ECO310 - Tutorial 2
Introduction to Stata

Francis Guiton

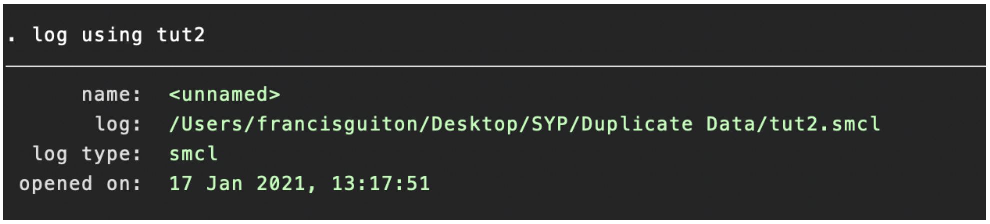
January 18, 2021

The following is an introduction to the basic commands used in Stata. We outline the main commands associated with creating variables, generating summary statistics, generating graphs and running Ordinary Least Squares regressions. If at any point we are unsure about a command, we simply type "help *command name*" in the command window in order to obtain the Stata documentation.

1 Creating a log file

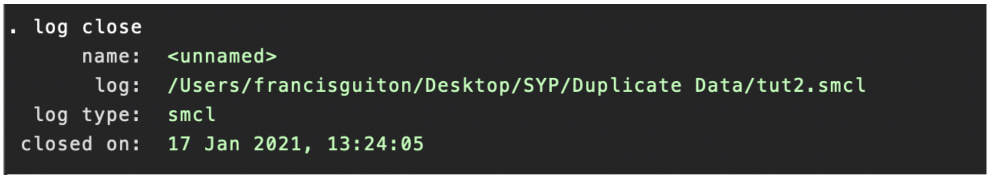
log

Stata can create a copy of everything that is sent to the Results window, with the exception of graphs. This is called a log file and can be helpful for us to save all of our session's output. This will also retain our commands. To create a log file, we simply type ”log using *filename*” in the command window. Stata will start a log, and save the file in our computer's default folder.



```
. log using tut2
--------------------------------------------------------------------------------
      name:  <unnamed>
       log:  /Users/francisguiton/Desktop/SYP/Duplicate Data/tut2.smcl
  log type:  smcl
 opened on:  17 Jan 2021, 13:17:51
```

Once we have finished our Stata session, we simply type ”log close” to close our existing log.

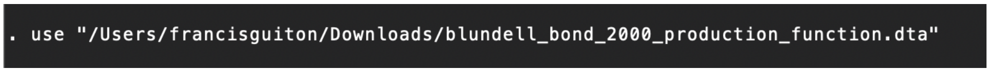


```
. log close
      name:  <unnamed>
       log:  /Users/francisguiton/Desktop/SYP/Duplicate Data/tut2.smcl
  log type:  smcl
 closed on:  17 Jan 2021, 13:24:05
```

2 Loading a dataset

use

In order to load a dataset, we simply type ”use ”*filepathname*”” in the command window.

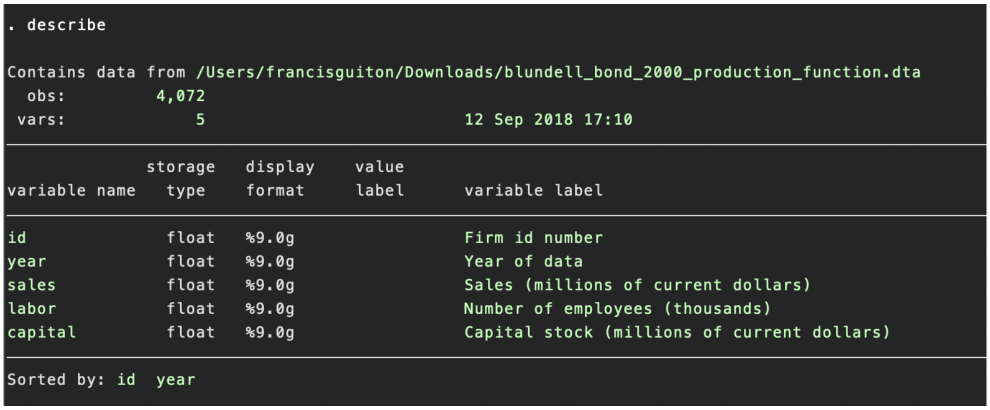


```
. use "/Users/francisguiton/Downloads/blundell_bond_2000_production_function.dta"
```

3 Descriptive statistics

describe

In order to view the dataset currently in memory, we type ”describe” in the command window.

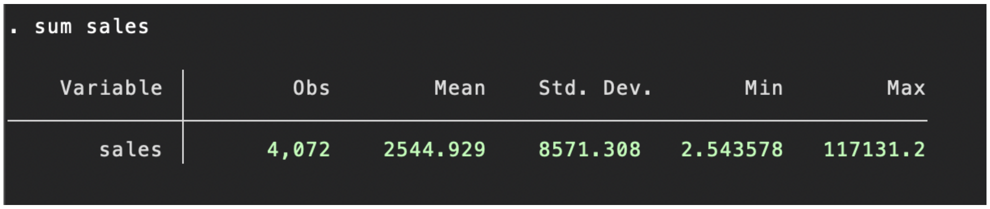


```
. describe

Contains data from /Users/francisguiton/Downloads/blundell_bond_2000_production_function.dta
  obs:         4,072
 vars:             5                          12 Sep 2018 17:10
-------------------------------------------------------------------------------------------
              storage   display    value
variable name   type    format     label      variable label
-------------------------------------------------------------------------------------------
id              float   %9.0g                 Firm id number
year            float   %9.0g                 Year of data
sales           float   %9.0g                 Sales (millions of current dollars)
labor           float   %9.0g                 Number of employees (thousands)
capital         float   %9.0g                 Capital stock (millions of current dollars)
-------------------------------------------------------------------------------------------
Sorted by: id  year
```

sum

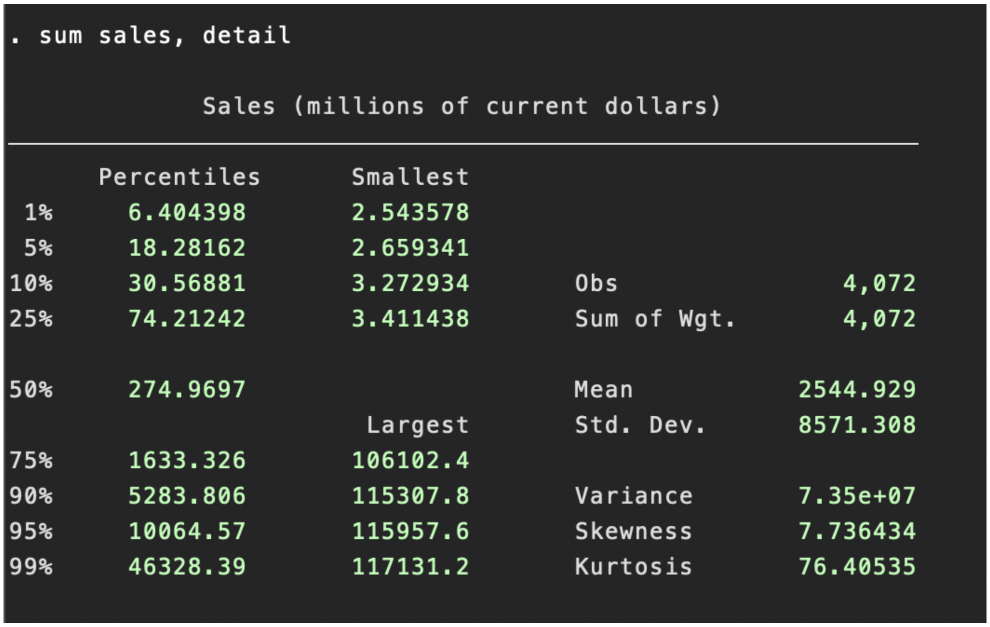
In order to view a variable’s summary statistics, we type ”sum *varname*” in the command window.



```
. sum sales

    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
       sales |      4,072    2544.929    8571.308   2.543578   117131.2
```

We can view additional summary statistics by including ", detail" after the command.

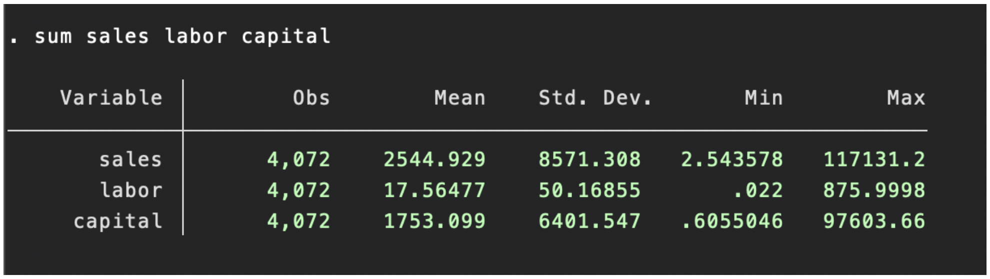


```
. sum sales, detail

             Sales (millions of current dollars)
-------------------------------------------------------------
      Percentiles      Smallest
 1%     6.404398       2.543578
 5%     18.28162       2.659341
10%     30.56881       3.272934       Obs               4,072
25%     74.21242       3.411438       Sum of Wgt.       4,072

50%     274.9697                      Mean           2544.929
                        Largest       Std. Dev.      8571.308
75%     1633.326       106102.4
90%     5283.806       115307.8       Variance       7.35e+07
95%     10064.57       115957.6       Skewness       7.736434
99%     46328.39       117131.2       Kurtosis       76.40535
```

Finally, we can summarize multiple variables at once by typing "sum *varname1 varname2* ...".

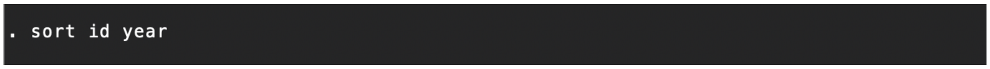


```
. sum sales labor capital

    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
       sales |      4,072    2544.929    8571.308   2.543578   117131.2
       labor |      4,072    17.56477    50.16855       .022   875.9998
     capital |      4,072    1753.099    6401.547   .6055046   97603.66
```

sort

To sort our dataset according to specific variables, we type "sort *varname1 varname2* ...".

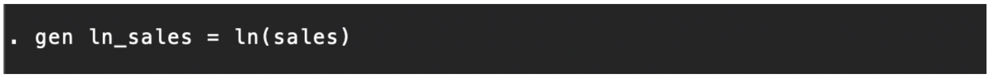


```
. sort id year
```

4 Generating new variables

gen

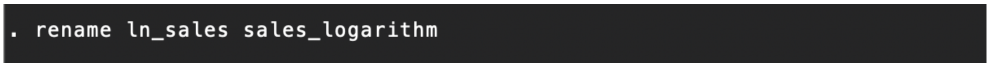
We create a new variable by typing ”gen *varname* = [*syntax*]” in the command window. For example, in order to generate the logarithm of the variable *sales*, we type:



```
. gen ln_sales = ln(sales)
```

rename

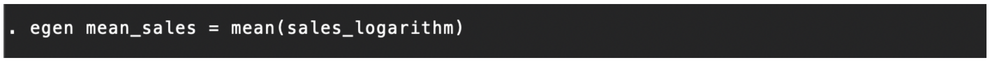
In order to change the name of an existing variable, we simply type ”rename *oldname newname*”.



```
. rename ln_sales sales_logarithm
```

egen

In order to generate a new variable based on descriptive statistics, we type ”egen *varname* = [*syntax*]”. For example, to generate a variable that yields the mean value of the logarithm of sales, we type the following:

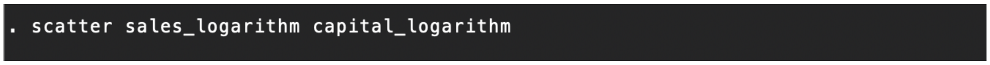


```
. egen mean_sales = mean(sales_logarithm)
```

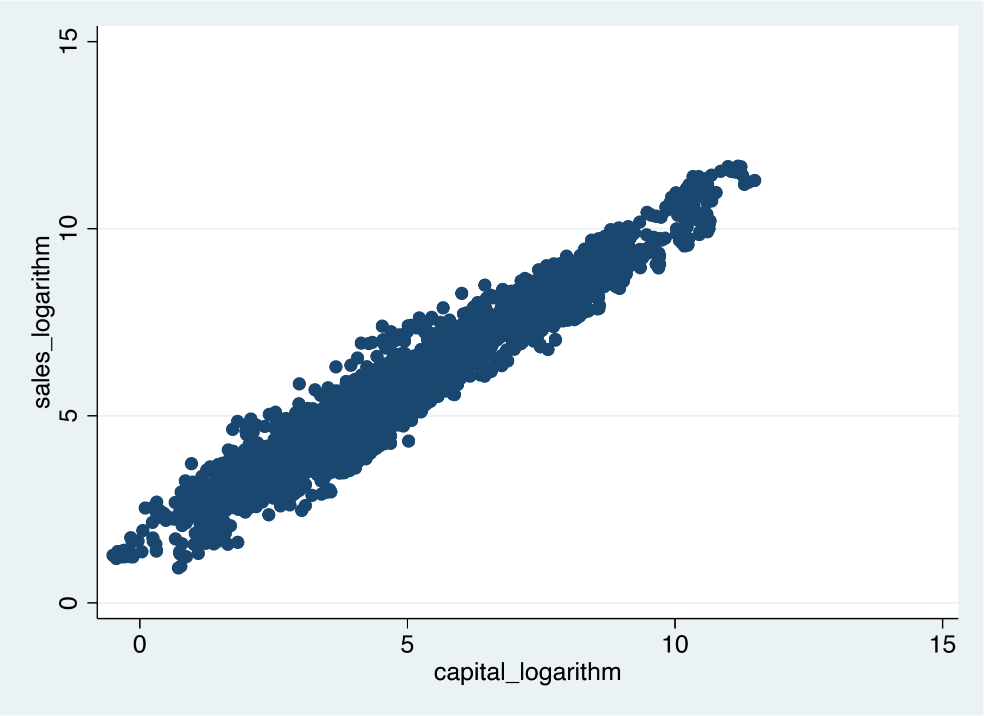
5 Generating graphs

scatter

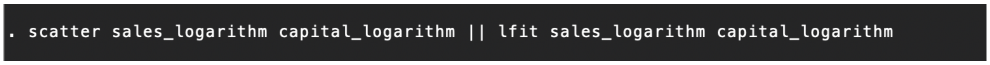
In order to generate a scatter plot of two variables x and y, we type "scatter *varname1 varname2*" in the command window.



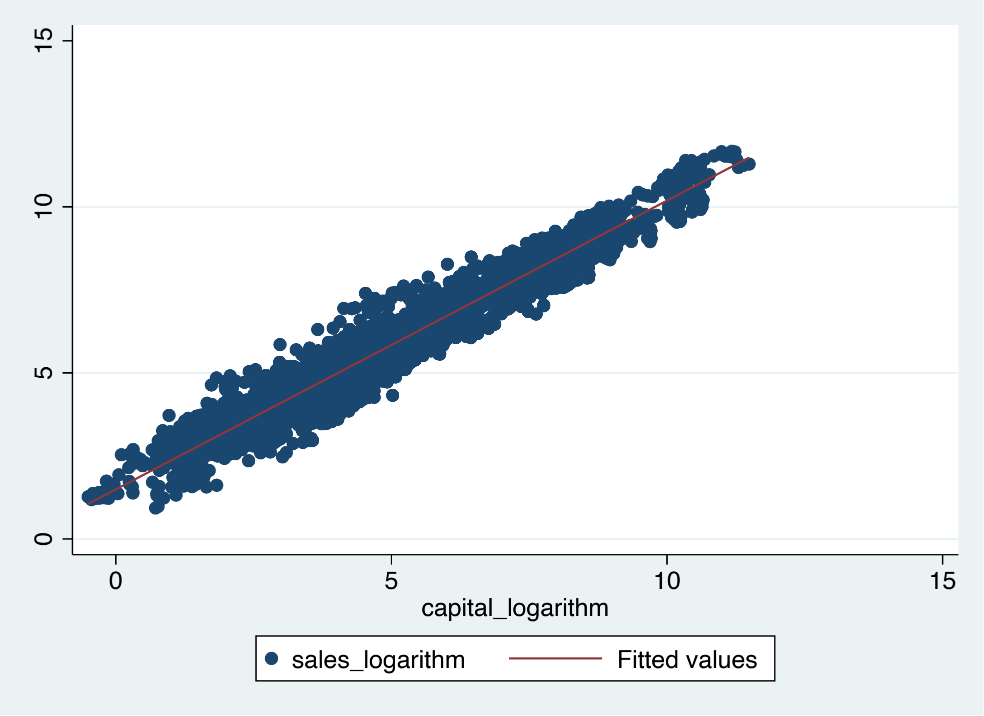
```
. scatter sales_logarithm capital_logarithm
```



To include a regression line in our scatter plot, we simply add ”|| lfit *varname1 varname2*” to our previous command.

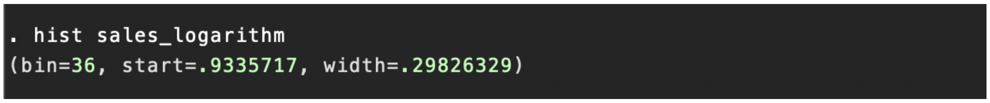


```
. scatter sales_logarithm capital_logarithm || lfit sales_logarithm capital_logarithm
```

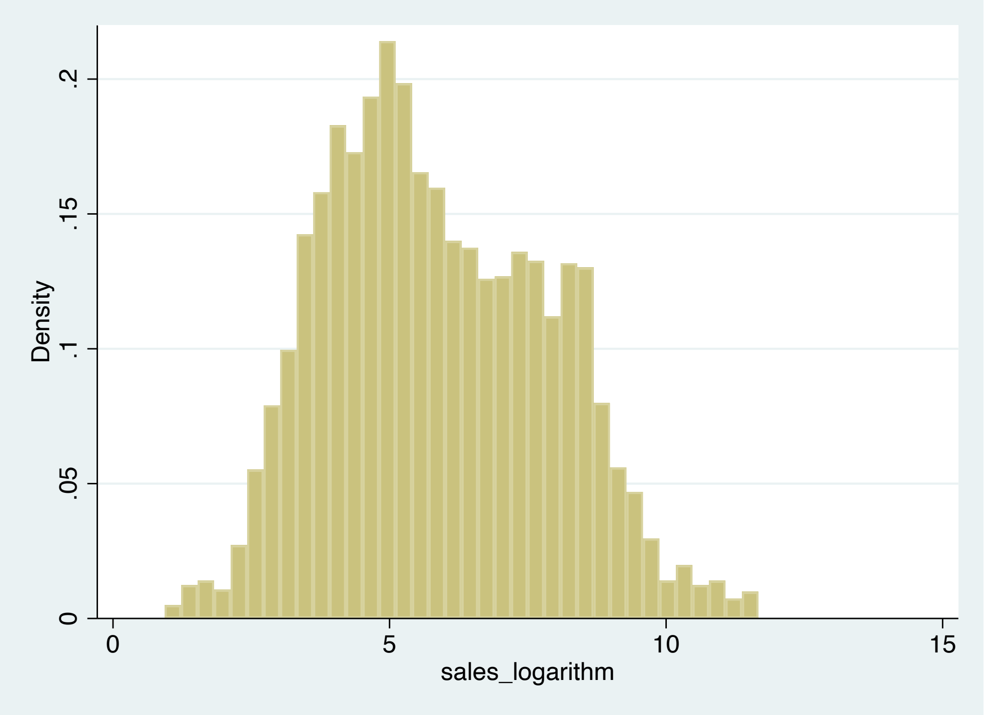


hist

In order to generate a histogram, we type ”hist *varname*” in the command window. Stata provides default bin sizes, but these can be modified by including ”, bin(*# of bins*)” at the end of our previous command.



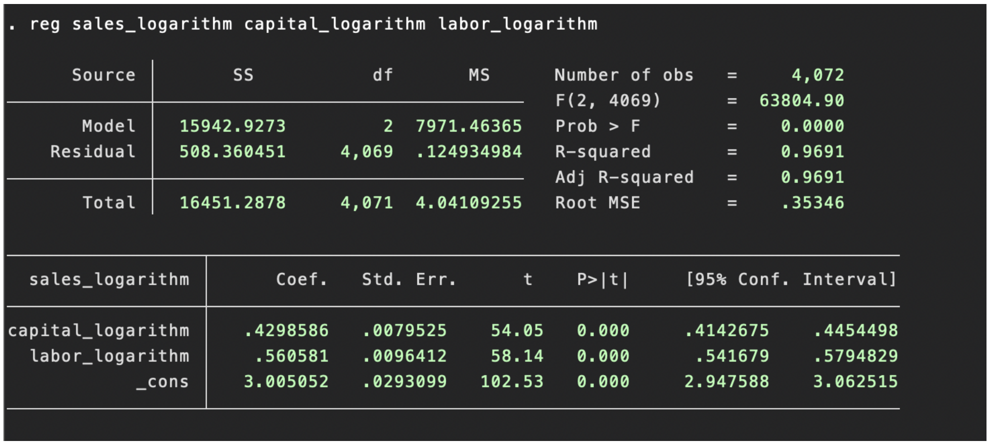
```
. hist sales_logarithm
(bin=36, start=.9335717, width=.29826329)
```



6 Ordinary Least Squares Regression

reg

In order to run an OLS regression of a variable y on variables $x1, x2...$, we type ”reg y *x1 x2*” in the command window.



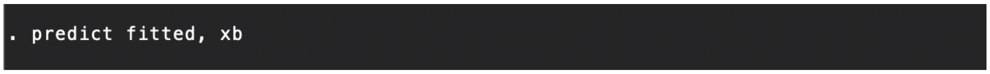
```
. reg sales_logarithm capital_logarithm labor_logarithm

      Source |       SS           df       MS      Number of obs   =     4,072
-------------+----------------------------------   F(2, 4069)      =  63804.90
       Model |  15942.9273         2  7971.46365   Prob > F        =    0.0000
    Residual |  508.360451     4,069  .124934984   R-squared       =    0.9691
-------------+----------------------------------   Adj R-squared   =    0.9691
       Total |  16451.2878     4,071  4.04109255   Root MSE        =    .35346

------------------------------------------------------------------------------
  sales_logarithm |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
------------------+-----------------------------------------------------------
capital_logarithm |   .4298586   .0079525    54.05   0.000     .4142675    .4454498
  labor_logarithm |    .560581   .0096412    58.14   0.000      .541679    .5794829
            _cons |   3.005052   .0293099   102.53   0.000     2.947588    3.062515
------------------------------------------------------------------------------
```

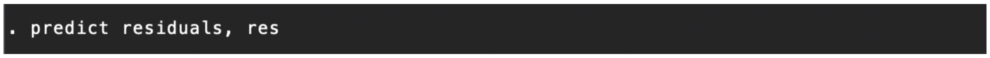
predict

In order to obtain a linear prediction of our dependent variable y, we simply type ”predict *newvar*, xb” in the command window after our regression output:



```
. predict fitted, xb
```

Similarly, in order to obtain the residuals of our regression, we type ”predict *newvar*, res” in the command window after our regression output:



```
. predict residuals, res
```