Introduction to Stata – Session 1

Tarjei Havnes

[1]ESOP and Department of Economics
University of Oslo

[2]Research department
Statistics Norway

ECON 3150/4150, UiO, 2014

Preparation

Before we start:

1. Sit in teams of two
2. Download the file auto.dta from the course homepage
   - http://www.uio.no/studier/emner/sv/oekonomi/ECON4150/v14/
3. Save the file in a new folder "statacourse" in your home directory (e.g. in your Documents folder)
4. Go to kiosk.uio.no (Internet Explorer!) and log on using your UIO user name
5. Navigate to Analyse (english: Analysis)
6. Open an available Stata version (preferably Stata 12, if not available then Stata 11).

Aims and challenges

We have 3×2 hours

- You will not learn STATA in these lessons
- This will require effort on your own time

My aim

- Dampen your fears: It really ain't that hard
- Show you what STATA is and how we work with it (roughly)
- Show you some important commands, and how to use them

Challenge:

- Wide difference in what you know and what you like
- You will NOT become an apt programmer unless you enjoy it (somewhat)
- Please try not to clam up: ask your partner, then me.

Outline of the course

Session 1: The basics of Stata, Reading data, Stata workflow
Session 2: Working with data, Do-files
Session 3: Merging and reshaping data sets, Drawing graphs

Outline of this session

What do we want? Why Stata?

Quick start: Your first interactive session

The basics of Stata

Reading data

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)

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

How does STATA differ?

Just like Excel, start by reading in data in a spreadsheet (matrix)

- columns: variables
- rows: observations

Just like Excel, define a formula for a new variable

- excel: =B1/C1
  - copy down to generate =B2/C2 etc.
- stata: gen y = B/C
  - generates new variable y equal to fraction of variables B and C

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

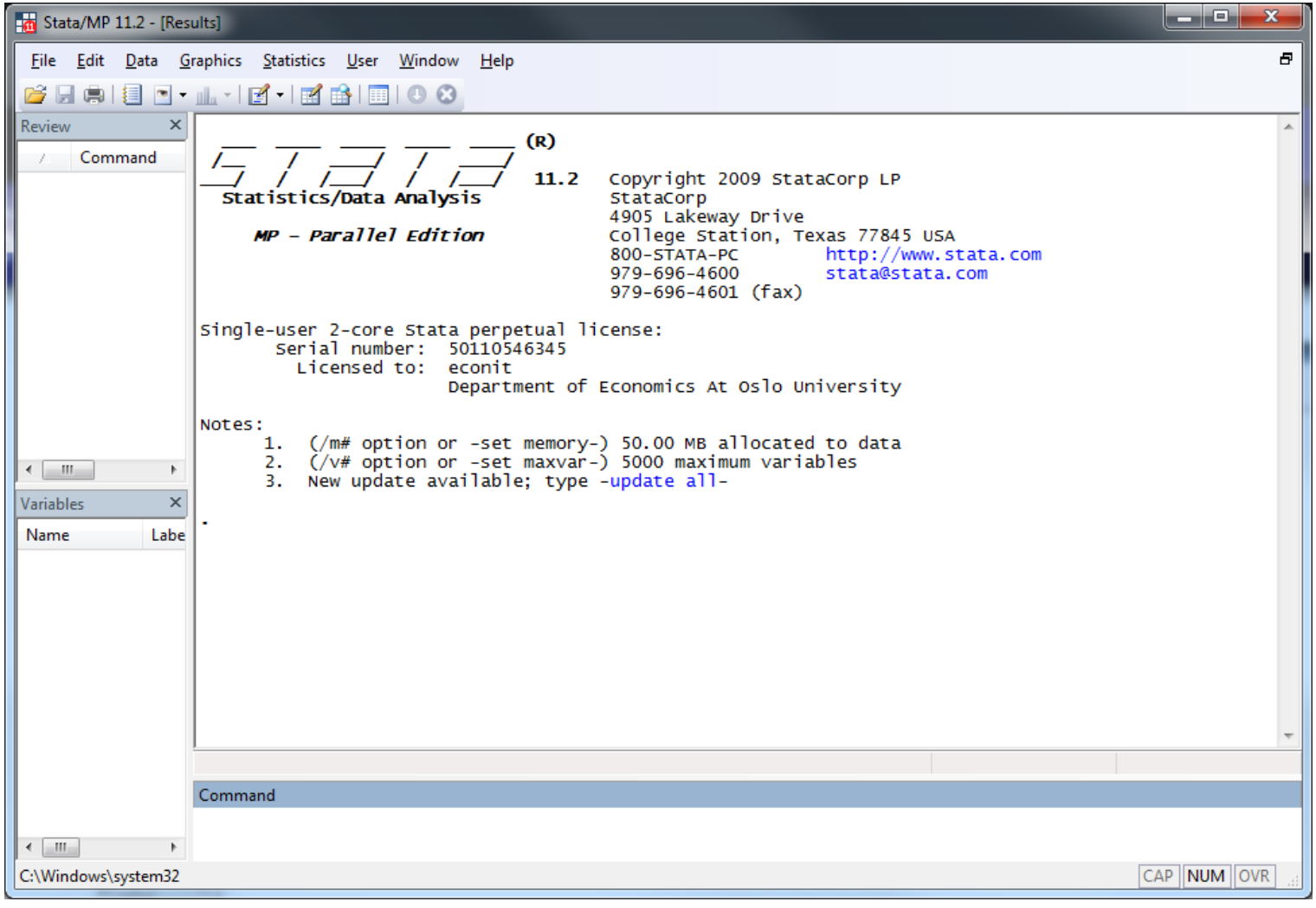
STATA is probably the most common in economics and the social sciences

- 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

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 User interface



Quick start

To see that this doesn't have to be hard, let's just dive straight in

First, do the following:

1. Go to File/Change Working Directory ...
2. Navigate to your statacourse folder. OK
3. Go to File/Open. Open the file auto.dta
4. Go to Data/Describe/Describe data in memory. OK
5. Go to Statistics/Summaries/ Summary.../Summary Statistics. OK

Next, figure it out:

1. Make a histogram of price
2. Make a table of price and weight by whether the car is foreign or not:

```
foreign   | price weight
----------+--------------------
Domestic | 6072.423 3302.549
Foreign   | 6384.682 2315.909
----------+--------------------
Total     |  6165.257 3005.205

------------------------------
```

Quick start

- Notice that your commands pop up in the Results-window
- You can actually generate a do-file (we'll talk about this much later) of what you just did
  - Right-click the Review-window
  - Select Send to Do-file editor
- You likely made many mistakes above, which are also included in this file
  - The actual commands we performed above are simply

```
cd "PATH\statacourse"
use auto, clear
describe
histogram price
summarize
tabstat price weight,
by(foreign)

(OR: table foreign,

contents(mean price mean

weight))
```

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 price weight if foreign==0
```

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

Stata syntax – Regression

We are interested in the relationship btw price and weight.

- OLS fits a line through this observation cloud

```
. twoway scatter price weight
```

- Specifically, we estimate α and β in the equation

$$price_i = \alpha + \beta\, weight_i + \varepsilon_i$$

```
. regress price weight
. predict phat
. sort weight
. twoway (scatter price weight) (line phat weight)
```

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.

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

Stata uses memory

Stata works like a text editor (or spreadsheet):

Editor	**Stata**
copy text from disk to memory	copy data from disk to memory
change text	change/analyse data
save to disk	[save to disk]

Reading Stata dataset

```
. cd "PATH\statacourse"

. 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
```

Managing memory

In Stata, the whole dataset needs to fit into memory!

Allocate memory with `-set mem-`

- f.e. `-set mem 250m-` or `-set mem 1g-`

Remove data from memory with `-clear-`, remove everything with `-clear all-`

Do not use more memory

- than physically available (virtual memory is slow)
- than needed (you are not alone on the server)

How much memory do you need?

- analyze data + 30-40%
- prepare data + 60-80%

Note: No longer necessary in Stata 12 or later!

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")

Working in the menu line

As we saw, you can also use Stata through the menus (instead of command line)

- You should try not to use them.
  - If you want to work more on Stata than for this course: JUST DON'T!
- With two potential exceptions:
  - Graphs: Save time
  - Learning syntax/Exploring what Stata can do
    - Over time, this is easier in help files, manuals or online

What you should have learned...

- How to move around in Stata and use the command line
- Stata's way of representing data
  - data area, missing values, wildcards
- Stata's command syntax
  - subsetting your data when executing a command
- Commands:
  - help, use, list, summarize, display
  - tabstat,

Homework for next week

1. Go through the tutorial "Introduction to Stata 1" on the course homepage
   - Alternatively, go through the first part of "Introduction to Stata 2"
2. Go through these slides and try all the commands
   - Use the help file to find some options you may be interested in