Introduction to Stata - Session 1

Andrea Papini

ECON 3150/4150, UiO

January 14, 2017

Preparation

Before we start

- Sit in teams of two
- Download the file auto.dta from the course homepage
- Save the file in a new folder "statacourse" in your home directory (e.g. in your Documents folder)
- Go to kiosk.uio.no (using Internet Explorer) and log on using your UiO user name
- Navigate to Analyse
- Open an available Stata version (the newest available)

Aim with stata sessions

Challenges:

- We will start learning a tool that you don't know what you need for yet.
- We have only three double hours, which means that learning STATA requires effort on your own time in addition.

Advantages:

- Stata is partly intuitively built so it is not as hard as it looks.
- Many things can be figured out through trial and error inside Stata.
- There is a ton of help on Google.

Aim:

- Equip you for using Stata for solving seminar exercises.
- Make you interested in learning more so that you can use Stata in later work.

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.
   - Label and structure variables.
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 spreadsheet

Excel may be useful for presenting data, inputting data and does allow you to do data management, manipulations and many types of analysis but:

- it is easy to make typographical errors and there are no protection against it. Difficult to check formulaes
- it is impossible to backtrack data manipulation. Provide no audit trail so others cannot easily control your work.
- possible truncation data values, (data that is truncated (norsk: avkortet) at top and/or bottom.
- it is cumbersome when dealing with a large number of observations.

.. and once you get used to the graphs from Stata you will think the graphs from excel look horrible.

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

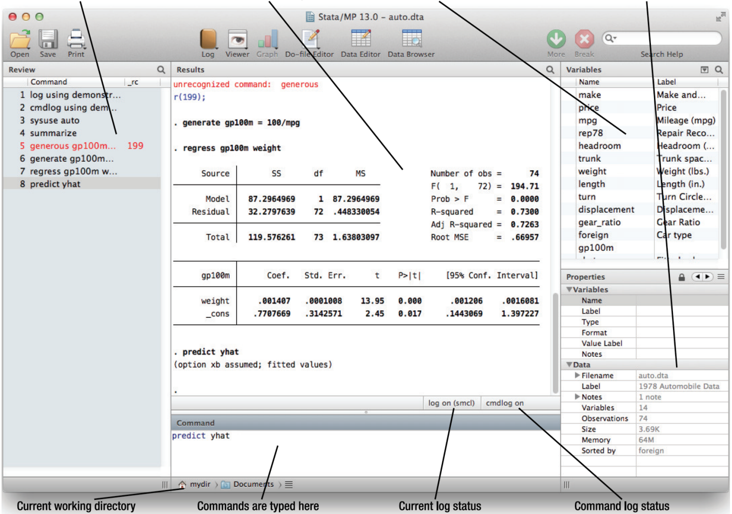
STATA is one of the most common in economics and the social sciences:

- Efficient in run time and in programming time.
- Lots of help, tutorials and discussions on the web.
- Stata offers lots of help, tutorials and discussions available on the web.
- Stata offers lots of ready-made programs for what you may want to do so you don't have to know programming.

Stata is one of the most common tools in economics and social sciences. Alternative statistical softwares:

- R: free and popular.
- MatLab: Popular in dynamic macro, very efficient at matrix operations.
- SPSS: popular in political sciences.
- Python etc etc

Stata 101 - user interface



Past commands appear here

Results are displayed here

Variable list appears here

Data properties appear here

Current working directory appears here

Commands are typed here

Current log status appears here

Command log status appears here

	Command	_rc
1	log using demonstr...	
2	cmdlog using dem...	
3	sysuse auto	
4	summarize	
5	generous gp100m...	199
6	generate gp100m...	
7	regress gp100m w...	
8	predict yhat	

```
unrecognized command:  generous
r(199);

. generate gp100m = 100/mpg

. regress gp100m weight

      Source |       SS       df       MS              Number of obs =      74
-------------+------------------------------           F(  1,    72) =  194.71
       Model |  87.2964969     1  87.2964969           Prob > F      =  0.0000
    Residual |  32.2797639    72  .448330054           R-squared     =  0.7300
-------------+------------------------------           Adj R-squared =  0.7263
       Total |  119.576261    73  1.63803097           Root MSE      =  .66957

------------------------------------------------------------------------------
      gp100m |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
      weight |    .001407   .0001008    13.95   0.000      .001206    .0016081
       _cons |   .7707669   .3142571     2.45   0.017     .1443069    1.397227
------------------------------------------------------------------------------

. predict yhat
(option xb assumed; fitted values)

.
```

Command

```
predict yhat
```

Name	Label
make	Make and...
price	Price
mpg	Mileage (mpg)
rep78	Repair Reco...
headroom	Headroom (...
trunk	Trunk spac...
weight	Weight (lbs.)
length	Length (in.)
turn	Turn Circle...
displacement	Displaceme...
gear_ratio	Gear Ratio
foreign	Car type
gp100m	

Properties	
▼Variables	
Name	
Label	
Type	
Format	
Value Label	
Notes	
▼Data	
▶Filename	auto.dta
Label	1978 Automobile Data
▶Notes	1 note
Variables	14
Observations	74
Size	3.69K
Memory	64M
Sorted by	foreign

Figure: Source:STATA manuals13

Working in Stata

You can either:

- Find the desired alternative in the menu
- Write the command associated with the desired alternative in the command window.
- Example: Change working directory:
  - Go to File/Change Working directory and navigate to your statacourse folder
  - Write cd "File-path"

Quick start

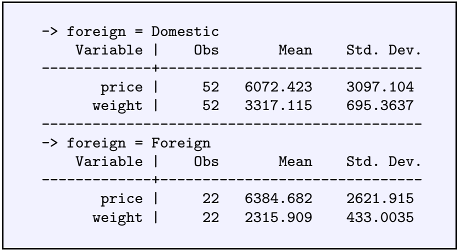
To see that this doesn't have to be hard, let's start using STATA!

First, do the following:

1. Go to File/Change Working Directory
2. Navigate to your stata course 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, try to:

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



```
-> foreign = Domestic
    Variable |     Obs        Mean    Std. Dev.
-------------+---------------------------------
       price |      52    6072.423    3097.104
      weight |      52    3317.115    695.3637
-----------------------------------------------
-> foreign = Foreign
    Variable |     Obs        Mean    Std. Dev.
-------------+---------------------------------
       price |      22    6384.682    2621.915
      weight |      22    2315.909    433.0035
```

Quick start

- Notice that your commands pop up in the Results-window
- You can actually generate a do-file (we'll talk about this later) of what you just did, in order to save the commands.

- You likely made many mistakes above,
  - The actual commands we performed above are simply

```
cd "PATH/stata"
use auto, clear
describe
histogram price
summarize
by foreign, sort : summarize  price weight
summarize  price weight if foreign == 0
summarize  price weight if foreign == 1
```

Loading data

We can either:

- Load existing Stata data (a .dta file). → Go to File/Open
- Load data from other sources
  - Load data from excel (a .xsl file) → Go to File/Import Excel Spread sheet. Find file and mark of "Import first row as variable names." Or alternatively copy and paste to data editor.
  - Load a comma-separated file (a .csv file) → go to File/Import/Text Data. Find file. In window choose delimiter Comma (or the correct delimiter) and "OK".
- Generate a new data set.
- Use online data sets without downloading:

  - use "link adress", clear -

Loading data (2)

The following data sets are ‘stored’ in Stata

```
. sysuse dir
  auto.dta        census.dta      network1.dta    sp500.dta
  auto2.dta       citytemp.dta    network1a.dta   surface.dta
  autornd.dta     citytemp4.dta   nlsw88.dta      tsline1.dta
  bplong.dta      educ99gdp.dta   nlswide1.dta    tsline2.dta
  bpwide.dta      gnp96.dta       pop2000.dta     uslifeexp.dta
  cancer.dta      lifeexp.dta     sandstone.dta   uslifeexp2.dta
```

These data sets can be loaded by the command -sysuse ‘filename’.

```
sysuse auto.dta, clear
```

Exercise

- Download and open the auto.csv data
- Insert the data to the left into Stata
- Download and open the auto.dta data

Browsing and editing data

Three alternatives to visually inspect the data: (browse)

- Go to Data, Data Editor, Data Editor (Browse).The information is stored in columns→ (variables) and rows↓ (observations).
- Press the browse data button
- `browse [varlist] [if]` in the command window

In a similar manner you can edit the data: (using 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 sessions of you edit something so you can backtrack.

List, Describe

`list`

- displays the values of variables.
- If no variables are specified the values of all the variables are displayed.

`describe`

- Go to Data/Describe/describe data in memory. Press "OK".
- Data types:
  - integer - only whole numbers.
  - float: a fractional (floating point) number.
  - String: A sequence of characters.
  - Byte: Is the smallest integer type.
- The variable labels tell you what each variable measures (and in what units).

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:

- Thus `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”)

Summary Statistics

To look at summary statistics (mean, obs, std. dev): go to Statistics/Summaries/Summary.../Summary Statistics. "OK". Use the auto.dta and find:

- What is the price of the five cars with a missing value for `rep78` (that is `rep78==.`)
- Get the summary statistic for the variable `price`
- Get the summary statistic for the variable `price` if `mpg` is less than 21.
- Get the summary statistic for the variable `rep78` (is it including missing values?)
- Get the summary statistic for the variable `price` if `rep78>10` (is it including observations with missing `rep78`?)

Working in Stata

So far we have worked Stata using the menus.

- This is useful for graphs, typically saves time.
- It is useful to learn commands and explore what Stata can do.

Over time you should rather use commands:

- Over time it is easier to explore new things through help files, manuals or online.

Summary Statistics: Commands

The previous example can be executed as:

```
 . list price if rep78==.
     |  price |
  3. |  3,799 |
  7. |  4,453 |
 45. |  6,486 |
 51. |  4,424 |
 64. | 12,990 |

. summarize price
    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
       price |        74    6165.257    2949.496       3291      15906

. summarize price if mpg<21
    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
       price |        38    6937.316    3262.392       3291      14500

.  summarize rep78
    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
       rep78 |        69    3.405797    .9899323          1          5

. summarize price if rep78 >10
    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
       price |         5      6430.4    3804.322       3799      12990
```

Stata syntax

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

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

where [..] indicate optional elements. Example:

- `summarize` or just `sum`
  - provides summary results for all variables
- `summarize price`
  - provides summary results for only the variable price.
- `summarize price if foreign==1`
  - provides summary results of price for only foreign cars.

Each alternative can be also be inserted through the summary statistic window.

Commands

To get help on a command in Stata simply write:

```
help command
```

which will open a window that explains the full syntax of the command and often includes examples.

If you don't know the name of your command but know what you want to do you can search for commands by:

```
findit keywords
```

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

Not all packages of commands are by default installed by stata. To install a new package write

```
ssc install "package name"
```

Core Commands

Task	Commands
getting help	`help`, `findit`, `lookfor`
using Stata data	`use`, `save`, `append`, `merge`
looking at data	`describe`, `list`, `tabulate`, `summarize`

preparing data	`generate`, `replace`, `rename`, `egen`, `encode`
	`by`, `reshape`, `sort`, `collapse`, `keep`, `drop`

saving output	`log`
“calculator”	`display`

Command efficiency

- There is no need to type the complete command or variable name. You may abbreviate commands and variables as long as Stata may not become confused about what you mean i.e. the shortest string of characters that uniquely identifies the variable suffices. (e.g. sum instead of summarize)
- List of variables can be selected using wildcards:
    - * = zero or more characters here
    - ? = one character here
    - - = range of variables.

Ex: If you have the variables year2000, year2005, year2010 then:

- year* selects all the variables
- year200? selects year 2000 and year 2005
- y*0 selects year2000 and year2010.

NOTE: Stata is case-sensitive

Efficiency (2)

Useful keyboard commands:

PgUp	Retrieves previous command
cursors	Back and forward to go back and forth inside your command
Home/End	To get to beginning/end of your command
ESC	Delete all written in command window
Ctrl + Del	Delete to the end of line

Exercise

use nlswide1.dta stored in STATA (`sysuse nlswide1.dta, clear`)

- Describe the data to understand what the data is about
- produce summary statistics for all variables
- produce summary statistics for all variables in 1968
- produce summary statistics for wage in 1968 and 1988
- produce summary statistics for variables `ttl_exp68 tenure68 hours68 wage68`

Exercise

```
sysuse nlswide1.dta, clear
describe
sum
sum *68
sum wage*
sum wage??
sum ttl_exp68 - wage68
```

Stata memory

Useful commands:

- `clear`
  - removes data and value labels from memory
- `clear results`
  - eliminates stored results from memory
- `clear all`
  - remove all data, value labels, matrices, scalars, constrains, clusters, stored results... from memory.

Versions of Stata newer than 12 have automatic memory management so you don't need to think about setting memory size.

Long output

Sometimes your command will produce output longer than the window. So it will look like this:

```
. list make price mpg

[ output omitted ]
 12. | Cad. Eldorado       14,500    14 |
 13. | Cad. Seville        15,906    21 |
 14. | Chev. Chevette       3,299    29 |
 15. | Chev. Impala         5,705    16 |
     |----------------------------------|
 16. | Chev. Malibu         4,504    22 |
--more--
```

- Pressing ⟨`Enter`⟩ show next line
- Pressing ⟨`Space`⟩: show next screen of output
- Typing ⟨`q`⟩: breaks (i.e. ask Stata to stop what it is doing)
- If you want stata to start showing all output instead of just what can fit on one screen use command `set more off`

What you should have learned

- To load and inspect data sets.
- Stata's command syntax
- Some useful commands: help, list, summarize, display

Useful commands

Summarize
Tabulate
Sort
label
describe
egen/gen
drop
list
regress
keep
count
rename