Norway

ECON 4136, UiO, 2012

Preparation

Before we start:

1. Create the folder statacourse in your home directory (e.g. in your Documents folder)
2. Download all .dta-files from the course homepage
    - http://www.uio.no/studier/emner/sv/oekonomi/ECON4136/h12/
3. Save the file to the folder statacourse
4. Go to kiosk.uio.no (Internet Explorer!) and log on using your UIO user name
5. Navigate to Analyse (english: Analysis)
6. Open StataIC 11

Aims

You should know

- The STATA interface (command line, results window, variables, review)
- Reading data into STATA
- Using help and some basic commands
- We will review some of these as we go along

Aims

You should know

- The STATA interface (command line, results window, variables, review)
- Reading data into STATA
- Using help and some basic commands
- We will review some of these as we go along

You should learn

- Using do-files and logging your session
- Combining data sets (merge, append)
- Using panel data (reshape, xt-commands)
- Running regressions (regress, logit, probit)
- Using and reporting estimation results (estimates, esttab, test)
- Using macros and loops (local, global, forvalues, foreach, while)

Challenges

- Wide difference in what you know and what you like
- You need to spend some time on this to get comfortable
- Please try not to clam up: ask a classmate, then me.

Let me know if things are too fast (or too slow).

Outline

1. What do we want? Why Stata?
2. The basics of Stata
3. Data in memory and subscripting
4. Stata workflow
5. Preparing your data

Tasks we want to perform

1. Data management
    - create a new data set
    - merge different data sets

Tasks we want to perform

1. Data management
   - create a new data set
   - merge different data sets

2. Data manipulation
   - create new variables from existing
   - sort observations
   - change order of variables

Tasks we want to perform

1. Data management
   - create a new data set
   - merge different data sets

2. Data manipulation
   - create new variables from existing
   - sort observations
   - change order of variables

3. Data analysis
   - graphs, tables, ...
   - summarize separately: mean, count, variation, ...
   - summarize jointly: correlations, regressions, inference, ...

Why not use a spreadsheet (Excel etc.)?

Excel allows you to do

- hands-on data management and manipulation
- many types of analysis (even regression)

Why not use a spreadsheet (Excel etc.)?

Excel allows you to do

- hands-on data management and manipulation
- many types of analysis (even regression)

But it is

- terribly cumbersome in practice
  - especially when no. of variables or observations is large
- very difficult to check formulas = very easy to make mistakes
- impossible to backtrack data manipulation

Why not use a spreadsheet (Excel etc.)?

Excel allows you to do

- hands-on data management and manipulation
- many types of analysis (even regression)

But it is

- terribly cumbersome in practice
  - especially when no. of variables or observations is large
- very difficult to check formulas = very easy to make mistakes
- impossible to backtrack data manipulation

Excel/spreadsheet programs

- are forbidden for analysis and data manipulation
- may be useful for presenting data, inputting data and (rarely) graphing/tabulating

Why not use a spreadsheet (Excel etc.)?

A major advantage is that Stata lets you

- log everything you do
- save the actual steps you have performed separately to run again later
    - potentially after changing (correcting) some steps

Why STATA, exactly

STATA is probably the most common in economics and the social sciences

- Efficient in run time
- Efficient in programming time
- Lots (and lots) of help, tutorials and discussions out there
- Lots of ready-made programs for what you may want to do

Why STATA, exactly

STATA is probably the most common in economics and the social sciences

- Efficient in run time
- Efficient in programming time
- Lots (and lots) of help, tutorials and discussions out there
- Lots of ready-made programs for what you may want to do

But there are many alternatives, e.g.

- R: free and popular
- MatLab: popular in dynamic macro, very efficient at matrix operations
- SPSS: popular in political science, perhaps simpler UI
- ...

Stata syntax

With a few exceptions, the basic language syntax in Stata is

```
command [varlist] [if] [, options]
```

where [...] indicate optional elements

Suppose you want to estimate an OLS regression of the variable *lnincome* on the variable *educ* for men only, this would look something like this:

```
.  regress lnincome educ if female==0
```

Note: Stata is case-sensitive (advice: only use upper-case for strings)

Core commands (know these!)

Task	Commands
getting help	help, findit, lookfor
moving around FS	cd, dir (ls)
memory	clear, set memory
using Stata data	use, save, append, merge
reading raw data	insheet, infix, infile
looking at data	describe, list, tabulate, summarize
preparing data	generate, replace, rename, egen, encode sort, by, reshape, collapse, keep, drop
formatting	format, label
saving output	log
swiss pocket knife	display

Wildcards

There is no need to type the complete variable name: the shortest string of characters that uniquely identifies the variable (given the data currently loaded in memory) suffices

Example: suppose you have data in the following order (country2.dta)
country y1980 y1985 y2000 y1990 y1995

- Lists of variables can be selected using wildcards
  * = zero or more chars here
  ? = one char here
  - y* selects y1980 y1985 y1990 y1995 y2000
  - y198? selects y1980 y1985
  - y*0 selects y1980 1990 y2000
- Ranges of variables can be selected using '-'
  - y1980-y1990 selects y1980 y1985 y2000 y1990

Getting help

Getting help on a command in Stata is easy, typing

```
.  help command
```

will open a window that explains the full syntax of -command- and often includes examples. Use -help- if you want to find out more about the commands.

To search for a command you can use

```
.  findit keyword(s)
```

which will search the keynote database and the Internet and pop-up a window with the search results.

- -hsearch- searches the help files only.

Reading Stata dataset

```
. use auto
(1978 Automobile Data)

. describe

Contains data from auto.dta
  obs:            74                          1978 Automobile Data
 vars:            12                          13 Apr 2009 17:45
 size:         3,774 (99.9% of memory free)   (_dta has notes)
----------------------------------------------------------------------------
              storage  display     value
variable name   type   format      label      variable label
----------------------------------------------------------------------------
make            str18  %-18s                  Make and Model
price           int    %8.0gc                 Price
mpg             int    %8.0g                  Mileage (mpg)
rep78           int    %8.0g                  Repair Record 1978
headroom        float  %6.1f                  Headroom (in.)
trunk           int    %8.0g                  Trunk space (cu. ft.)
weight          int    %8.0gc                 Weight (lbs.)
length          int    %8.0g                  Length (in.)
turn            int    %8.0g                  Turn Circle (ft.)
displacement    int    %8.0g                  Displacement (cu. in.)
gear_ratio      float  %6.2f                  Gear Ratio
foreign         byte   %8.0g       origin     Car type
----------------------------------------------------------------------------
Sorted by:  foreign
```

Stata keeps one (1) table in memory at a time

columns (variables) are named

```
. list make price mpg

     +-----------------------------------+
     | make                  price   mpg |
     |-----------------------------------|
  1. | AMC Concord           4,099    22 |
  2. | AMC Pacer             4,749    17 |
  3. | AMC Spirit            3,799    22 |
  4. | Buick Century         4,816    20 |
  5. | Buick Electra         7,827    15 |
     |-----------------------------------|
  6. | Buick LeSabre         5,788    18 |
  7. | Buick Opel            4,453    26 |
  8. | Buick Regal           5,189    20 |
```

Stata keeps one (1) table in memory at a time

rows (observations) are numbered

```
. list make price mpg in 3/5

     +-----------------------------+
     | make              price   mpg |
     |-----------------------------|
  3. | AMC Spirit        3,799    22 |
  4. | Buick Century     4,816    20 |
  5. | Buick Electra     7,827    15 |
     +-----------------------------+

. display mpg[3]
22

. display "km/l " 0.425*mpg[3]
km/l 9.35
```

Stop it! (or not)

```
. list make price mpg

[output omitted]

 30. | Merc. Cougar          5,379    14 |
     |---------------------------------|
 31. | Merc. Marquis         6,165    15 |
 32. | Merc. Monarch         4,516    18 |
--more--
```

- typing <Enter> : shows next line
- typing <Space> : shows next screen of output
- typing <q> : breaks

You can `-set more off-` (or `-set more on-`)

- to break output that scrolls by use <Ctrl+Break> (<Ctrl+C> on Unix)

Example session

```
. list make price mpg rep78 in 1/5

     +-------------------------------------+
     | make               price   mpg   rep78 |
     |-------------------------------------|
  1. | AMC Concord     4,099    22       3 |
  2. | AMC Pacer       4,749    17       3 |
  3. | AMC Spirit      3,799    22       . |
  4. | Buick Century   4,816    20       3 |
  5. | Buick Electra   7,827    15       4 |
     +-------------------------------------+

. sum make price mpg rep78

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
        make |         0
       price |        74    6165.257    2949.496       3291      15906
         mpg |        74     21.2973    5.785503         12         41
       rep78 |        69    3.405797    .9899323          1          5
```

Browsing and editing data

You can also look at the data with the data editor (browse)

- launch using: -browse [varlist] [if]-
- try: browse make price if rep==.

You can edit data in a spreadsheet calling the command `edit`

- ONLY do this if you are constructing a new data set, or
- if you know EXACTLY what you're doing
- ALWAYS log your session if you edit something
  - or you lose the ability to backtrack

Missing values

How Stata defines missing values:

- Numeric missing values are represented by large positive values
  - shown as a dot '.'
- Empty strings are treated as missing values of type string

Watch out:

- `income > 100` evaluates to TRUE (=1) for income larger than 100 AND missing values!!!
- `income >= .` evaluates to TRUE for missing values

Most Stata statistical commands deal with missing values by disregarding observations with one or more missing values (called "listwise deletion" or "complete cases only")

Stata workflow

Personal hygiene

In practice you should always try to strictly separate changing & analysing data:

1. first prepare your data for analysis
   - copy data from disk to memory
   - change data (prepare for analysis)
   - save data to disk under new name
2. then analyze these data
   - copy analysis data into memory
   - start logging results to file
   - perform analysis
   - close log file

Advice: one directory per project & start session in project dir

Working in the menu line

You can use Stata through the menus (instead of command line)

- Don't use them
- With two potential exceptions:
  - Graphs: Save time
  - Learning syntax/Exploring what Stata can do
    - ★ This is usually easier in help files, manuals or online

Data types and memory

Keep track of data types

- numeric (byte, int, long, float, double)
- string
- large difference in memory
- try `compress`

In Stata 11 and earlier, you often need to assign memory

- Allocate memory with `-set mem-`
  - e.g `-set mem 250m-` or `-set mem 1g-`
- Assign as much memory as you need, no more
  - analyze data = data set + 30-40%
  - prepare data = data set + 60-80%

Do files

Until now we have used the command line:

- great to develop but not to reproduce your analysis
- ALWAYS organize your work in Stata scripts

Stata scripts are called do-files after their extension (.do)

Use do-files (with informative names) to organize your work:

- create dataset
  crincome.do makes data file income.dta
- analysis
  andescr.do calculates my descriptive statistics
  anreg.do performs my regression analysis
- making graphs
  grwageplot.do makes the graph wageplot.eps

Note: do-files can call do-files.

- You can create a master do-file which calls the do-files which reproduce your complete preparation and analysis trail

Do files

Make a do-file that does the following (USE HELP!)

- navigate to your working directory
- read in data file cps1992to2008.dta
- summarize your data
- summarize log hourly wages with and without a bachelor
- regress log hourly wages on bachelor
- table the estimated coefficients, SEs, R2 and sample size
  - with three decimals
  - indicating 1%, 5% and 10% significance with stars
- include a control for age and age squared, and table both results in the same table
- use robust standard errors, and include results in the same table

Do-file

```
//ancps1992to2008.do - ECON4136 session 1
cd D:\Dropbox\mine-ting\undervisning\statacourse
use cps1992to2008.dta, clear
d
sum
sum ahe if bachelor == 0
sum ahe if bachelor == 1
gen lnahe = ln(ahe)
regress lnahe bachelor
est store lnwbach
est tab , b(%8.3f) se(%8.3f) stats(N r2)

// EXTRA
gen agesq = age*age
regress lnahe bachelor age agesq
est store lnwbach_age
regress lnahe bachelor, robust
est store lnwbach_robust
regress lnahe bachelor age agesq, robust
est store lnwbach_age_robust
est tab lnwbach lnwbach_age lnwbach_robust lnwbach_age_robust ///
, b(%8.3f) se(%8.3f) stats(N r2)
```

Do-file

```
//ancps1992to2008.do - ECON4136 session 1
cd D:\Dropbox\mine-ting\undervisning\statacourse
use cps1992to2008.dta, clear
//DATA MANIPULATION
gen lnahe = ln(ahe)
gen agesq = age*age
//DESCRIPTIVES
sum
sum ahe if bachelor == 0
sum ahe if bachelor == 1
//REGRESSION ANALYSIS
regress lnahe bachelor
est store lnwbach
regress lnahe bachelor age agesq
est store lnwbach_age
regress lnahe bachelor, robust
est store lnwbach_robust
regress lnahe bachelor age agesq, robust
est store lnwbach_age_robust
//REPORT
est tab lnwbach , b(%8.3f) se(%8.3f) stats(N r2)
est tab lnwbach lnwbach_age lnwbach_robust lnwbach_age_robust ///
, b(%8.3f) se(%8.3f) stats(N r2)
```