

Stata Brush Up Course

Session 6

Sergi Quintana

IDEA

September 5, 2024

Plan for the session

Programing with stata:

- ▶ Macros
- ▶ Loops and if conditions
- ▶ Programs
- ▶ Mata

Macros

A macro is a string of characters, called the macroname, that stands for another string of characters, called the macro contents.

Macros can be local or global.

Stata forgets about the locals every time the code stops running.
Stata does remember the globals for all our session.

Loops

forvalues

Syntax

```
forvalues lname = range {  
    Stata commands referring to 'lname'  
}
```

where *range* is

$\#_1(\#_d)\#_2$	meaning $\#_1$ to $\#_2$ in steps of $\#_d$
$\#_1/\#_2$	meaning $\#_1$ to $\#_2$ in steps of 1
$\#_1 \#_t \text{ to } \#_2$	meaning $\#_1$ to $\#_2$ in steps of $\#_t - \#_1$
$\#_1 \#_t : \#_2$	meaning $\#_1$ to $\#_2$ in steps of $\#_t - \#_1$

The loop is executed as long as calculated values of '*lname*' are $\leq \#_2$, assuming that $\#_d > 0$.

Braces must be specified with **forvalues**, and

1. the open brace must appear on the same line as **forvalues**;
2. nothing may follow the open brace except, of course, comments; the first command to be executed must appear on a new line;
3. the close brace must appear on a line by itself.

Loops

foreach

Syntax

```
foreach lname { in | of listtype } list {  
    Stata commands referring to 'lname'  
}
```

Allowed are

```
foreach lname in any_list {
```

```
foreach lname of local lname {
```

```
foreach lname of global lname {
```

```
foreach lname of varlist varlist {
```

```
foreach lname of newlist newvarlist {
```

```
foreach lname of numlist numlist {
```

Loops

while

Syntax

```
while exp {  
    stata_commands  
}
```

Braces must be specified with `while`, and

1. the open brace must appear on the same line as `while`;
2. nothing may follow the open brace, except, of course, comments; the first command to be executed must appear on a new line;
3. the close brace must appear on a line by itself.

Loops

if or else

Syntax

```
if exp {                               or    if exp single_command  
    multiple_commands  
}
```

which, in either case, may be followed by

```
else {                                   or    else single_command  
    multiple_commands  
}
```

If you put braces following the `if` or `else`,

1. the open brace must appear on the same line as the `if` or `else`;
2. nothing may follow the open brace except, of course, comments; the first command to be executed must appear on a new line;
3. the close brace must appear on a line by itself.

Loops

continue

Syntax

```
continue [ , break ]
```

Description

The `continue` command within a `foreach`, `forvalues`, or `while` loop breaks execution of the current loop iteration and skips the remaining commands within the loop. Execution resumes at the top of the loop unless the `break` option is specified, in which case execution resumes with the command following the looping command. See [\[P\] `foreach`](#), [\[P\] `forvalues`](#), and [\[P\] `while`](#) for a discussion of the looping commands.

Programs

Stata programming is an advanced topic. Some Stata users live productive lives without ever programming Stata.

Stata has two programming languages:

- ▶ One, known informally as “ado”. It is based on Stata’s commands, and you can write scripts and programs to automate reproducible analyses and to add new features to Stata.
- ▶ The other language, Mata, is a byte-compiled language with syntax similar to C/C++, but with extensive matrix capabilities. The two languages can interact with each other. You can call Mata functions from ado-programs, and you can call ado-programs from Mata functions.

For the full documentation visit [here](#).

Programs

When you type a command that Stata does not recognize, Stata first looks in its memory for a program of that name. If Stata finds it, Stata executes the program.

There is no Stata command named `hello`,

```
. hello
command hello is unrecognized
r(199);
```

but there could be if you defined a program named `hello`, and after that, the following might happen when you typed `hello`:

```
. hello
hi there
. _
```

This would happen if, beforehand, you had typed

```
. program hello
1. display "hi there"
2. end
. _
```

That is how programming works in Stata. A program is defined by

```
program progrname
    Stata commands
end
```

and it is executed by typing *progrname* at Stata's dot prompt.

Mata

Mata is a matrix programming language that can be used by those who want to perform matrix calculations interactively and by those who want to add new features to Stata.

Exercise - Mata

Using the auto data set, estimate the following regression using mata commands:

$$mpg_j = \beta_0 + \beta_1 weight_j + \beta_2 foreign_j + \epsilon_j$$

Include as an output the estimates for the coefficients and the standard error.