Introduction to Stata

LPO 9951 | Fall 2015

PURPOSE In this class we'll walk through some of Stata's basic functionality. We'll also get used to the idea of interacting with Stata through the command line and `*.do` files.

Stata as a calculator

Stata can be used as a calculator via the `display` command. All of the normal rules of arithmetical precedence apply to the Stata syntax.

```
. display sqrt(42)
6.4807407

. di sqrt(42) + 4
10.480741

. di (sqrt(42) + 4) - 10
.4807407
```

NB: The `display` command can be shortened to `di`. Many Stata commands and options are like this. The help files underline the minimum part of the command/option that must be specified in order for the package to understand what you want.

Using `*.do` files

Stata syntax is stored in what's called `*.do` files. These are all of the typed commands that you use to manipulate and analyze the data. A properly formatted `*.do` file can be run from the command line using the `do` command:

```
. do "../do/lecture1_introduction_hello.do"

. display "Hello, World!"
Hello, World!

end of do-file
```

One of the key skills you'll learn this year is properly annotating a `*.do` file. Remember that these files are primarily meant to be read by humans, and only incidentally meant to be read by computers. Stata assumes that everything in a `*.do` file is a command unless it's preceded by a comment sign. To set off a line of text as a comment, place `*` or `//` in front of it. You can also use the `/* ... */` format for comments, which can be used on the same line as the syntax itself:

```
. * comments can be on their own rows...
. // ...like this
. /* ...and this, or */
. di 1           // to the
1

. di 2           /* side of commands */
2
```

Directory structure

We'll talk about directory structure more detail later, but for now, make sure that your course files have at least the following structure:

```
.
|-- /data
|    |
|    |-- <data files>
|
|-- /do
|    |
|    |-- <Stata do files>
|
|-- /plots
|    |
|    |-- <plot files>
```

Place your Stata do files in the `./do` subdirectory, all data files in the `./data` folder, and all saved graphics in the `./plots`. We'll add more as the semester goes on but these will do for now. The primary directory (represented by the `.`) can be anywhere on your computer or a thumbdrive. What really matters are the relative paths between the subfolders. Just make sure that wherever you choose to hold your course files you have enough storage space. While do files are usually very small, some of our datasets will be fairly large.

Loading Stata data files

All Stata data files are saved in the `*.dta` format. Today we'll be using the `census.dta` file which contains information on characteristics of the 50 states from the 1980 census.

To locate a data file, you first have to tell Stata where to look on your computer. With some very rare exceptions, you should always use the `cd` command to set the working directory:

```
. cd "~/Github/lpo9951/markdown"
/Users/benski/Github/lpo9951/markdown
```

NB: The exception to this rule would be (1) when you double-click your `*.do` file and have Stata configured to open automatically; and (2) your `*.do` file is set to work in the directory in which it is currently located (i.e., all the relative links are correctly specified).

The above directory is where I keep the class files, hence the `cd` command doesn't really do anything. Your files will be in a different location on your computer. Changing the working directory just once makes it *much* easier to exchange `*.do` files across computers. Don't place a `cd` command in your `*.do` file. This will make collaboration much easier.

To open a Stata file, use the `use` command:

```
. use "../data/census.dta", clear
(1980 Census data by state)

.
```

We'll go over for other commands for importing more complex data files later.

Looking at the data

list

We can use the `list` command to take a look at the data:

```
. list

     +-------------------------------------------------------------------+
  1. | state          | region    |         pop  |     poplt5  |    pop5_17  |
     | Alabama        | South     |   3,893,888  |    296,412  |    865,836  |
     |-------------------------------------------------------------------|
     |     pop18p |     pop65p |   popurban | medage |    death | marriage |
     |  2,731,640 |    440,015 |  2,337,713 |  29.30 |   35,305 |   49,018 |
     |-------------------------------------------------------------------|
     |                              divorce                              |
     |                               26,745                              |
     +-------------------------------------------------------------------+

     +-------------------------------------------------------------------+
  2. | state          | region    |         pop  |     poplt5  |    pop5_17  |
     | Alaska         | West      |     401,851  |     38,949  |     91,796  |
     |-------------------------------------------------------------------|
     |     pop18p |     pop65p |   popurban | medage |    death | marriage |
     |    271,106 |     11,547 |    258,567 |  26.10 |    1,604 |    5,361 |
     |-------------------------------------------------------------------|
     |                              divorce                              |
     |                                3,517                              |
     +-------------------------------------------------------------------+

...and so on.
```

describe

As should be obvious, this usually gives too much information back. A better place to start with a well-formatted data file is to use the `describe` command:

```
. describe

Contains data from ../data/census.dta
  obs:            50                          1980 Census data by state
 vars:            12                          28 Aug 2012 11:20
 size:         2,800
-------------------------------------------------------------------------------
              storage   display    value
variable name   type    format     label      variable label
-------------------------------------------------------------------------------
state           str14   %-14s                 State
region          int     %-8.0g     cenreg     Census region
pop             long    %12.0gc               Population
poplt5          long    %12.0gc               Pop, < 5 year
pop5_17         long    %12.0gc               Pop, 5 to 17 years
pop18p          long    %12.0gc               Pop, 18 and older
```

```
pop65p          long    %12.0gc               Pop, 65 and older
popurban        long    %12.0gc               Urban population
medage          float   %9.2f                 Median age
death           long    %12.0gc               Number of deaths
marriage        long    %12.0gc               Number of marriages
divorce         long    %12.0gc               Number of divorces
-------------------------------------------------------------------------------
Sorted by:
```

`codebook`

To get more information about a single variable, the`codebook` command is a good option:

```
. codebook pop

-------------------------------------------------------------------------------
pop                                                                  Population
-------------------------------------------------------------------------------

                  type:  numeric (long)

                 range:  [401851,23667902]           units:  1
         unique values:  50                      missing .:  0/50

                  mean:   4.5e+06
              std. dev:   4.7e+06

           percentiles:        10%       25%       50%       75%       90%
                            671743   1.1e+06   3.1e+06   5.5e+06   1.1e+07
```

`list` with `if` statement

If I add the condition `if _n < 11`, I can see data for only the first ten states. `_n` represents the row number of each observation. Since the states are in alphabetical order in the dataset, I can use the logical statement `_n < 11` to get the first ten:

```
. list if _n < 11

     +-------------------------------------------------------------------------+
  1. | state       | region |        pop |    poplt5 |   pop5_17 |     pop18p |
     | Alabama     | South  |  3,893,888 |   296,412 |   865,836 |  2,731,640 |
     |-------------------------------------------------+-----------------------|
     |    pop65p   |   popurban  | medage   |   death   |  marriage  |  divorce  |
     |   440,015   |  2,337,713  |  29.30   |  35,305   |    49,018  |   26,745  |
     +-------------------------------------------------------------------------+

     +-------------------------------------------------------------------------+
  2. | state       | region |        pop |    poplt5 |   pop5_17 |     pop18p |
     | Alaska      | West   |    401,851 |    38,949 |    91,796 |    271,106 |
     |-------------------------------------------------+-----------------------|
     |    pop65p   |   popurban  | medage   |   death   |  marriage  |  divorce  |
     |    11,547   |    258,567  |  26.10   |   1,604   |     5,361  |    3,517  |
```

```
   +------------------------------------------------------------------------+

   +------------------------------------------------------------------------+
3. | state       | region |        pop |    poplt5 |   pop5_17 |     pop18p |
   | Arizona     | West   |  2,718,215 |   213,883 |   577,604 |  1,926,728 |
   |-----------------------------------------------+------------------------|
   |    pop65p |   popurban | medage |      death  |  marriage  |  divorce  |
   |   307,362 |  2,278,728 |  29.20 |     21,226  |    30,223  |   19,908  |
   +------------------------------------------------------------------------+

   +------------------------------------------------------------------------+
4. | state       | region |        pop |    poplt5 |   pop5_17 |     pop18p |
   | Arkansas    | South  |  2,286,435 |   175,592 |   495,782 |  1,615,061 |
   |-----------------------------------------------+------------------------|
   |    pop65p |   popurban | medage |      death  |  marriage  |  divorce  |
   |   312,477 |  1,179,556 |  30.60 |     22,676  |    26,513  |   15,882  |
   +------------------------------------------------------------------------+

   +------------------------------------------------------------------------+
5. | state       | region |        pop |    poplt5 |   pop5_17 |     pop18p |
   | California  | West   | 23,667,902 | 1,708,400 | 4,680,558 | 17,278,944 |
   |-----------------------------------------------+------------------------|
   |    pop65p |   popurban | medage |      death  |  marriage  |  divorce  |
   | 2,414,250 | 21,607,606 |  29.90 |    186,428  |   210,864  |  133,541  |
   +------------------------------------------------------------------------+

   +------------------------------------------------------------------------+
6. | state       | region |        pop |    poplt5 |   pop5_17 |     pop18p |
   | Colorado    | West   |  2,889,964 |   216,495 |   592,318 |  2,081,151 |
   |-----------------------------------------------+------------------------|
   |    pop65p |   popurban | medage |      death  |  marriage  |  divorce  |
   |   247,325 |  2,329,869 |  28.60 |     18,925  |    34,917  |   18,571  |
   +------------------------------------------------------------------------+

   +------------------------------------------------------------------------+
7. | state       | region |        pop |    poplt5 |   pop5_17 |     pop18p |
   | Connecticut | NE     |  3,107,576 |   185,188 |   637,731 |  2,284,657 |
   |-----------------------------------------------+------------------------|
   |    pop65p |   popurban | medage |      death  |  marriage  |  divorce  |
   |   364,864 |  2,449,774 |  32.00 |     26,005  |    26,048  |   13,488  |
   +------------------------------------------------------------------------+

   +------------------------------------------------------------------------+
8. | state       | region |        pop |    poplt5 |   pop5_17 |     pop18p |
   | Delaware    | South  |    594,338 |    41,151 |   125,444 |    427,743 |
   |-----------------------------------------------+------------------------|
   |    pop65p |   popurban | medage |      death  |  marriage  |  divorce  |
   |    59,179 |    419,819 |  29.80 |      5,123  |     4,437  |    2,313  |
   +------------------------------------------------------------------------+

   +------------------------------------------------------------------------+
9. | state       | region |        pop |    poplt5 |   pop5_17 |     pop18p |
   | Florida     | South  |  9,746,324 |   570,224 | 1,789,412 |  7,386,688 |
   |-----------------------------------------------+------------------------|
```

```
     |    pop65p  |    popurban  | medage  |   death  |  marriage  |  divorce  |
     | 1,687,573  |  8,212,385  |  34.70  | 104,190  |   108,344  |   71,579  |
     +------------------------------------------------------------------------+

     +------------------------------------------------------------------------+
 10. | state       | region |        pop |    poplt5 |   pop5_17 |     pop18p |
     | Georgia     | South  |  5,463,105 |   414,935 | 1,231,195 |  3,816,975 |
     |-----------------------------------------------+------------------------|
     |    pop65p  |    popurban  | medage  |   death  |  marriage  |  divorce  |
     |   516,731  |  3,409,081  |  28.70  |  44,230  |    70,638  |   34,743  |
     +------------------------------------------------------------------------+
```

Most of the time, I only want to see a couple of variables. In this case, I'll use `list` with what Stata calls a `varlist` and is in fact just a list of variables. In this case, I only choose `state` and `pop`.

```
. li state pop if _n < 11

     +--------------------------+
     | state                pop |
     |--------------------------|
  1. | Alabama        3,893,888 |
  2. | Alaska           401,851 |
  3. | Arizona        2,718,215 |
  4. | Arkansas       2,286,435 |
  5. | California    23,667,902 |
     |--------------------------|
  6. | Colorado       2,889,964 |
  7. | Connecticut    3,107,576 |
  8. | Delaware         594,338 |
  9. | Florida        9,746,324 |
 10. | Georgia        5,463,105 |
     +--------------------------+
```

QUICK EXERCISE

Take a look at deaths in the first 10 states. Which is highest, which is lowest?

Recoding variables

To start off with, I'm interested in knowing which states have the largest proportion of the population under 5. The data only give the total number of people under 5, so I'm going to need a new variable, which will be total population under 5 divided by total population. To create this variable I'll need Stata's `generate` command:

```
. generate poplt5_pr = poplt5 / pop
```

NB: Stata will not allow you to generate a new variable with an old variable's name. `generate poplt5 = poplt5 / pop` will not work because you already have a `poplt5` variable. This is a feature to make sure you don't overwrite your data accidentally.

Summarizing data

Now that I have my new variable, let's use the summarize command to take a look at it:

```
. summarize poplt5_pr

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
   poplt5_pr |        50     .075981    .0119612   .0585066   .1300186
```

This is nice, but if I'd like even more information I should use the `detail` subcommand, like so:

```
. sum poplt5_pr, detail

                          poplt5_pr
-------------------------------------------------------------
      Percentiles      Smallest
 1%     .0585066       .0585066
 5%     .0595924       .0587786
10%     .0629542       .0595924       Obs                  50
25%     .0698113       .0598551       Sum of Wgt.          50

50%     .0750633                      Mean            .075981
                        Largest       Std. Dev.      .0119612
75%     .0786851       .0955049
90%     .0870086        .096924       Variance       .0001431
95%      .096924       .0990863       Skewness        1.96293
99%     .1300186       .1300186       Kurtosis       9.799138
```

QUICK EXERCISE

> Create a variable for the proportion of the population living in urban areas. Use `summarize` to describe your new variable. What's the mean and median of your new variable?

Using the by and bysort commands

Many times we'd like to summarize a variable by subgroups in the data. For instance, what if we'd like to know which regions have the highest proportions of children under 5? We could try to use the `by` command like this, `by region: sum poplt5_pr`, but it won't work. Stata will refuse to run it because the data are not sorted on the `region` variable. However, the `bysort` command gives us an easy way around that problem:

QUICK EXERCISE

> Create a table of proportion urban by region. Which region has the highest proportion of people living in cities?

Univariate graphics

histogram

To describe a data point, we can use the `histogram` command. If we want to save the plot, we use the `graph export` command:

```
. histogram poplt5_pr, name(h_poplt5_pr)
(bin=7, start=.05850657, width=.01021601)

. graph export "../plots/h_poplt5_pr.eps", name(h_poplt5_pr) replace
(note: file ../plots/h_poplt5_pr.eps not found)
(file ../plots/h_poplt5_pr.eps written in EPS format)
```

histogram with by

You can combine the histogram command with a `by` command to show the distribution of a variable by groups:

```
. histogram poplt5_pr, by(region) name(h_poplt5_pr_reg)

. graph export "../plots/h_poplt5_pr_reg.eps", name(h_poplt5_pr_reg) replace
(note: file ../plots/h_poplt5_pr_reg.eps not found)
(file ../plots/h_poplt5_pr_reg.eps written in EPS format)
```

kdensity

You can also use the `kdensity` command to describe the data using a kernel density plot:

```
. kdensity poplt5_pr, name(kd_poplt5_pr)

. graph export "../plots/kd_poplt5_pr.eps", name(kd_poplt5_pr) replace
(note: file ../plots/kd_poplt5_pr.eps not found)
(file ../plots/kd_poplt5_pr.eps written in EPS format)
```

QUICK EXERCISE

List state name and population less than 5 if population less than 5 is greater than .1

Bivariate graphics

scatterplot

A scatterplot is a very useful tool for the looking at the relationship between two (or more) variables. Right now I'd like to look at the relationship between the number of children under 5 and the number of people over 65. The variable `pop65p` is not a proportion, so I need to generate a new proportion variable to get them both on the same scale:

```
. gen pop65p_pr = pop65p / pop
```

With my new variable, I can now create a scatterplot:

```
. gen pop65p_pr = pop65p / pop

. // scatterplot of young population as a function of older population
. graph twoway scatter poplt5_pr pop65p_pr, name(sc_poplt5_pr)
```

```
. graph export "../plots/sc_poplt5_pr.eps", name(sc_poplt5_pr) replace
(note: file ../plots/sc_poplt5_pr.eps not found)
(file ../plots/sc_poplt5_pr.eps written in EPS format)
```

We can add state labels:

```
. graph twoway scatter poplt5_pr pop65p_pr, ///
>     msymbol(none) mlabel(state) name(sc_poplt5_pr_1)

. graph export "../plots/sc_poplt5_pr_1.eps", name(sc_poplt5_pr_1) replace
(note: file ../plots/sc_poplt5_pr_1.eps not found)
(file ../plots/sc_poplt5_pr_1.eps written in EPS format)
```

The labels are too big. We can make them smaller.

```
. graph twoway scatter poplt5_pr pop65p_pr, ///
>     msymbol(none) mlabel(state) mlabsize (tiny) name(sc_poplt5_pr_2)

. graph export "../plots/sc_poplt5_pr_2.eps", name(sc_poplt5_pr_2) replace
(note: file ../plots/sc_poplt5_pr_2.eps not found)
(file ../plots/sc_poplt5_pr_2.eps written in EPS format)
```

EXERCISES

1. Create variables for rate of marriages and divorces
2. Which region has the highest rates of marriage and divorce in the population?
3. What do the distributions of these two variables look like?
4. What does a scatterplot say about the possible relationship between the two?

Init: 07 June 2015; Updated: 16 August 2015