TUTORIAL 1

Yiran Hao

Sep. 17.2018

Note (1): Today we learned how to run codes in Command Window one by one. Next time we will learn how to create a do.file to create and organize your own codes.

Note (2): If you are not sure how to use a specific command, please type help "the command name" in Command Window. For example, help describe.

Note (3): All following codes are mentioned in tutorial and are in ***Bold Italic style***.

Step(1): create log file

you can have Stata 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 you to save all of your output. This will also retain your commands, although it will not save them in the same way a do-file does (they will be embedded in the output). To create a log file, go to "File" -> "Log" -> "Begin." This will bring up a dialogue box where you will save your log file. The default in Stata is to save the file with the extension .smcl. This will allow you to open the log file in Stata, but other programs will not read this type of file. Since I save the file called "tut1" , the output window shows the following:

```
. log using "F:\tut1.smcl"
-----------------------------------------------
      name:  <unnamed>
       log:  F:\tut1.smcl
  log type:  smcl
 opened on:  17 Sep 2018, 15:20:32
```

The other extension available is .log. This file format will allow you to open your log file in other programs and may be easier to manage than the .smcl files. To save it as a .log file, just select the Stata Log option under the "File Format" menu in the dialogue box.

```
. log using "F:\tut.log"
-----------------------------------------------
      name:  <unnamed>
       log:  F:\tut.log
  log type:  text
 opened on:  17 Sep 2018, 15:35:36
```

Step(2): load dataset

use "C:\Users\admin\Downloads\blundell_bond_2000_production_function.dta"

This directory should correspond to where you saved your dataset. Alternatively, you can choose **File>Open** to open a dataset in Stata format.

Step(3): Summary Statistics

1) The describe command shows you basic information about a Stata data file. As you can see, it tells us the number of observations in the file, the number of variables, the names of the variables, and more:

describe

Alternatively, type the abbreviation:

d

```
Contains data from C:\Users\admin\Downloads\blundell_bond_2000_production_function.dta
  obs:         4,072
 vars:             5                          12 Sep 2018 17:10
 size:        81,440
------------------------------------------------------------------------------------------------
              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
```

2) The list command is useful for viewing all or a range of observations. To list variable id, type following:

list id

```
. list id

     +------+
     |   id |
     |------|
  1. |  886 |
  2. |  886 |
  3. |  886 |
  4. |  886 |
  5. |  886 |
     |------|
  6. |  886 |
  7. |  886 |
  8. |  886 |
  9. | 1030 |
 10. | 1030 |
     |------|
 11. | 1030 |
 12. | 1030 |
 13. | 1030 |
 14. | 1030 |
 15. | 1030 |
     |------|
 16. | 1030 |
 17. | 1723 |
 18. | 1723 |
 19. | 1723 |
 20. | 1723 |
     |------|
 21. | 1723 |
 22. | 1723 |
```

To list out 1st observation:

list in 1

```
. list in 1

     +-------------------------------------------+
     |  id   year      sales   labor    capital  |
     |-------------------------------------------|
  1. | 886   1982   97.43913   1.771   35.78286  |
     +-------------------------------------------+
```

To list out first 10 observations:

list in 1/10

```
. list in 1/10
```

	id	year	sales	labor	capital
1.	886	1982	97.43913	1.771	35.78286
2.	886	1983	87.54815	1.898	36.99793
3.	886	1984	96.2583	1.554	40.23543
4.	886	1985	132.2913	1.729	44.54794
5.	886	1986	148.2734	1.729	53.32573
6.	886	1987	166.863	1.796	59.247
7.	886	1988	172.9347	1.68	68.26998
8.	886	1989	181.1696	1.896	79.65293
9.	1030	1982	64.21842	1.235	22.47299
10.	1030	1983	73.63306	1.318	25.20519

3) Use command: sort, which arranges the observations of the current data into ascending order based on the values of the variables in varlist.

sort id

when you take a look of Data Browser, id is ordered as following:

	id
1	886
2	886
3	886
4	886
5	886
6	886
7	886
8	886
9	1030
10	1030
11	1030
12	1030
13	1030
14	1030
15	1030
16	1030

sort year

year
1982
1982
1982
1982
1982
1982
1982
1983
1983
1983
1983
1983
1983
1983
1983
1983

To sort id first then sort year:

sort id year

	id	year
1	886	1982
2	886	1983
3	886	1984
4	886	1985
5	886	1986
6	886	1987
7	886	1988
8	886	1989
9	1030	1982
10	1030	1983
11	1030	1984
12	1030	1985
13	1030	1986
14	1030	1987
15	1030	1988
16	1030	1989

To sort year first then sort id in each group of year:

sort year id

	id	year
1	886	1982
2	1030	1982
3	1723	1982
4	1909	1982
5	2824	1982
6	4626	1982
7	4644	1982
8	4816	1982
9	5313	1982
10	7903	1982
11	9158	1982
12	12041	1982

4) command: summarize--to get summary statistics: mean , min, max, etc.

summarize id

or alternatively:

sum id

Variable	Obs	Mean	Std. Dev.	Min	Max
id	4,072	483123.5	297055.5	886	989349

The output shows following:

Variable – This column indicates which variable is being described. You can list more than one variable after the summarize command; when you do, you will see each variable on its own line of the output.

Obs – This column tells you the number of observations (or cases) that were valid (i.e., not missing) for that variable.

Mean – This is the mean of the variable.

Std. Dev. – This is the standard deviation of the variable. This gives information regarding the spread of the distribution of the variable.

If add the option detail to summarize, this will give us lots more information, including the median and other percentiles:

sum id,detail

```
. sum id,detail

                       Firm id number
-------------------------------------------------------------
      Percentiles      Smallest
 1%         4626            886
 5%        29429            886
10%        53492            886       Obs               4,072
25%       237688            886       Sum of Wgt.       4,072

50%       460146                      Mean           483123.5
                        Largest       Std. Dev.      297055.5
75%       751277         989349
90%       890278         989349       Variance       8.82e+10
95%       922204         989349       Skewness      -.0096851
99%       974637         989349       Kurtosis       1.745787
```

The output shows following:

1% – This is the first percentile. Percentiles are calculated by ordering the values of a variable from lowest to highest, and then finding the value that corresponds to whatever percent you are interested in, in this case, 1%. Hence, 1% of the values of the variable write are equal to or less than 4626.

25% – This is the 25th percentile, also known as the first quartile.

50% – This is the 50th percentile, also known as the median. If you order the values of the variable from lowest to highest, the median would be the value exactly in the middle. In other words, half of the values would be below the median, and half would be above. This is a good measure of central tendency if the variable has outliers.

75% – This is the 75th percentile, also known as the third quartile.

Smallest – This is a list of the four smallest values of the variable. In this example, the four smallest values are all 886.

Largest – This is a list of the four largest values of the variable. In this example, the four largest values are all 989349.

Obs – This column tells you the number of observations (or cases) that were valid (i.e., not missing) for that variable.

Sum of Wgt. – This is the sum of the weights. In Stata, you can use different kinds of weights on your data. By default, each case (i.e., subject) is given a weight of 1. When this default is used, the sum of the weights will equal the number of observations.

Mean – This is the arithmetic mean across the observations. It is the most widely used measure of central tendency. It is commonly called the average. The mean is sensitive to extremely large or small values.

Std. Dev. – This is the standard deviation of the variable. This gives information regarding the spread of the distribution of the variable.

Variance – This is the standard deviation squared (i.e., raised to the second power). It is also a measure of spread of the distribution.

Skewness – Skewness measures the degree and direction of asymmetry. A symmetric distribution such as a normal distribution has a skewness of 0, and a distribution that is skewed to the left, e.g., when the mean is less than the median, has a negative skewness.

Kurtosis – Kurtosis is a measure of the heaviness of the tails of a distribution. A normal distribution has a kurtosis of 3. Heavy tailed distributions will have kurtosis greater than 3 and light tailed distributions will have kurtosis less than 3.

Multiple Variables at Once: To get descriptives for multiple variables at once just add the variable names after summarize:

sum id sales labor capital

```
. sum id sales labor capital
```

Variable	Obs	Mean	Std. Dev.	Min	Max
id	4,072	483123.5	297055.5	886	989349
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

Step(4): generate new variables

You create a new variable in Stata using the generate command, usually abbreviated gen. The something you're setting the variable to will be the result of some math, but it can be really simple math, like a single number. Here we want to transform cobb-douglas function to a logarithm form. To get log form of Y, K ,L:

gen ln_sales=log(sales)
gen ln_labor=log(labor)
gen ln_capital=log(capital)

Then the 3 new variables : ln_sales ln_labor ln_capital are created in variables list:

Variables

Filter variables here

Name	Label
id	Firm id number
year	Year of data
sales	Sales (millions of ...
labor	Number of emplo...
capital	Capital stock (mill...
ln_sales	
ln_labor	
ln_capital	

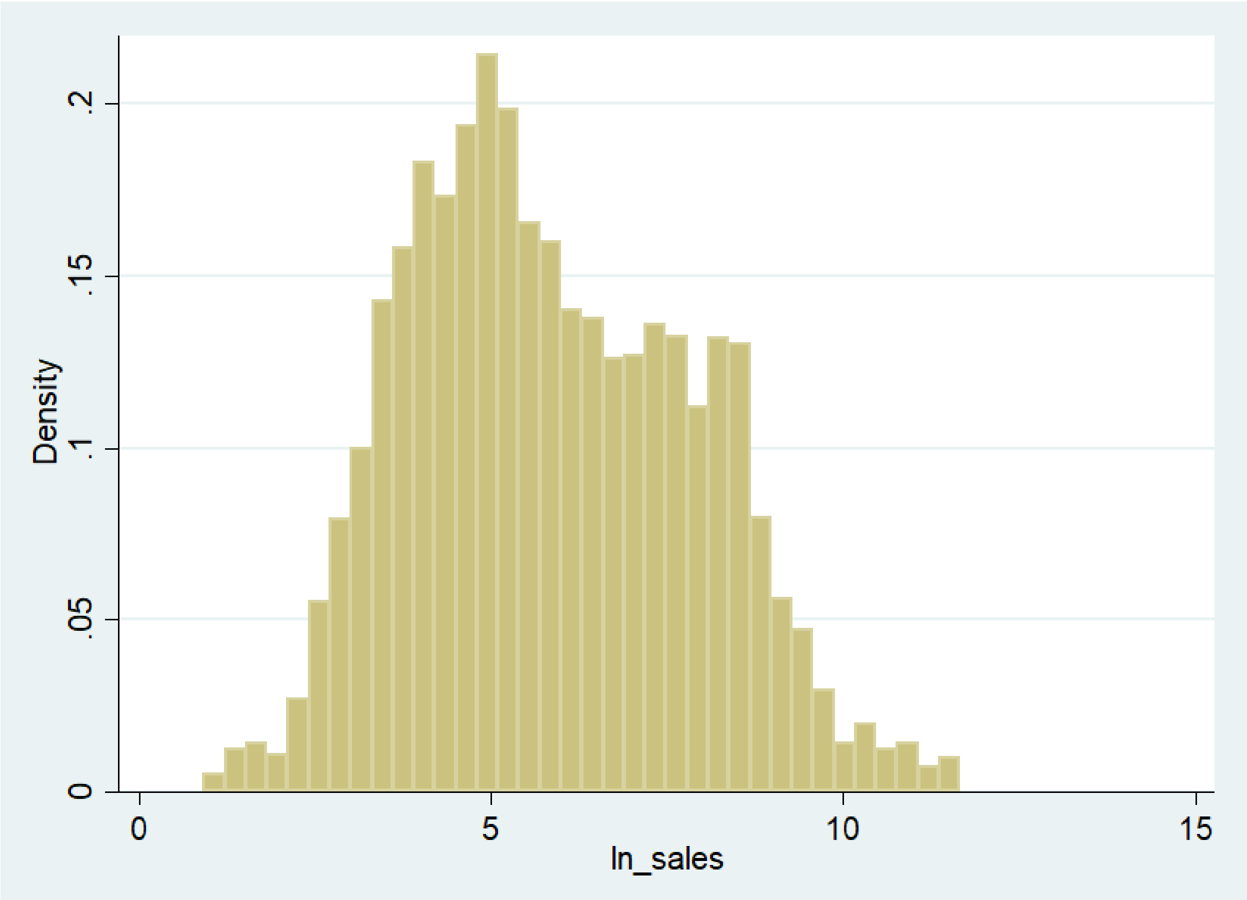
You can check the values of new variables in DATA Browser as follows:

	id	year	sales	labor	capital	ln_sales	ln_labor	ln_capital
1	886	1982	97.43913	1.771	35.78286	4.579228	.5715443	3.577469
2	1030	1982	64.21842	1.235	22.47299	4.16229	.211071	3.112314
3	1723	1982	732.1416	11.69899	166.9971	6.595974	2.459503	5.117976
4	1909	1982	34.2494	.47	8.770642	3.533669	-.7550226	2.17141
5	2824	1982	3105.544	32.145	1330.276	8.040944	3.470257	7.193142
6	4626	1982	390.1764	3.621999	155.3813	5.966599	1.287026	5.045882
7	4644	1982	73.969	1.142	16.74413	4.303646	.1327811	2.818048
8	4816	1982	47.51787	.8	9.571863	3.861106	-.2231435	2.258828
9	5313	1982	63.0072	.665	10.19674	4.143249	-.4079682	2.322068
10	7903	1982	427.6191	10.521	282.8981	6.058233	2.353373	5.645087

Step(5): Draw graphs

The command to create a histogram is just histogram, which can be abbreviated hist. It is followed by the name of the variable you want it to act on:

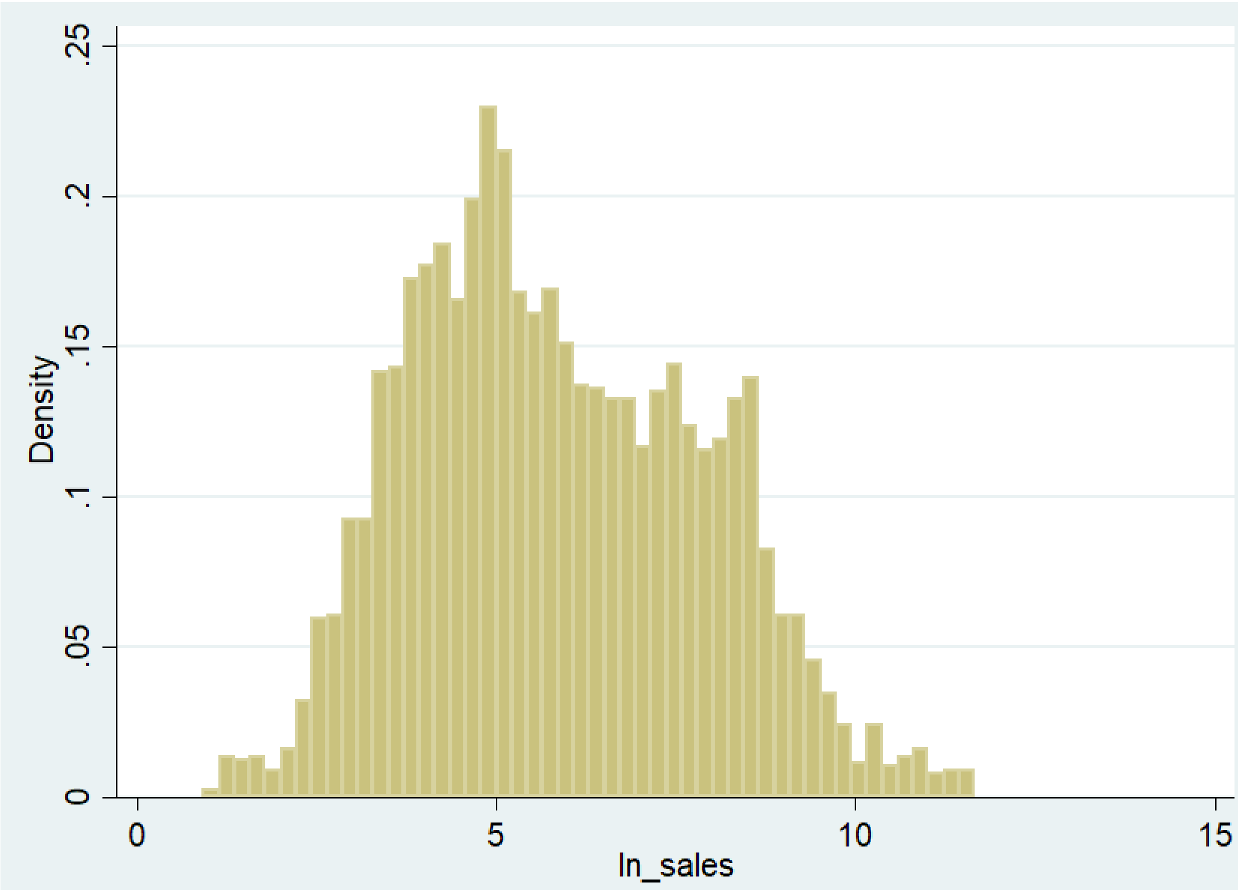
hist ln_sales



The y-axis is labeled as Density because Stata likes to think of a histogram as an approximation to a probability density function.

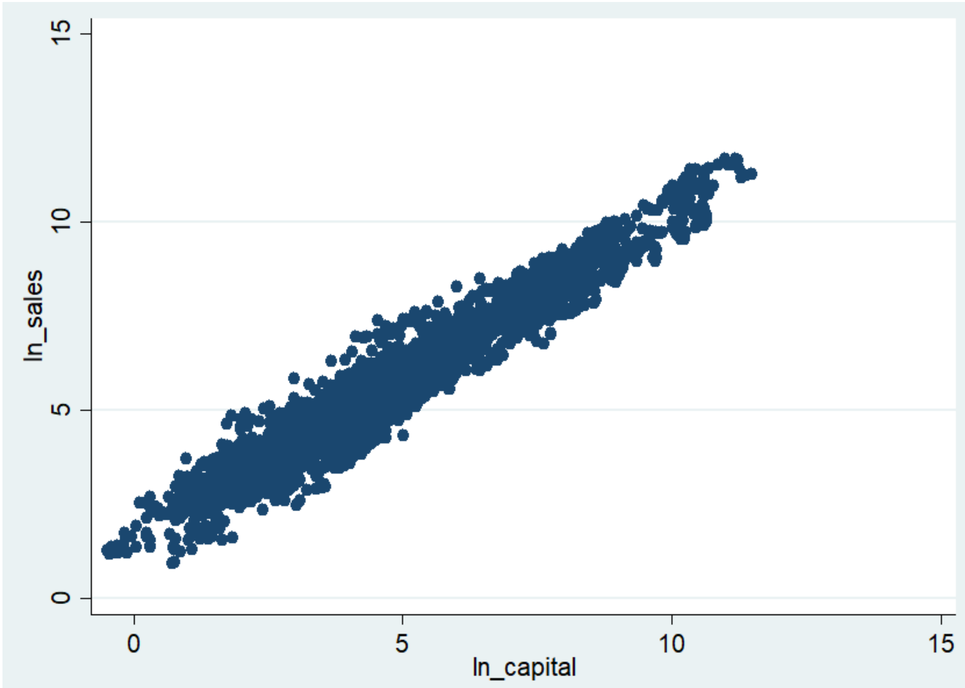
You can control how many "bins" the data are divided into with the bin() option, putting the desired number of bins in the parentheses. Compare the above with 50 bins:

hist ln_sales, bin(50)



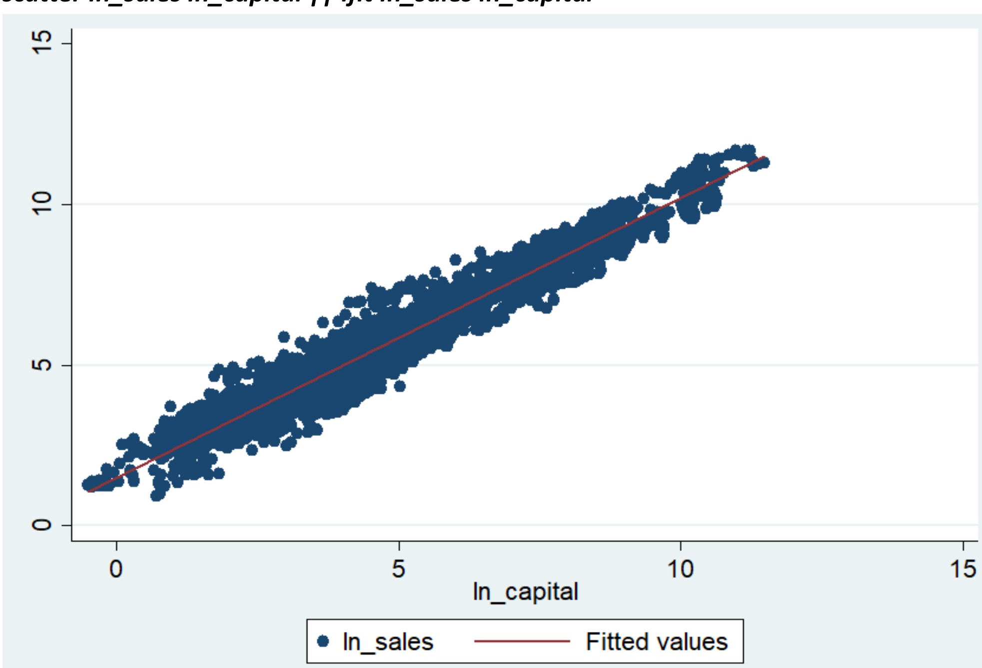
A scatterplot is an excellent tool for examining the relationship between two quantitative variables. One variable is designated as the Y variable and one as the X variable, and a point is placed on the graph for each observation at the location corresponding to its values of those variables. If you believe there is a causal relationship between the two variables, convention suggests you make the cause X and the effect Y, but a scatterplot is useful even if there is no such relationship. To create a scatterplot, use the scatter command, then list the variables you want to plot. The first variable you list will be the Y variable and the second will be the X variable:

scatter ln_sales ln_capital



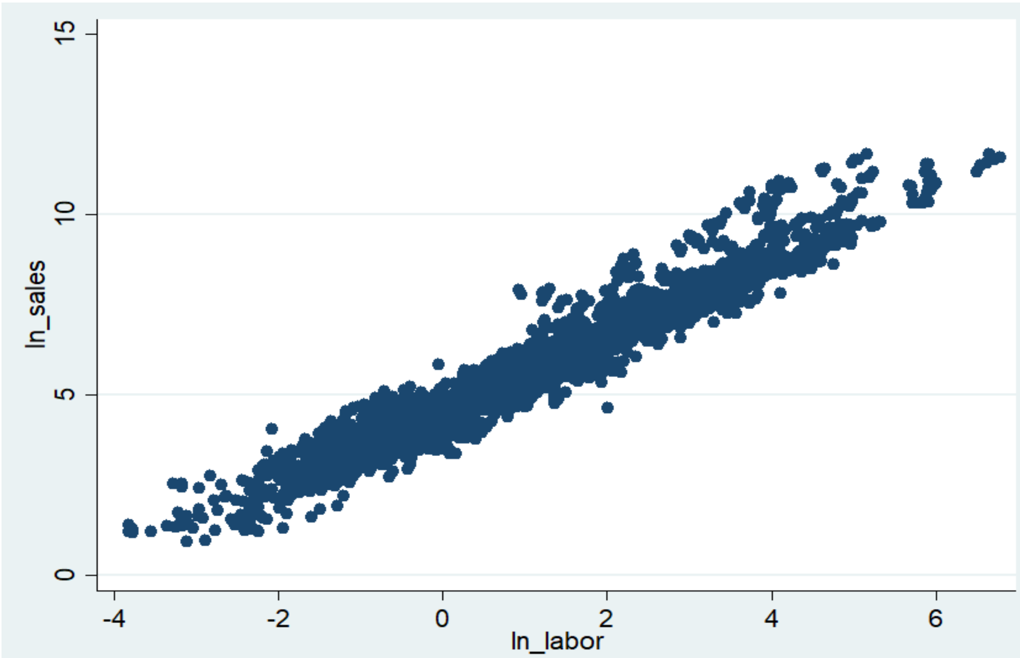
To add a fitted line: Regression attempts to find the line that best fits these points. You can plot a regression line or "linear fit" with the lfit command followed, as with scatter, by the variables involved. To add a linear fit plot to a scatterplot, first specify the scatterplot, then put two "pipe" characters (what you get when you press shift-Backslash) to tell Stata you're now going to add another plot, and then specify the linear fit.

scatter ln_sales ln_capital || lfit ln_sales ln_capital

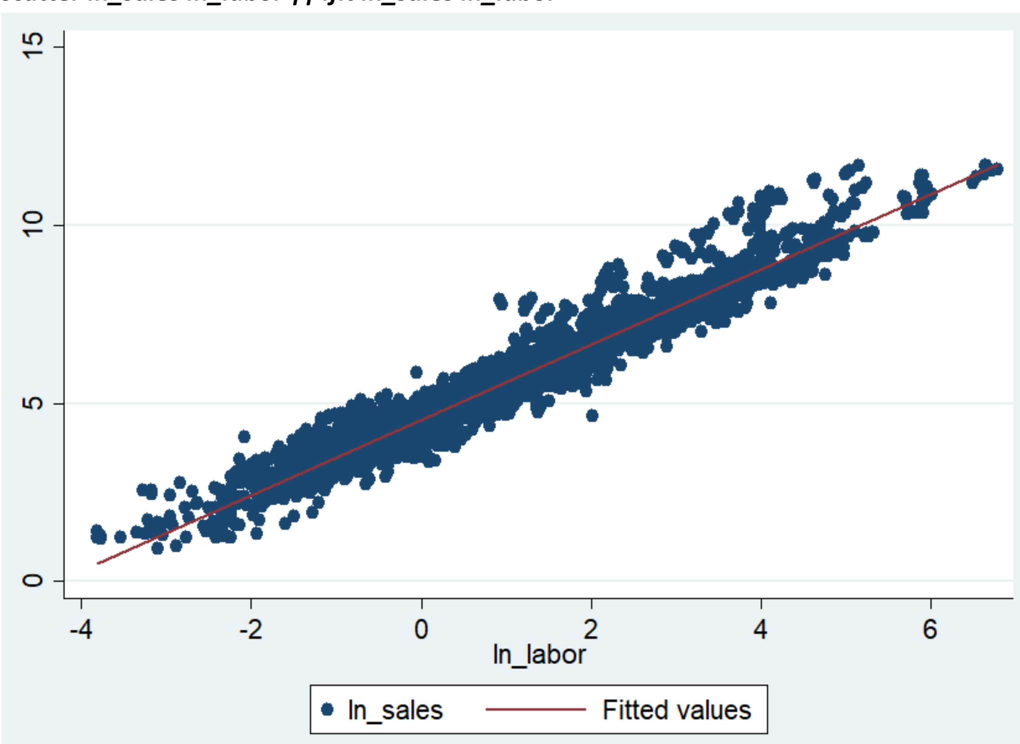


Similarly for labor:

scatter ln_sales ln_labor



scatter ln_sales ln_labor || lfit ln_sales ln_labor



Step(6): run OLS regression (production function estimation)

linear regression estimates how much Y changes when X changes one unit. In Stata we use command: regress, type dependent variable first then followed by explanatory variables:

reg ln_sales ln_capital ln_labor

```
. reg ln_sales ln_capital ln_labor

      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

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

The regress command reports many statistics. In particular,

- The number of observations is at the top of the small table on the right
- The sum of squared residuals is in the first column of the table on the left (under SS), in the row marked "Residual".
- The least-squares estimate of the error variance is in the same table, under "MS" and in the row "Residual". The estimate of the error standard deviation is its square root, and is in the right table, reported as "Root MSE".
- The coefficient estimates are repoted in the bottom table, under "Coef".
- Standard errors for the coefficients are to the right of the estimates, under "Std. Err."

Step(7): postestimation

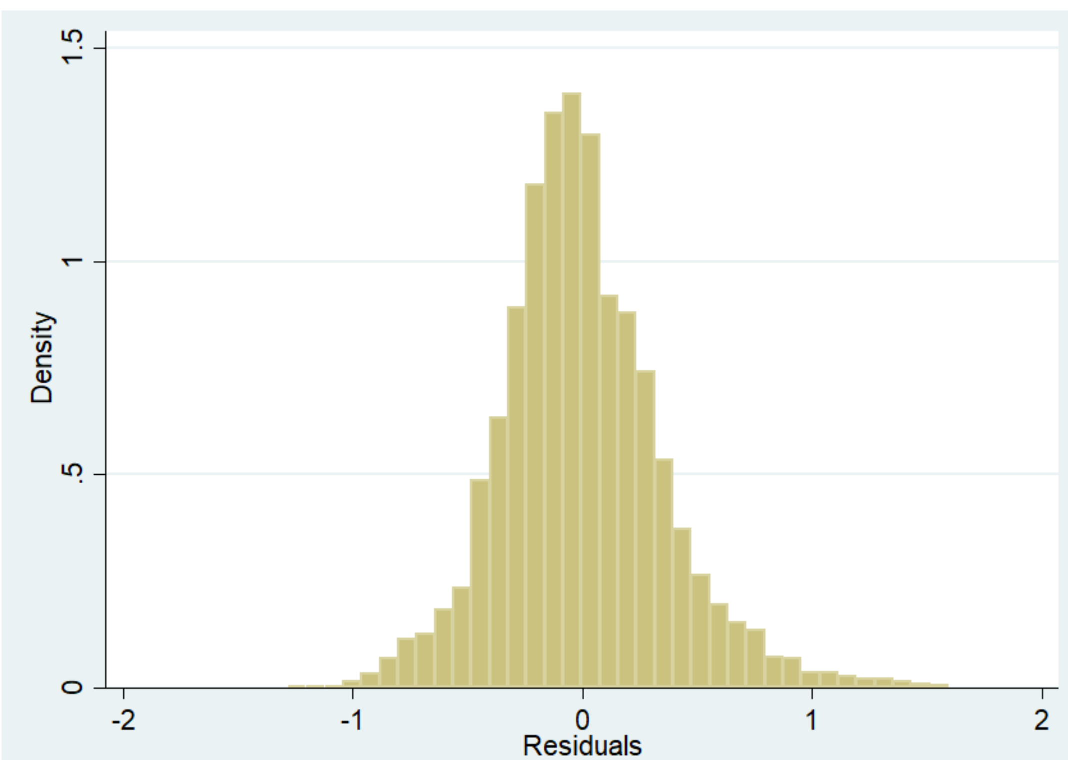
Since TFP (total factor productivity) can be estimated as the residual of the regression above (please refer to lecture notes), we use command: predict to get residual from the regression. The following code creates a variable "ln_tfp" of the in-sample residuals y-x'beta.

predict ln_tfp,residuals

ln_tfp
-.2840267
-.2989391
.0121655
.0184691
-.0015018
.0710489
.0127975
.0101675
.3687355
-.6926644
-.6894046
-.6294537
-.3781957

To see the distribution of TFP:

hist ln_tfp



To see different percentiles of TFP:

sum ln_tfp, detail

```
. sum ln_tfp, detail

                           Residuals
-------------------------------------------------------------
      Percentiles      Smallest
 1%    -.8008512      -1.279936
 5%    -.5239248      -1.180148
10%    -.4060703      -1.069904       Obs               4,072
25%    -.2174065      -.9967354       Sum of Wgt.       4,072

50%    -.0298519                      Mean          -7.30e-12
                        Largest       Std. Dev.      .3533746
75%     .1962789       1.458571
90%     .4354635       1.464603       Variance       .1248736
95%     .6227146       1.549903       Skewness       .5475013
99%     1.083821       1.594861       Kurtosis       4.348316
```

Finally, to get a revolution of TFP for percentile 10th 50th and 90th over time:

egen: "egen" is used to create (generate) variables from information across multiple rows of data. Examples of variables that can be defined using "egen" are means, percentiles, min values, max values, and groups.
By: runs a command seperately for each value of a variable. by requires that the data is sorted by the variable in question and cannot be abbreviated.
e10 means the 10th percentile of TFP for a given year; e10 means the median of TFP for a given year; e90 means the 90th percentile of TFP for a given year.

sort year
by year: egen e10 = pctile(ln_tfp), p(10)
by year: egen e50 = pctile(ln_tfp), p(50)
by year: egen e90 = pctile(ln_tfp), p(90)

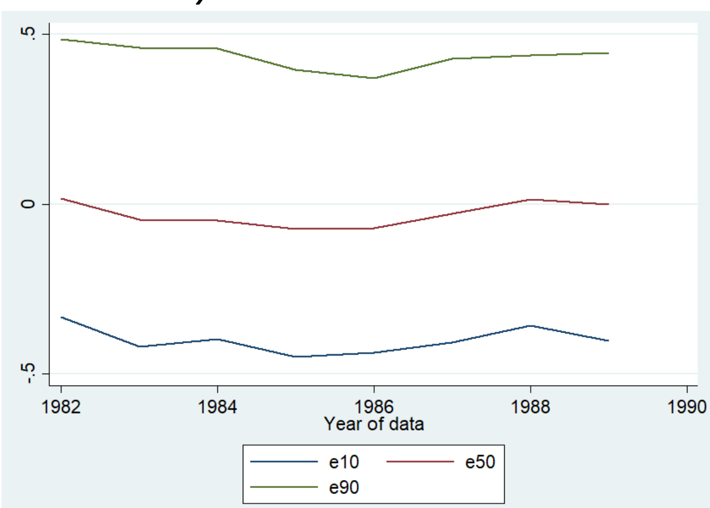
e10	e50	e90
-.3328474	.0158628	.4866805
-.3328474	.0158628	.4866805
-.3328474	.0158628	.4866805
-.3328474	.0158628	.4866805
-.3328474	.0158628	.4866805
-.3328474	.0158628	.4866805
-.3328474	.0158628	.4866805
-.3328474	.0158628	.4866805
-.3328474	.0158628	.4866805

or alternatively use median function to get 50th percentile of TFP

by year: egen e50 = median(ln_tfp)

Draw a time trend graph of 10th 50th 90th percentile's TFP, remember to type time variable as the last variable:

line e10 e50 e90 year



Step(8): close the log file

log close

```
. log close
      name:  <unnamed>
       log:  F:\tut1.log
  log type:  text
 closed on:  17 Sep 2018, 17:59:15
```

Or choose **File>Log>Close**