Introduction to Stata (part 2)

Biostat 511 - Fall 1999

Table of Contents pg.

Logical operators, functions ……. 1
Subsetting commands: by, if and in ……... 1
Missing values ……..... 2
More Graphics ……... 2
do files …….. 3
large files …….. 3

Logical operators, functions

The `if` command (see below) requires that you be able to write logical expressions such as

```
age>=8
```

which is read as "age greater than or equal to 8". These expressions can get fairly complex as in

```
(age>=8)&(sex==1)&(fev~=.)
```

which is read as "age greater than or equal to 8 and sex equal to 1 and fev not missing" (a period signifies a missing value in Stata) or more briefly as "males 8 and older with nonmissing fev values". The key operators that you need to know about to create these expressions are

>	greater than	~	not
<	less than	|	or
>=	greater than or equal to	&	and
<=	less than or equal to		
==	equal to (a double equal sign)		
~=	not equal to		

You have to be careful about order of evaluation in these expressions since the two expressions

```
((age>=8)|(sex=1))&(smoke==0)
(age>=8)|((sex=1)&(smoke==0))
```

give different results. Use parentheses () to make sure you get the order of evaluation that you want (the terms in parentheses get evaluated first).

There are a large number of functions built into Stata that you can use in creating variables (see `generate` and `replace` in the Introduction to Stata, part 1). A complete listing and documentation of these functions can be found in the online help under "functions". Here are a few examples

```
generate lblood = log10(blood)      * make a new variables equal to the log10 of blood
generate ratio = weight/height^2    * make a new variable equal to weight over height
                                    * squared
```

Subsetting commands: `by`, `if` and `in`

Many Stata commands are preceded by the "`by`" option and followed by the "`if`" and/or "`in`" options. The general form is

`by` `varlist: command … ` **`if`** `expression` **`in`** `range`

where `varlist` is a list of variables, `command` is a Stata command, `expression` is a logical expression and `range` is a range of observation numbers. `by` is used to repeat the command for all combinations of the values of the variables in `varlist`, as in

```
sort sex smoke
by sex smoke: summarize age – fev, detail * summarize age – fev by levels of sex and
                                          * smoking; give additional details about
                                  * each variable, including median and other percentiles
```

Note that the dataset must be sorted by the variables in `varlist` before you can use the `by` command.

`if` and `in` are used to restrict processing of the command. For instance,

```
summarize age - fev if sex == 0      * summarize age - fev for cases where sex = 0
tabulate smoke if age>=8             * frequency table of smoke for ages 8 and above
list in 1/10                         * list cases 1 through 10
```

Missing values

If you are using `infile` to read in your data, numeric missing values are entered as a period (.). Missing values for character strings are typically entered as an empty string (""). With `insheet`, missing values can just be omitted, as in

```
insheet make price mpg weight gratio
```

and the data file might look like

Datsun 810,8129,,2750,3.55
,4099,22,2930,3.58

Here, the third variable is missing in the first observation and the first variable is missing in the second observation.

Calculations on missing values using `generate` or `replace` yield a missing value for the result.

More Graphics

In part 1, we showed how to do a scatterplot and a boxplot. The other two types of plots you might commonly want to do are histograms and barcharts. These can be created with the `graph` command using the `hist` and `bar` options, respectively. Here are some examples:

```
graph fev, hist xlab ylab bin(8)     * histogram of fev with labels on x and y axis;
                                     * use 8 "bins" (categories) to make the histogram

graph race if status==1, bar xlab ylab   * bar chart of race with labels on x and y
                                         * axis; only include in the plot those subjects
                                         * with status equal to 1
```

Some common options that you might want to use with your graphs are documented in the help files for

```
graxes      * help on title and axis options
grsym       * help on symbol and line options
grcolor     * help on color and shading options
grother     * help on saving, printing and multiple images options
```

A few examples:

```
graph wt ht, xlab ylab bor gap(4) title("Figure 1. Scatterplot of height
vs weight") symbol(T)
```
* Do a scatterplot of height versus weight including labels on x and y axes and a border; set gap for the
* label on the y-axis to 4; add a title below the graph and make the plotting symbol a large triangle

```
graph fev, box ylab by(sex) t1title("Figure 2. Boxplot of fex by sex)
s(o) vwidth
```
* Do a boxplot of fev for each sex, add a title at the top, use small circles to indicate outliers and make
* the width of the box proporitonal to the sample size

do files

Often you will want to repeat a set of commands. Rather then typing the commands in every time, you can create a program to do what you want. In Stata, such programs are called "do- files". To create a do-file you simply type in a series of commands and save them in a text file (you can use Notepad or any other text editor for this purpose. If you use Word, be sure to save the file as a text-only file, not as a Word document. There is also an editor included with Stata - click on Do-file Editor under the "Window" menu.). Once you have saved a series of commands, you just type "`do`" and the filename in Stata. For example,

```
do "a:fevanalysis.txt"
```

Large files

If you are dealing with a large dataset, you may need to increase the amount of memory available to Stata. You can increase the amount of memory Stata uses by starting it with the /k option:

H:\stata\wstata.exe /k2000

will allocate 2000K (2 megabytes) to the Stata data area.