

Soci708-001 Statistics for Sociologists

Fall 2009

Professor François Nielsen

Stata Commands for Module 10 – Inference for Regression

For further information on any command in this handout, simply type `help` followed by the name of the command in Stata.

See also the Stata and SAS Guide pdf (click on Documents in side bar; guide is linked under Software Documentation).

1 Statistical Functions in Stata

The following statistical functions in Stata are useful for regression work. The regression printout itself usually comprises all necessary statistics.

1.1 Normal Distribution Functions

The function `normal(z)` returns $P(Z \leq z)$, the area under the standard normal curve to the left of z . (Compare with Table A.)

```
. display normal(1.207)  
.88628393
```

The function `invnormal(p)` returns z such that $P(Z \leq z) = p$, i.e. such that the area under the standard normal curve to the left of z is p . (Compare with Table A and Table D (bottom row).)

```
. display invnormal(0.975)  
1.959964
```

1.2 Student t Distribution Functions

The function `ttail(df, t)` returns $P(T > t)$, the area under the Student's t distribution with df degrees of freedom to the right of t . (Compare with Table D.).

```
. display ttail(7, 1.960)  
.04540985
```

The function `invttail(df, p)` returns t such that $P(T > t) = p$, i.e. t such that the area under Student's t distribution with df degrees of freedom to the right of t is p . (Compare with Table D.)

```
. display invttail(7, 0.025)  
2.3646243
```

1.3 F Distribution Functions

The function `Ftail(n1, n2, f)` returns $P(F > f)$, the area under the F distribution with `n1` and `n2` degrees of freedom to the right of `f`. (Compare with Table E.)

```
. display Ftail(1, 14, 21.55)
.00038068
```

The function `invFtail(n1, n2, p)` returns `f` such that $P(F > f) = p$, i.e. `f` such that the area under the `F` distribution with `n1` and `n2` degrees of freedom to the right of `f` is `p`. (Compare with Table D.)

```
. display invFtail(1, 14, .00038068)
21.549979
```

2 Simple Regression With Direct Data Input

Simple example where `x` is years of education and `y` is yearly income in dollars.

```
. input x
```

```

          x
1. 12
2. 13
3. 12
4. 14
5. 12
6. 15
7. 12
8. end
```

```
. input y
```

```

          y
1. 20000
2. 22000
3. 23000
4. 25000
5. 18000
6. 30000
7. 26000
8. end
```

```
. reg y x
```

Source	SS	df	MS			
Model	51843318	1	51843318	Number of obs =	7	
Residual	43870967.7	5	8774193.55	F(1, 5) =	5.91	
Total	95714285.7	6	15952381	Prob > F =	0.0593	
				R-squared =	0.5416	
				Adj R-squared =	0.4500	
				Root MSE =	2962.1	

	y	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]
	x	2419.355	995.3064	2.43	0.059	-139.1616 4977.871

```
_cons | -7677.419 12845.68 -0.60 0.576 -40698.29 25343.45
```

3 Simple Regression With Data Input from a Spreadsheet

Another way to enter data for a simple regression in Stata is to enter data in a spreadsheet or use an existing spreadsheet. I will use as an example data data on nonexercise activity (NEA) and fat gain from IPS6e Example 2.12 p.109. Researchers deliberately overfed 16 healthy young adults for 8 weeks. They measured fat gain (in kg) and increase in energy use (in calories) from activity other than deliberate exercise – fidgeting, daily living, and the like. Open the spreadsheet eg02_012.xls from the textbook disk and copy (Ctrl-C) the data to the clipboard:

```
nea      fat
-94      4.2
-57      3
-29      3.7
135      2.7
143      3.2
151      3.6
245      2.4
355      1.3
392      3.8
473      1.7
486      1.6
535      2.2
571      1
580      0.4
620      2.3
690      1.1
```

Then in Stata paste (Ctrl-V) the table into the Data Editor and *close* the Data Editor (click on ×). You may have to clear data currently in memory. Then do the regression. Note that in the regression command the response variable (fat in the example) is listed first.

```
. reg fat nea
```

Source	SS	df	MS	Number of obs =	16
Model	11.7941477	1	11.7941477	F(1, 14) =	21.55
Residual	7.66335139	14	.547382242	Prob > F =	0.0004
Total	19.4574991	15	1.29716661	R-squared =	0.6061
				Adj R-squared =	0.5780
				Root MSE =	.73985

fat	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
nea	-.0034415	.0007414	-4.64	0.000	-.0050317	-.0018513
_cons	3.505123	.3036164	11.54	0.000	2.85393	4.156315

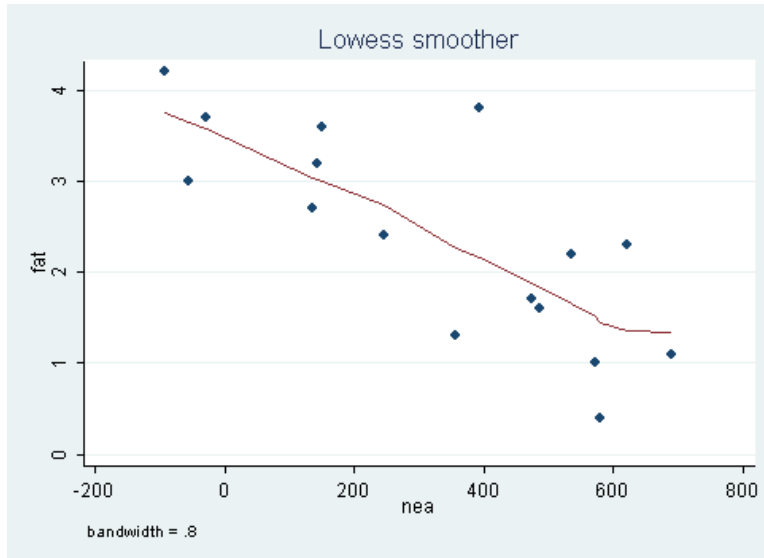


Figure 1: Plot of fat gain by nonexercise activity with lowess curve.

4 Graphs and Residual Analysis for Simple Regression

We continue with the previous example to show some useful commands.

To check linearity of the regression we may want to do a lowess plot of the response variable *fat* against the explanatory variable *nea*. (Lowess is a type of nonparametric regression used to show the main trend in the data.)

```
. lowess fat nea
```

The lowess plot is shown in Figure 1. The linear appearance of the lowess curve gives us confidence that the relationship between *fat* and *nea* is linear. To make a scatterplot with the regression line, use the following command. When the graph appear you can save it in a variety of formats from the graph window. This produces the graph shown in Figure 2.

```
. twoway (scatter fat nea) (lfit fat nea)
```

Now make a similar plot using the residuals of the regression of *fat* on *nea* on the y-axis. First we repeat the regression command (which is not necessary if you have already run it). Then we get to the residuals by creating a variable *fatresid* (my invented name) and telling Stata that this variable corresponds to the residuals of the regression we just ran, with the command `predict fatresid, residuals`. Then we create another variable called *fatpredict* (which will be essentially equivalent to the explanatory variable *nea*), using the command `predict fatpredict, xb`. Finally we do a plot of *fatresid* (the residual e) against *fatpredict* (the predictor \hat{y}). The plot is shown in Figure 3.

```
. reg fat nea
```

```
. predict fatresid, residuals
```

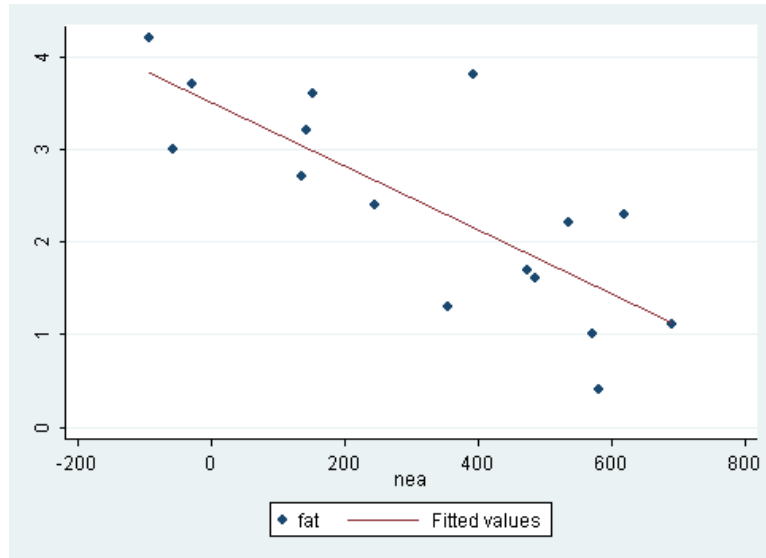


Figure 2: Scatterplot of fat gain by nonexercise activity with fitted regression line.

```
. predict fatpredict, xb
. twoway (scatter fatresid fatpredict), yline(0)
```

We further analyze the distribution of residuals by creating a histogram (with superposed kernel density) (Figure 4) and a normal quantile plot (Figure 5). We note that except for one rather large positive residual the residuals in this case appear fairly normal.

```
. histogram fatresid, kdensity
(bin=4, start=-1.1090604, width=.68825009)
. qnorm fatresid
```

Finally another useful plot is the linear prediction plot with confidence limits for the mean response. This is obtained with the following command. Make sure you place the `lfitci fat nea` part first, and the `scatter fat nea` part second, as otherwise the confidence band produced by `lfitci` will hide the data points. The plot is shown in Figure 6.

```
. twoway (lfitci fat nea) (scatter fat nea)
```

Finally the last command shows the prediction plot with both the 95% confidence limits for the mean response (`lfitci` with default `stdp` option; compare with IPS6e Figure 10.9 p.573) and the the 95% confidence limits for individual responses (`lfitci, stdf`; compare with IPS6e Figure 10.10 p.575). The latter confidence includes uncertainty due to prediction of the mean *and* uncertainty due to prediction of individual responses and is useful to detect outliers. The plot is shown in Figure 7.

```
. twoway (lfitci fat nea) (lfitci fat nea, stdf ciplot(rline)) (scatter fat nea)
```

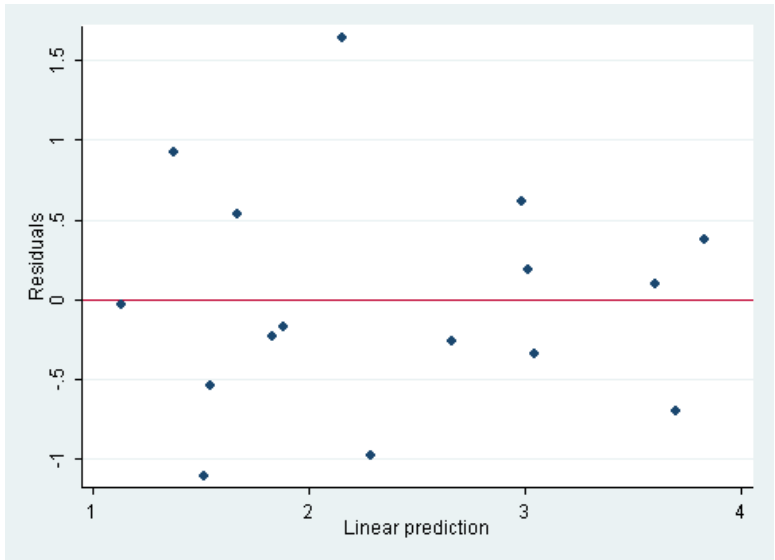


Figure 3: Scatterplot of fat gain residuals e by predicted value \hat{y} as function of nonexercise activity.

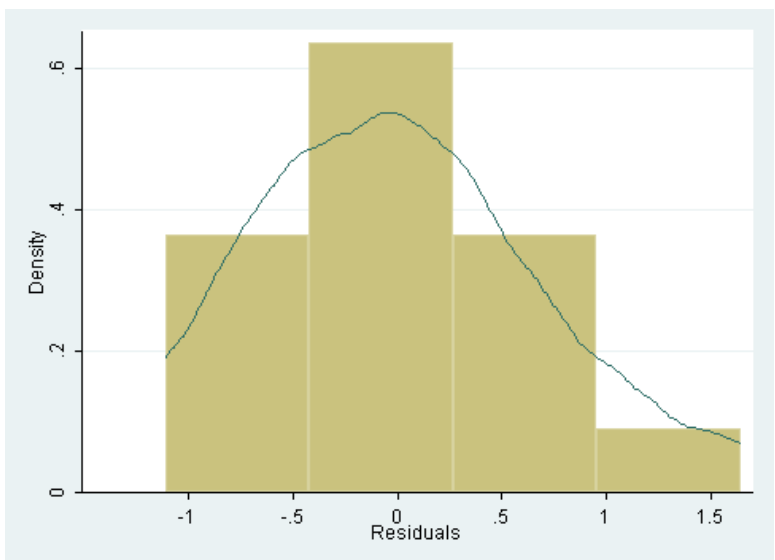


Figure 4: Histogram of fat gain residuals fatresid with superposed kernel density.

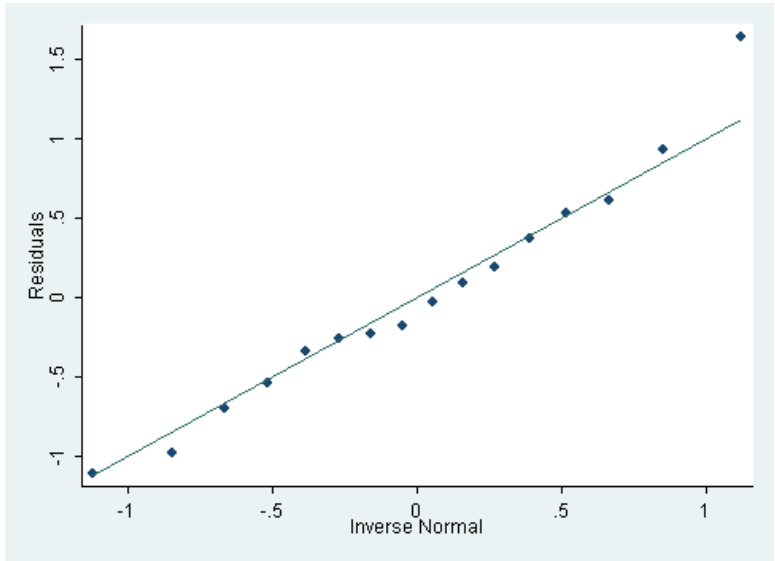


Figure 5: Normal quantile plot of fat gain residuals fatresid.

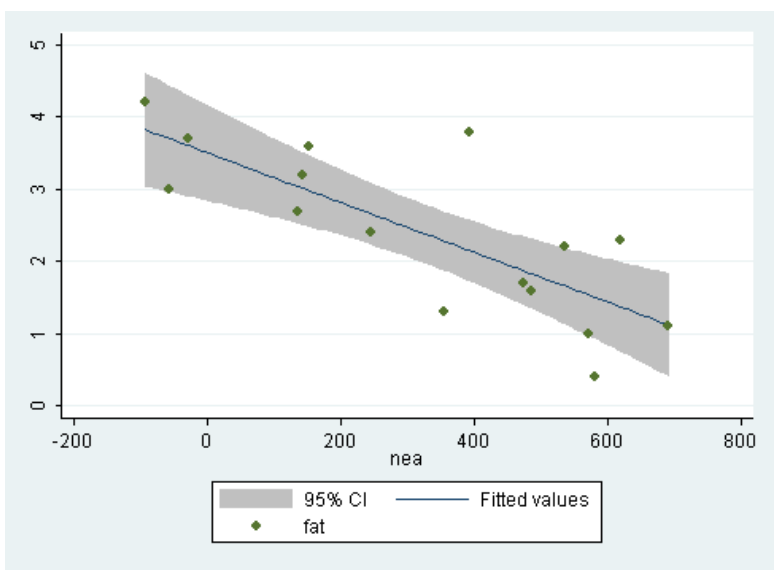


Figure 6: Linear predictor plot of fat with nea with 95% confidence limits for the mean response.

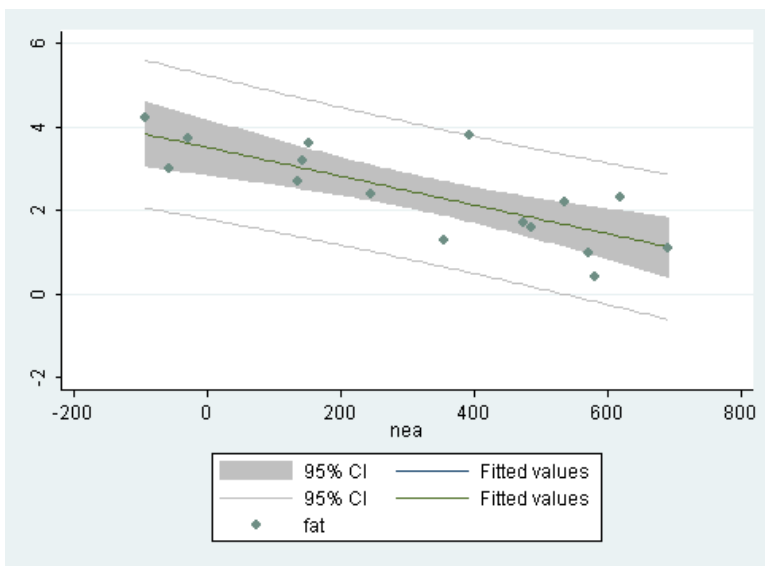


Figure 7: Linear predictor plot of fat with nea with 95% confidence limits for the mean response (`lfitci` with default `stdp` option; shaded area) and the 95% confidence limits for individual responses (`lfitci`, `stdf`; broader confidence band).