Introduction to Stata

September 23, 2014

Stata is one of a few statistical analysis programs that social scientists use. Stata is in the mid-range of how easy it is to use. Other options include SPSS, considered easier to use, but clunky if performing many commands, SAS, and R.

Stata features a primarily command line interface. What that means is you type in commands into the command field, and when you press enter the results of the command appear in the results field (as opposed to using menu options, the primarily interface for SPSS). This requires learning what the commands are and how to use them, making the initial learning curve a bit steep. But once you've learned the basic commands, it becomes easy to work with your dataset.

For this introduction, we will be using a sample of data from the General Social Survey, a popular dataset based on a national survey given approximately every two years.

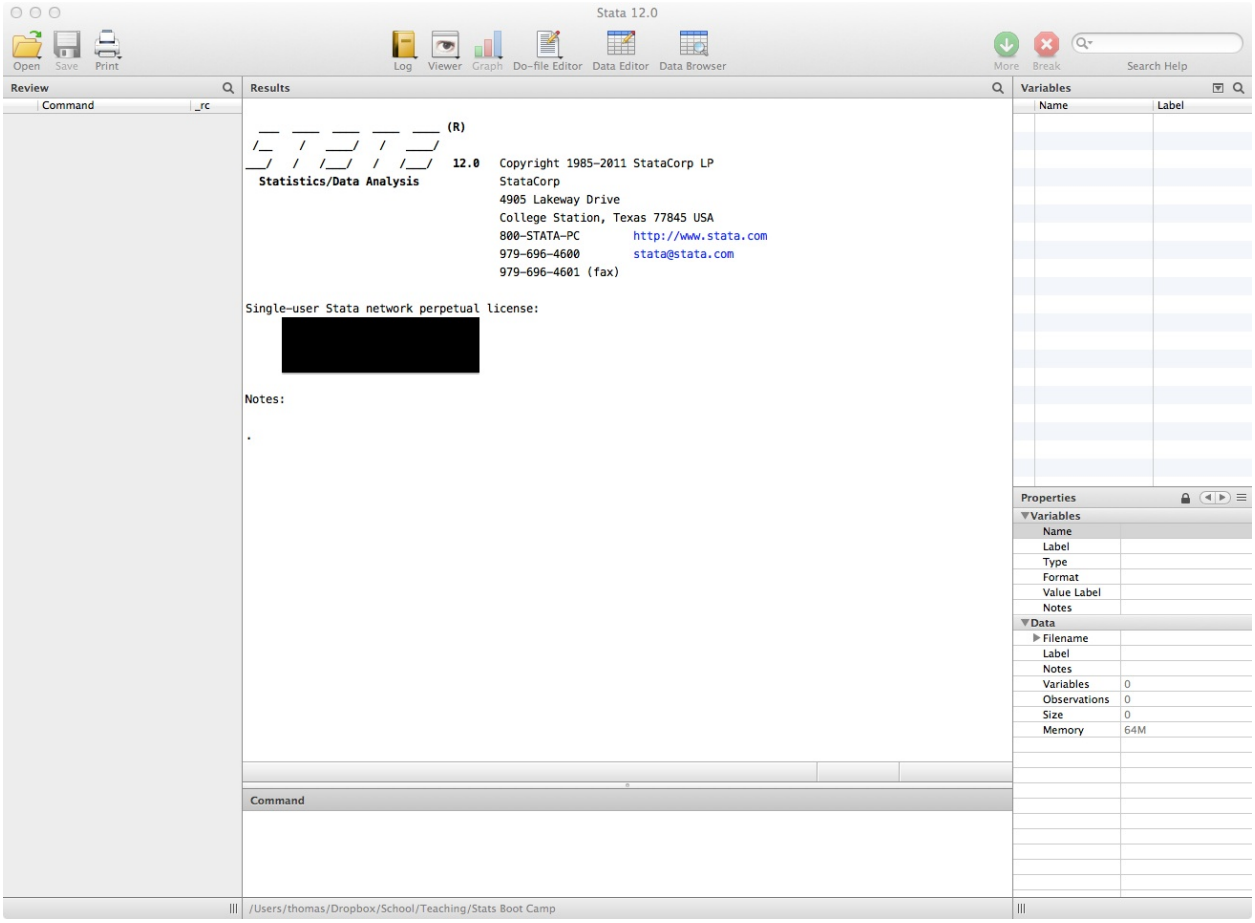
1 Starting Stata

When you first start up Stata, you will be presented with the main window. Figure 1 shows what this looks like. On the left is the command history, which will list all the previous commands given during the current session. On the right is the variable list. When you load up a dataset, the available variables will be listed in the top, and the bottom will contain information about the variable currently selected in the list. In between is the results and command fields. The command field, located at the bottom, is where you type commands to tell Stata what to do. The results of those commands appear as plain text in the results field.

A few things to note:

1. Stata's variable names are case sensitive. So the variable `AGE` is different than the variable `Age`, which is different than the variable `age`.
2. Stata allows commands and options to be shortened so you don't need to type the full name of the command - just enough for Stata to know which one you are referring to. In Stata's help pages, the part of the command underlined is the minimum you need to type for Stata to recognize the command.

Figure 1: Stata



3. This shortening also applies to variable names. As long as you type enough of a variable name to distinguish it from the other variables, Stata will know which one you are talking about.

1.1 Opening and Saving Datasets

To open a dataset, we use the `use` command. But first, we need to tell Stata where the dataset is saved on the hard drive. We do this by setting our working directory to the directory containing the dataset using the `cd` command (for change directory):

```
cd "~/Documents/Stats"
```

```
cd "C:\Users\Thomas\Documents\Stats"
```

You can also go to the File menu and select Change Working Directory for a directory menu.

After you have changed the working directory, you should see the current working directory in the status bar along the bottom of the Stata window.

You can now load up a dataset. Stata datasets are stored as `.dta` files. Open the dataset using the `use` command:

```
use gss.dta
```

You'll see the command appear in the results window. If there was an error, it will also be reported in the results window. If Stata successfully opened the file, you'll see the variables in the dataset listed on the right hand side.

To save a dataset, you use the `save` command:

```
save gss.dta, replace
```

Notice the `, replace` in the command - this is an option. In Stata, command options are included after a comma and modify the command in specific ways. We'll see more options later. Here, the replace option tells Stata to replace any current files named `gss.dta`, if they exist. If you didn't include the replace, Stata would give you an error, saying the filename already exists.

Now that you know how to open and save datasets, let's explore the data.

2 Central Tendency and Variance

2.1 Measures of Central Tendency

The first things we learned earlier this week were measures of central tendency, including mean, median, and mode. Stata can easily compute these for us for any interval or ratio level variable in the dataset. To calculate the mean of a variable, we can use the `summarize` command, which can be shortened to `sum`

```
. sum AGE

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
         AGE |      2041    47.96717    17.67799         18         89
```

So the mean age in this sample is 47.97 years.

To get the median, we use the same command but include the `details` option:

```
. sum AGE, d

                        AGE OF RESPONDENT
-------------------------------------------------------------
      Percentiles      Smallest
 1%           19             18
 5%           22             18
10%           25             18       Obs                2041
25%           33             18       Sum of Wgt.        2041

50%           47                      Mean           47.96717
                        Largest       Std. Dev.      17.67799
75%           61             89
90%           72             89       Variance       312.5112
95%           80             89       Skewness       .2921034
99%           88             89       Kurtosis       2.234348
```

The 50th percentile is the median, so in this case the median age is 47.

Stata does not have a command to calculate the mode, though rarely do people care about the mode so this usually isn't a problem.

2.2 Variance and Standard Deviation

You may have noticed that the `summarize` command also calculates the standard deviation, and variance with the details option. Looking at the results above, we see that the standard deviation for AGE is 47.97 years and the variance is 312.51 years squared.

We can list more than one variable at a time in the summarize command:

```
. sum AGE EDUC AGEKDBRN

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
         AGE |      2041    47.96717    17.67799         18         89
        EDUC |      2039    13.46101    3.149267          0         20
    AGEKDBRN |      1470    23.91633    6.022915         12         55
```

This makes generating tables of descriptive statistics super easy.

2.3 Proportions

Proportions can be a little tricky to work with, depending on how they are coded. To use Stata to analyze proportions, we need them coded as 0s and 1s. We can see how variables are coded in Stata using the `codebook` command:

```
. codebook SEX

-------------------------------------------------------------------------------
SEX                                                          RESPONDENTS SEX
-------------------------------------------------------------------------------

                  type:  numeric (int)
                 label:  SEX

                 range:  [1,2]                        units:  1
          unique values:  2                       missing .:  0/2044

             tabulation:  Freq.   Numeric  Label
                           891         1  MALE
                          1153         2  FEMALE
```

So according to the codebook command, the SEX variable is coded as 1 for male, 2 for female. If we wanted to work with proportions of men, we need to recode this variable as 1 for male, 0 for female. We can do that with the `recode` command:

```
. recode SEX (2=0), gen(male)
(1153 differences between SEX and male)

. codebook male

-------------------------------------------------------------------------------
male                                              RECODE of SEX (RESPONDENTS SEX)
-------------------------------------------------------------------------------

                  type:  numeric (int)

                 range:  [0,1]                        units:  1
          unique values:  2                       missing .:  0/2044

             tabulation:  Freq.  Value
                          1153  0
                           891  1
```

In the recode command, we told Stata we wanted to recode SEX into a new variable called male (the `gen(male)` option), and that we wanted the new variable to transform all the 2s to 0s. The other values will remain the same. The codebook command for the new variable male shows that the number of 1s is the same as the SEX variable, and that the number of 0s is the same as the number of 2s in the SEX variable. You'll notice that the new variable does not have value labels (MALE, FEMALE) - we would need to do that manually, but its not necessary.

Now that we have our male variable, we can use the `summarize` command to calculate the proportion male:

```
. sum male

    Variable |       Obs        Mean    Std. Dev.       Min        Max
-------------+--------------------------------------------------------
        male |      2044      .43591    .4959968          0          1
```

Recall that the mean of a variable coded 0/1 is the proportion of 1s. Here, 1 stands for male, so the mean of the variable is the proportion of males in the dataset, which is .436, or 43.6%. Note, though, that the standard deviation reported here assumes this is a mean - this is not the proportion standard deviation. Recall that the proportion standard deviation is:

$$s = \sqrt{\pi(1-\pi)}$$

So in this case, it would be

$$s = \sqrt{0.43591(1-0.43591)} = \sqrt{0.2559} = 0.495875$$

As you can see, this is very close to the standard deviation reported by the summarize command, but not exactly the same. The command for calculating confidence intervals for proportions will calculate the correct standard error for us.

3 Confidence Intervals

3.1 Means

The command for calculating confidence intervals in Stata is `ci`

```
. ci AGE

    Variable |        Obs        Mean    Std. Err.       [95% Conf. Interval]
-------------+---------------------------------------------------------------
         AGE |       2041    47.96717    .3913013        47.19978    48.73456
```

As you can see, the command calculates the mean and standard error, along with the confidence interval. By default, the command calculates the 95% confidence interval, but we can use the `level()` option to specify a different level:

```
. ci AGE, level(99)

    Variable |        Obs        Mean    Std. Err.       [99% Conf. Interval]
-------------+---------------------------------------------------------------
         AGE |       2041    47.96717    .3913013         46.9583    48.97604
```

We can be 95% confident that the true population mean age is between 47.19978 and 48.73456 years old. We can be 99% confident that the true population mean age is between 46.9583 and 48.97604 years old.

3.2 Proportions

To tell Stata that we are using proportions instead of means, we include the `binomial` option:

```
. ci male, b

                                                        -- Binomial Exact --
    Variable |        Obs        Mean    Std. Err.       [95% Conf. Interval]
-------------+------------------------------------------------------------
        male |       2044      .43591    .0109681        .4142732    .4577308
```

This makes sure that Stata calculates the standard error correctly for proportions.

We can be 95% confident that the true population proportion of men is between 0.4142732 and 0.4577308.

4 Hypothesis Testing

4.1 Means

Hypothesis testing is very easy in Stata using the `ttest` command. The command take the form:

`ttest` *varname* `= #`

```
. ttest AGE = 47

One-sample t test
------------------------------------------------------------------------------
Variable |     Obs        Mean    Std. Err.   Std. Dev.   [95% Conf. Interval]
---------+--------------------------------------------------------------------
     AGE |    2041    47.96717    .3913013    17.67799    47.19978    48.73456
------------------------------------------------------------------------------
    mean = mean(AGE)                                              t =   2.4717
Ho: mean = 47                                  degrees of freedom =     2040

    Ha: mean < 47               Ha: mean != 47                 Ha: mean > 47
 Pr(T < t) = 0.9932         Pr(|T| > |t|) = 0.0135          Pr(T > t) = 0.0068
```

We told Stata to test whether the mean age is equal to 47. So in the above, $H_0 = 47$. The command calculates the mean, standard deviation, standard error, a 95% confidence interval, and a t statistic. Above, our t statistic is 2.4717. Stata also calculates p-values for the three possible tests, a one tailed test for the mean less than the null, a two tailed test for the mean not equal to the null, and a one tailed test for the mean greater than the null. We can choose the appropriate p-value for the test we are doing. Here, we see that even the two tailed test is significant at an α of 0.05.

To perform a test of the difference between two means, we include the `by()` option, with a variable that defines the two groups:

`ttest` *depvar*`, by(`*indvar*`)`

Note that `indvar` must define two and only two groups.

```
. ttest EDUC, by(SEX)

Two-sample t test with equal variances
------------------------------------------------------------------------------
   Group |     Obs        Mean    Std. Err.   Std. Dev.   [95% Conf. Interval]
---------+--------------------------------------------------------------------
    MALE |     889    13.44544    .1092575    3.257632    13.23101    13.65988
  FEMALE |    1150    13.47304    .0903597    3.064247    13.29575    13.65033
---------+--------------------------------------------------------------------
combined |    2039    13.46101     .069743    3.149267    13.32424    13.59779
---------+--------------------------------------------------------------------
    diff |           -.0275992    .1406763               -.3034835    .2482852
------------------------------------------------------------------------------
    diff = mean(MALE) - mean(FEMALE)                              t =  -0.1962
Ho: diff = 0                                     degrees of freedom =     2037

    Ha: diff < 0                 Ha: diff != 0                 Ha: diff > 0
 Pr(T < t) = 0.4222         Pr(|T| > |t|) = 0.8445          Pr(T > t) = 0.5778
```

Notice that Stata automatically calculates the mean for each group, the overall mean, and the mean difference. So here, we are testing whether men and women have different education levels. `EDUC` here is years of education. According to the p-value calculated for the two tailed test, we fail to reject the null that men and women have the same amount of education.

5 Introduction to Regression

Regression analysis is about exploring linear relationships between a dependent variable and one or more independent variables. Regression models can be represented by graphing a line on a cartesian plane. Think back on your high school geometry to get you through this next part.

Suppose we have the following points on a line:

x	y
-1	-5
0	-3
1	-1
2	1
3	3

What is the equation of the line?

$$y = \alpha + \beta x$$

$$\beta = \frac{\Delta y}{\Delta x} = \frac{3-1}{3-2} = 2$$

$$\alpha = y - \beta x = 3 - 2(3) = -3$$

$$y = -3 + 2x$$

If we input the data into STATA, we can generate the coefficients automatically. The command for finding a regression line is `regress`. The STATA output looks like:

```
. regress y x

      Source |       SS       df       MS              Number of obs =       5
-------------+------------------------------           F(  1,     3) =       .
       Model |          40     1          40           Prob > F      =       .
    Residual |           0     3           0           R-squared     =  1.0000
-------------+------------------------------           Adj R-squared =  1.0000
       Total |          40     4          10           Root MSE      =       0

------------------------------------------------------------------------------
           y |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
           x |           2          .        .       .            .           .
       _cons |          -3          .        .       .            .           .
------------------------------------------------------------------------------
```

The first table shows the various sum of squares, degrees of freedom, and such used to calculate the other statistics. In the top table on the right lists some summary statistics of the model including number of observations, R^2 and such. However, the table we will focus most of our attention on is the bottom table. Here we find the coefficients for the variables in the model, as well as standard errors, p-values, and confidence intervals.

In this particular regression model, we find the x coefficient (β) is equal to 2 and the constant (α) is -3. This matches the equation we calculated earlier. Notice that no standard errors are reported. This is because the data fall exactly on the line so there is zero error. Also notice that the R^2 term is exactly equal to 1.0, indicating a perfect fit.

Now, let's work with some data that are not quite so neat. We'll use the `gss2010.dta` data.

```
use hire771

. regress salary age

      Source |       SS       df       MS              Number of obs =    3131
-------------+------------------------------           F(  1,  3129) =  298.30
       Model |  1305182.04     1  1305182.04           Prob > F      =  0.0000
    Residual |  13690681.7  3129  4375.41762           R-squared     =  0.0870
-------------+------------------------------           Adj R-squared =  0.0867
       Total |  14995863.8  3130  4791.01079           Root MSE      =  66.147

------------------------------------------------------------------------------
      salary |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
         age |   2.335512   .1352248    17.27   0.000     2.070374    2.600651
       _cons |   93.82819   3.832623    24.48   0.000     86.31348    101.3429
------------------------------------------------------------------------------
```

The table here is much more interesting. We've regressed age on salary. The coefficient on age is 2.34 and the constant is 93.8 giving us an equation of:

$$salary = 93.8 + 2.34age$$

How do we interpret this? For every year older someone is, they are expected to receive another $2.34 a week. A person with age zero is expected to make $93.8 a week. We can find the salary of someone given their age by just plugging in the numbers into the above equation. So a 25 year old is expected to make:

$$salary = 93.8 + 2.34(25) = 152.3$$

Looking back at the results tables, we find more interesting things. We have standard errors for the coefficient and constant because the data are messy, they do not fall exactly on the line, generating some error. If we look at the R^2 term, 0.087, we find that this line is not a very good fit for the data.