

This document is a basic guide to STATA. For the full list of options available for any command please check the STATA Manuals. More information is available for <http://rlab.lse.ac.uk/data>. If you have any questions or requests then please contact the Data Assistant, Gordon Knowles - [g.m.knowles@lse.ac.uk](mailto:g.m.knowles@lse.ac.uk), phone 020-7955-7806 or by visiting Zone 9, desk B on the 4<sup>th</sup> floor.

## CONTENT

STATA editions and their limits	P2
RLAB Access to STATA	P2
The STATA 8 Toolbar and Window	P2
Commands and Variables Windows	P3
Working Directory	P4
Command Interface	P4
File Extensions	P4
Opening Files	P4
<b>use</b>	P4
Memory	P4
<b>set mem</b>	P4
Saving Files	P5
<b>save</b>	P5
<b>saveold</b>	P5
Log Files	P5
<b>log using</b>	P5
<b>log on /off</b>	P5
<b>log close</b>	P5
annotating logs and program files	P5
Controlling output	P6
<b>more</b>	P6
<b>set more on/off</b>	P6
break	P6
Descriptive Commands	P5
<b>describe</b>	P6
<b>summarize</b>	P7
<b>list</b>	P8
arguments for use with descriptive commands	P7
-, *, ?	P7
<b>aorder</b>	P7
<b>in</b>	P7
Creating new variables	P7
<b>generate</b>	P8
<b>replace</b>	P9
<b>string variables</b>	P9
missing values	P9
Sort and By Commands	P10
<b>sort</b>	P10
checking unique ids	P10
<b>by</b>	P10
Cross tabulations	P11
STATA resources	P11

## STATA EDITIONS AND THEIR LIMITS

There are a number of different versions of STATA available, these are STATA SE (Special Edition), Intercooled STATA and Small STATA. STATA is available for all modern versions of Windows, and for UNIX and Macintosh.

### Limits for different editions of STATA

	STATA SE	Intercooled STATA	Small STATA
max. no. of variables	32,766	2,047	99
max no. of observations	2,147,483,647*	2,147,483,647*	1,000
max no. of characters for a string variable	244	80	80
matrices	1,000 x 1,000	800 x 800	40 x 40

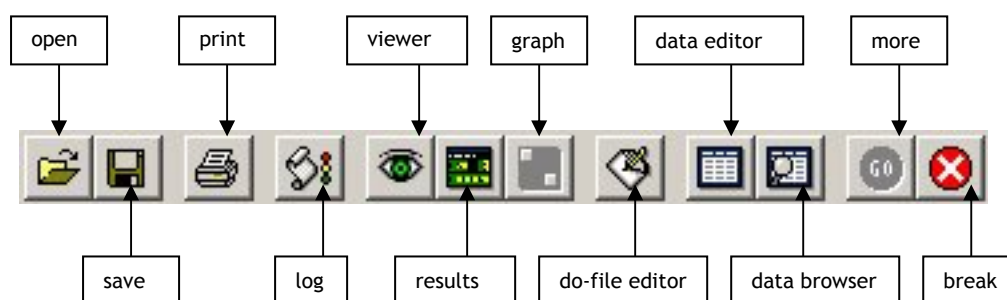
\* limited by memory

## RLAB ACCESS TO STATA

The LSE Research Laboratory currently runs STATA SE version 8.1 for Windows 2000. RLAB members have access to 40 network licenses.

If you have lost your link to STATA or still have STATA 7 installed, please go to the IT website <http://rlab.lse.ac.uk/itsupport/>, and download the shortcut.

## STATA 8 TOOLBAR and WINDOW



open: open a stata dataset.  
 save: save a dataset.  
 print: print contents of active window.  
 log: to start or stop, pause or resume a log file.  
 viewer: open viewer window, or bring to the front  
 results: open results window, or bring to the front.  
 graph: open graph window, or bring to the front.  
 do-file editor: open do-file editor, or bring window to the front.  
 data editor: open data editor, or bring window to the front.  
 data browser: open data browser, or bring window to the front.  
 more: command to continue when paused in long output.  
 break: stop the current task. This command returns the system to as it was before you issued the command.

The screenshot shows the Stata/SE 8.0 interface with several windows and callouts:

- Past commands appear here:** Points to the Review window, which contains a list of previously executed commands.
- Results appear here:** Points to the Stata Results window, which displays the output of the current command, including variable types and descriptions.
- Working directory displayed here:** Points to the bottom left corner of the Stata window, showing the current directory path.
- Variable list displayed here:** Points to the Variables window, which lists all variables in the current dataset.
- Displays destination of variables clicked in window below:** Points to the Stata Command window, which shows the command that will be executed when a variable is selected in the Variables window.
- Commands typed appear here:** Points to the Stata Command window, where new commands are entered.
- Log status appears here:** Points to the log on (text) button in the Stata Results window, which allows users to log the output of the current command.

## COMMANDS and VARIABLES

It is possible to scroll through **past commands** by using the **page up** and **page down** buttons on your keyboard. Alternatively you can double click on a command in the Review window and it will appear in your Command window. Similarly you can click on any variable that appears in the **Variables window** and they will appear in the Command window (or wherever the Target in the Variables window specifies).

## WORKING DIRECTORY

The working directory displayed at the bottom left hand corner of the window is your default directory. Any files you save without specifying a directory will be saved here. To change your working directory, use the cd command: **cd *directoryname***.

Note: You are advised to use the cd command at the beginning of your do-files and programs, this will save a lot of editing if the data you are using is moved.

## COMMAND INTERFACE

There have been some significant changes in **STATA 8**. One of the main ones is that it now has a Statistics Menu in the style of SPSS. This enables the user to select an item from a pull down menu which opens a dialogue box in which you can build STATA commands. I will not go into detail on how to use this method of analysing data as I would encourage users to learn the commands so that they can write do-files and programs.

However, one point that may be useful: The command issued by the dialogue box is submitted as if you typed it by hand. Therefore if you cannot remember the syntax of a command, using the dialogue box and then checking the command in the Review window is a good way to get a reminder.

## FILES EXTENTIONS

Data file	<i>filename.dta</i>	
Do file	<i>filename.do</i>	(program file)
Dictionary file	<i>filename.dct</i>	
Log file	<i>filename.scml</i>	(only readable in stata)
Log file	<i>filename.log</i>	(text file)

## OPENING FILES

Most of the commands discussed below can also be run from the toolbar or the menus, however in this document I will be discussing the syntax of typed commands.

To open a file:

**use filename, clear**  
**use varlist using filename, clear** [for a subset of the data file]

In some cases you may get the message **no room to add more observations** or **no room to add more variables**. This is because not enough memory has been assigned to STATA.

## MEMORY

To change the memory assigned to STATA:

**set mem #k**

where # is a number greater than the size of the dataset, and less than the total amount of memory available on your system.

To check the size of the dataset, look in **My Computer** or your Explorer package. To check the amount of memory (RAM) your system has available, go to the Start menu and click on \Settings\Control Panel\System. The bottom line, under General tells you how many KB of RAM you have available.

STATA opens with a default memory of 1mg. To increase the default memory:

Right click on the STATA icon and choose Properties\Shortcut  
Edit the **Target** field to say: [\\St-server5\stata8\\$\wsestata.exe /k#](#)

Where k# is the number of kb you wish to assign to STATA.

Note: If you do not have enough memory available on your machine to read a whole dataset, open a subset of the variables you need.

## SAVING FILES

To save a datafile:

<b>save, replace</b>	[overwrites current file]
<b>save filename, replace</b>	[saves file as <i>filename</i> . Replace is optional, but necessary if a file of that name already exists]

**saveold filename, intercooled replace** [to save a file in STATA 7 format]

## LOG FILES

All output appearing in the Results window can be captured in a log file. The log file can be saved as a STATA formatted (SMCL) or as a text (ASCII) file.

By default, logs are written in SMCL (Stata Markup and Control Language). However, logs written in SMCL can only be read and printed from the Viewer as other packages cannot read SMCL.

To start a log:

<b>log using filename</b>	[starts an smcl log]
<b>log using filename, replace</b>	[overwrites filename.smcl]
<b>log using filename.log</b>	[starts a text log]

Note: to translate a log file created in smcl to text, go to \File\Log\Translate

To pause and resume a log:

**log off** [temporarily suspends log file]  
**log on** [resumes log file]

These commands can be useful to create a log that contains only results and not intermediate programming.

To close a log:

**log close** [closes current log file]

You can add comments to your log as you work by entering any comments in the command line (or in your do-file) preceded by a \*.


eg. \*unemployment rate

Any input preceded by a \* will not be read as a command.

## CONTROLLING OUTPUT

**-more-** may appear in your results window when you are trying to output a long listing.

To see the next line: press Enter  
To see the next screen: press any key or click on the **-more-** at the bottom of the results window  
**set more off /on** To switch the more command off/on

**break** To interrupt a STATA command at any time uses the **Break** button .

## DESCRIPTIVE COMMANDS

There are various ways of examining a dataset in STATA, including describe, list, and summarise.

**describe** produces a summary of the contents of a dataset

**d** [describes dataset in current memory]  
**d using filename** [describes a stored STATA format dataset]

you can also describe a subset of a dataset by specifying

**d varlist**

the output for the **describe** command looks like

```
-----
Contains data from L:\LICENSE DATA\L.F.S\raw data\LFS00Q1.dta
  obs:      142,941
  vars:      640                               16 Jul 2002 11:21
  size: 112,923,390 (44.9% of memory free)
-----
```

```
-----
      storage   display      value
variable name  type   format      label      variable label
-----
caseid         int    %8.0g              case id
remserno       long   %12.0g            part of hhold id
quota         int    %8.0g              stint number
-----
```

## SUMMARISE

**summarize** calculates and displays a variety of univariate statistics.

**su** [summarise whole dataset]  
**su varlist** [summarise subset varlist]

the output for this command looks like

```
-----
. su

Variable |      Obs      Mean   Std. Dev.   Min   Max
-----+-----
caseid   | 142941   115.7063   66.47398     1    223
remserno | 142941   7.75e+07   3.75e+07   1.01e+07  1.39e+08
quota   | 142941   115.7063   66.47398     1    223
week    | 142941   7.012628   3.729972     1    13
wlyr    | 142941   7.146368   3.639619     0     9
-----+-----
qrtr    | 142941   2.213123   1.168243     1     4
-----
```

you can also use summarize with the **detail** command, if you need more information about the shape of a dataset.

**su varlist, d**

here is the output for a detailed summary of the variable age

```
-----
. su age, d

                        age
-----
Percentiles      Smallest
1%                0          0
5%                3          0
10%               7          0      Obs          142941
25%              18          0      Sum of Wgt.  142941

50%              37
75%              55      Largest
90%              71          99      Mean          37.97934
95%              77          99      Std. Dev.     23.07457
99%              86          99      Variance      532.436
                        Kurtosis      2.059994
-----
```

## LIST

Finally the most detailed of the commonly used descriptive commands is `list`. `list` displays the values of variables by observation. If `varlist` is not specified the output will contain the value for every variable.

`l varlist`

### Arguments for use with descriptive commands:

Note: The examples below use the `describe` command however, these are standard arguments and as such can be used with all the descriptive commands explained above, except where otherwise stated.

`d numal-nvqhi` [describes all variables between `numal` and `nvqhi`. This will only be an alphabetical list if the variables are stored in alphabetical order]

Note: The command `aorder varlist` alphabetises `varlist` and moves it to the front of the dataset. If no `varlist` is specified all variables in the dataset are sorted in alphabetical order.

`d meth*` [describes all variables beginning with the string `meth`]

`d meth?1` [describes all variables beginning with `meth` and ending with `1`]

The commands listed below do not run with `describe`, and are meaningless when used with `summarise`.

`l in 3` [list the third observation]  
`l in -2` [list the second from last observation]  
`l in 1/3` [list observations 1 through 3]  
`l in 15/-3` [list observation 15 to third from last]

## CREATING NEW VARIABLES

The `generate` command is used to create a new variable. `generate` can create a new variable that is an algebraic expression of other variables.

`generate newvar = exp` [where `exp` is an algebraic expression]

To change the contents of an existing variable you must use the `replace` command.

`replace oldvar = exp`

For example:

1. to create a new variable `agerange` from an existing variable `age`.

```
g agerange = . if age<16           [where . is a missing value, see below]
replace agerange=1 if 16<=age & age<25
replace agerange=2 if 25<= age & age<45
etc.
```

2. to create a dummy variable `age16` identifying all 16 year olds in the dataset.

```
g age16=0
replace age16=1 if age==16       [note the == for an existing value]
```

### String variables:

Values for a string variable are denoted by inverted commas "".

For example

```
g age="young" if agerange==1
replace age="" if agerange~=1    [~= is not equal to]
```

Note: for more information on the operators available when creating new variables please see STATA manuals.

### Missing values:

The default code for a missing value in STATA is a single period (.) or a blank "" in the case of a string.

eg. **replace** var = . if var == 99  
**replace** string = "" if string == "not answered"

Stata 8 now has a range of 27 extended missing values, the . discussed above and .a, .b, ..., .z . These can be used to indicate why a value is unknown ie. not applicable, answer refused etc.

Note: if you are using extended missing values, it is no longer possible to test whether an expression is missing by typing ....if exp == .  
 In order to capture all missing values you should use the command ...if exp >= .

## SORT and BY COMMANDS

**Sort:** Arranges the observations of the current data into ascending order of the values of the variables of *varlist*.

The sort command is particularly useful in two situations.

1. when using the **by** *varlist:* prefix (discussed below) data must be sorted by *varlist*.

ie. **sort** *region*  
**by** *region: su income*

2. when merging datasets the data must be sorted by so that observations can be uniquely identified so that the merge command can match observations correctly.

ie. **u** *dataset1*  
**sort** *year region*  
**merge** *year region using dataset2*

Note: both datasets must be sorted.

To ensure that the *varlist* you are using uniquely identifies observations, use the following command

**sort** *var1 var2 var3*  
**l if** *var1==var1[\_n-1] & var2==var2[\_n-1] & var3==var3[\_n-1]*

if the *var1 var2* and *var3* sort the dataset uniquely there will be no observations listed.

**By:** The prefix **by** *varlist:* causes the command that follows to be repeated for each unique set of values of the variables in *varlist*.

For example the following output compares the average gross weekly wage by region.

```
----->
. by region: su grsswk
----->
region = tyne & weir
Variable |      Obs      Mean    Std. Dev.    Min    Max
-----+-----
  grsswk |      295    267.8034    173.9138     17   1250
-----+-----
-> region = inner london
Variable |      Obs      Mean    Std. Dev.    Min    Max
-----+-----
  grsswk |      484    452.9298    385.0805     6   3208
-----+-----
-> region = outer london
Variable |      Obs      Mean    Std. Dev.    Min    Max
-----+-----
  grsswk |     1120    387.2348    293.953     2   2308
----->
```

## CROSS TABULATIONS

The **tabulate** command is one of the basic analyses offered by STATA, it produces one- and two-way tables of frequency counts.

**tab** *varname1 varname2 [weight] [if exp]*

```
-----
. sort region
. by region: tab marstt
```

```
-> region = tyne & weir
```

marital status	Freq.	Percent	Cum.
single, never married	1,239	45.37	45.37
married, living with husband/wife	1,074	39.33	84.69
married, separated from husband/wife	58	2.12	86.82
divorced	160	5.86	92.68
widowed	200	7.32	100.00
Total	2,731	100.00	

```
-> region = inner london
```

marital status	Freq.	Percent	Cum.
single, never married	3,252	59.40	59.40
married, living with husband/wife	1,532	27.98	87.38
married, separated from husband/wife	170	3.11	90.48
divorced	262	4.79	95.27
widowed	259	4.73	100.00
Total	5,475	100.00	

## STATA RESOURCES

There are a **STATA manuals** distributed throughout the Research Lab. To find out where your nearest manual is go to STATA help section of the RLAB data website <http://rlab.lse.ac.uk/DataService/stata.asp>.

Also available on the RLAB site are the **LSE PhD STATA course notes** written by Arnaud Chevalier.

The data service also provides access to all recent **STATA journals** (back to 2000) through the library. We also have quite a good selection of back issues available in hard copy from the Data Manager's office R443.

Another useful site is the **support page of the STATA website**

<http://www.stata.com/support/>  
<http://www.stata.com/links/resources1.html>

including links to the very useful **UCLA STATA reference page**

<http://statcomp.ats.ucla.edu/stata/>