

Lab 3*Regression diagnostics --- are model assumptions met?*

In this lab we will discuss how to check whether model assumptions are met. Refer to the lec2.pdf on the main course website: [2. Linear Regression Model](#), pp. 18—23, “Checking the regression model”.

We will begin by reading in the bloodloss data from lecture 1. To continue the example in the lectures, from the labs website download the bloodloss data and load into Stata.

Look at this file, it contains column headers:

```
weight time bloodloss
44.3 105 503
40.6 80 490
69.0 86 471
43.7 112 505
50.3 109 482
50.2 100 490
35.4 96 513
52.2 120 464
```

When we attempt to read the data into Stata below, it expects to see a table of three columns of numbers, and does not expect these headers. This example is to show you what happens when Stata experiences text data when it expects numerical data. It puts in missing values “.”, an observation which we can then drop.

```
. infile weight time loss using stat539_bloodloss_data.txt
'weight' cannot be read as a number for weight[1]
'time' cannot be read as a number for time[1]
'bloodloss' cannot be read as a number for loss[1]
(9 observations read)
```

```
. list
```

```
+-----+
| weight   time   loss |
+-----+
1. |      .      .      . |
2. |  44.3   105   503 |
3. |  40.6    80   490 |
4. |   69    86   471 |
5. |  43.7   112   505 |
+-----+
6. |  50.3   109   482 |
7. |  50.2   100   490 |
8. |  35.4    96   513 |
9. |  52.2   120   464 |
+-----+
```

```
. help drop
```

```
. drop in 1/1
```

```
(1 observation deleted)
```

```
. list
```

```

+-----+
| weight   time   loss |
+-----+
1. |    44.3    105    503 |
2. |    40.6     80    490 |
   |          ...         |
8. |    52.2    120    464 |
+-----+

```

```
. save bloodloss, replace
```

We can then run the code on pp. 16—17 of lecture 2, which creates a set of new weights, appends them to our bloodloss dataset, then does a regression of loss on weight, and plots the confidence interval and prediction interval. The new weights are included because we want nice plots for the regression line and confidence and prediction bands. The `regress` and subsequent `predict` statements will fill in values for these new observations. I have added a few “list” commands to demonstrate what is happening.

```

clear
input weight
30
35
40
45
50
55
60
65
70
75
end
list
save weight.dta, replace
use bloodloss
list
append using weight
list
regress loss weight
predict loss_hat,xb
predict se_line, stdp
predict se_pred, stdf
generate lci=loss_hat-invttail(6,0.025)*se_line
generate uci=loss_hat+invttail(6,0.025)*se_line
generate lpi=loss_hat-invttail(6,0.025)*se_pred
generate upi=loss_hat+invttail(6,0.025)*se_pred
list
graph twoway (scatter loss weight) (line loss_hat weight) ///
             (line lci weight,sort)(line uci weight,sort) ///
             (line lpi weight,sort)(line upi weight, sort) ///
             , title(Blood Loss Data) subtitle(CI for Line and Prediction Int.)
** the /// slashes continue a line in a do file, but not in the command window
** in the command window, paste all the lines in, and remove the slashes and
** line feeds so that it appears all on one line

```

Ok, now that we're up to speed with our data, let's check our model assumptions.

Model assumptions

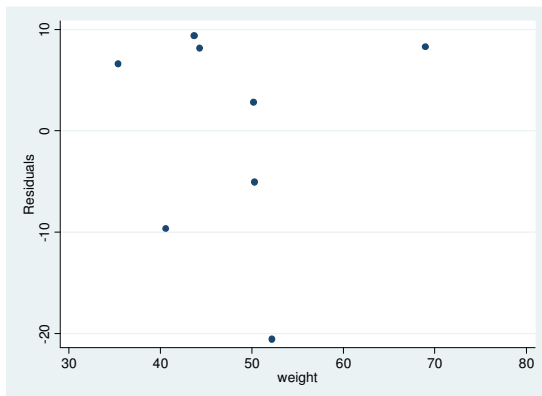
Our model assumptions are on the residuals. We assume the residuals are **independent**, **normally distributed**, have **constant variance** over the range of the predictor variables, and are **unstructured**.

We can not check independence from the data itself. Independence should be incorporated into the way the data were collected. Each observation should be independent, that is, no observation should influence any other observation.

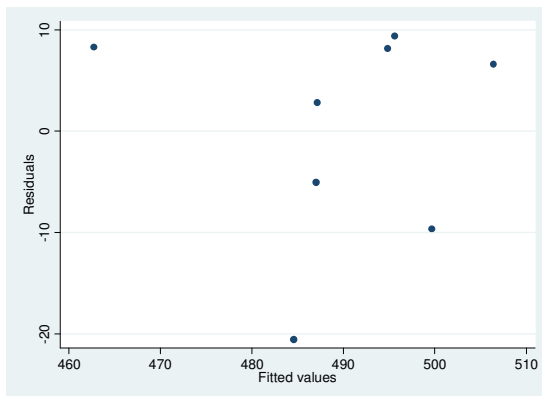
Residual Plots

See lecture 2 pp. 18—20 for examples of looking at residual plots (simply scatter plots with residuals on the vertical axis) for constant variance and structure.

```
. help rvppplot  
. rvppplot weight
```



```
. rvfplot
```



Create your residuals (called “e” for error), and list them to identify which observation has the large residual (far from zero):

```
. generate e=loss-loss_hat
```

Or let the predict command create the residuals; these are the “raw” residuals, which are the (observed value – predicted value).

```
. predict res, r
```

```
. list e r
```

	e	res
1.	8.162476	8.162469
2.	-9.648743	-9.648743
3.	8.280548	8.28055
4.	9.382263	9.382275
5.	-5.035553	-5.035568
6.	2.834412	2.834401
7.	6.589569	6.589561
8.	-20.56494	-20.56495
9.	.	.
10.	.	.
11.	.	.
12.	.	.
13.	.	.
14.	.	.
15.	.	.
16.	.	.
17.	.	.
18.	.	.

Note that `rvpplot` is equivalent to a scatter plot of the residuals versus the predictor variable.

```
. scatter e weight
```

Cook's Distance

Create a variable `cooksd`, which has the value of Cook's D in it

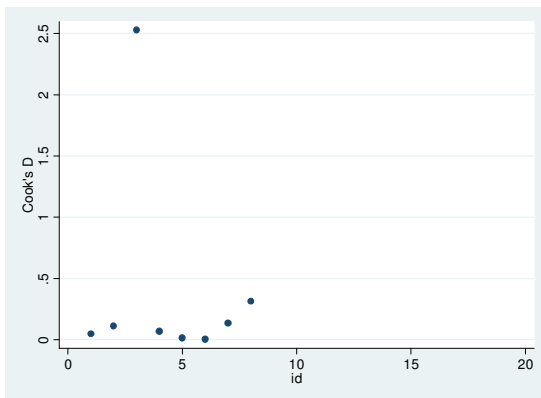
```
. predict cooksd, cooksd
```

Create a variable with the observation (`help _variables`):

```
. generate id=_n
```

Create a scatterplot for the value of Cook's distance. Observation 3 clearly has a much larger Cook's D than the other observations.

```
. scatter cooksd id
```



Normality of the residuals

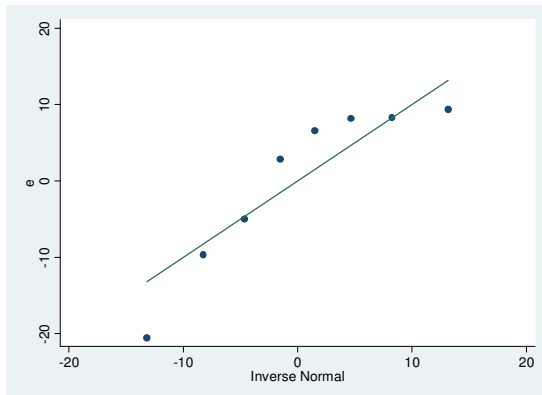
Shapiro-Wilk W test for normality (is similar to the Anderson-Darling test from Minitab last semester). A small p-value indicates that the data is not normal. In this case, the p-value=0.09204, which is small but not tiny (eg., not less than 0.05) so we might fail to reject the null hypothesis that the residuals are normal.

```
. swilk e
```

Shapiro-Wilk W test for normal data					
Variable	Obs	W	V	z	Prob>z
e	8	0.84852	2.110	1.328	0.09204

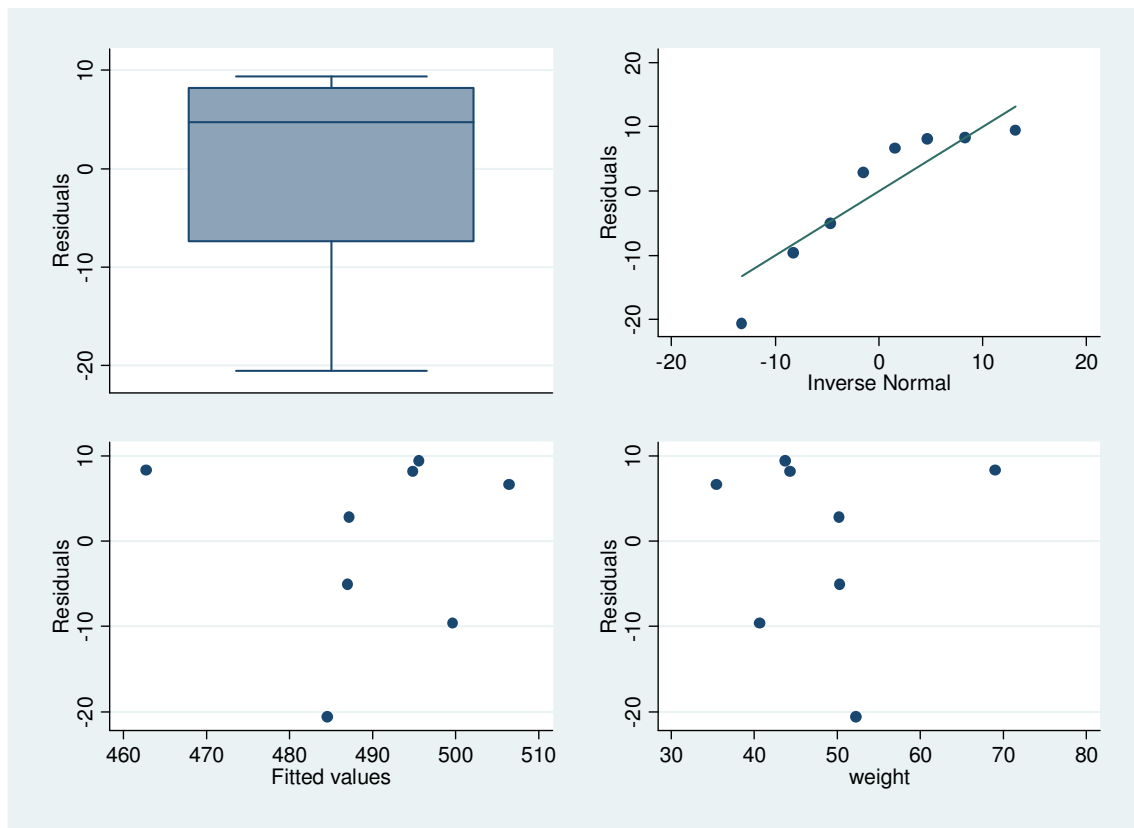
We can also create a normal probability plot of the residuals. If the residuals were normal, they would follow the line closely. In this case, we see some deviation from the line. The curvature indicates that the residuals are skewed, and not symmetric (probably due to the outlier we observed earlier).

```
. qnorm e
```



Creating a four-in-one plot including a residual boxplot, normal probability plot, residual scatterplot vs fitted values, and residual scatterplot vs predictor variable.

```
clear  
use bloodloss  
regress loss weight  
predict res, r  
swilk r  
graph box res, saving(boxplot)  
qnorm r, saving(probplot)  
rvfplot, saving(respredplot)  
rvpplot weight, saving(resweightplot)  
graph combine boxplot.gph probplot.gph respredplot.gph resweightplot.gph, saving(all)
```



From the boxplot, we see that the residuals are skewed to the left (long negative left tail). From the normal probability plot, the residuals do not follow the line very well. Both the residual scatter plots indicate the presence of one outlier.