Lab 4: Model diagnostics

Yifan Jin

Today's Objectives

- Regression assumptions and common issues may reduce the power of your model
- Basic on diagnostic plots: Residuals vs Fitted; Normal Q-Q plot; Scale-location
- What is outlier, leverage, collinearity? Residuals vs Leverage etc.

0. Regression assumptions and some common issues:

- Linearity: The true relationship between reponse and predictors is linear
- Normality: Errors are normally distributed
- Equal variances: the variances of errors are equal
- Independence of errors: observations are independet with each other

Some common issues:

- Outliers
- High Leverage Points
- Collinearity

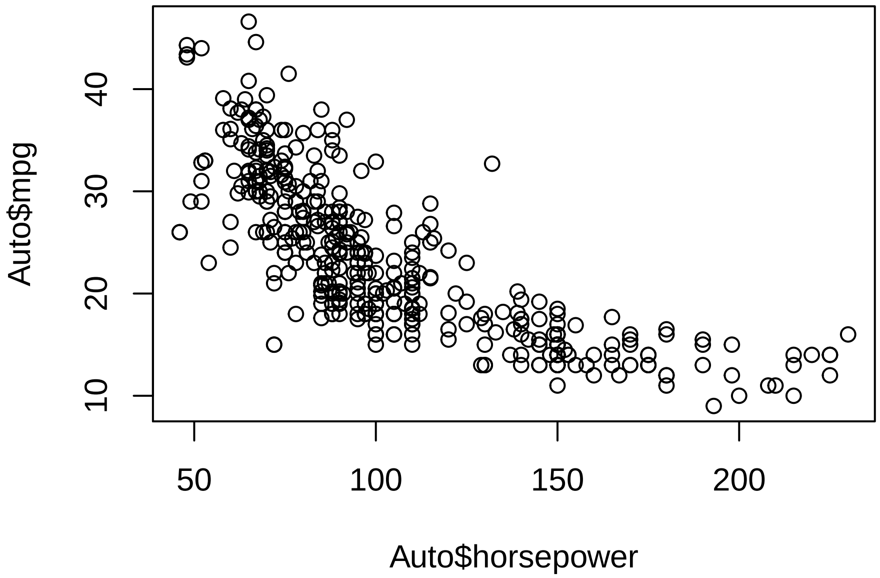
1. Basic on diagnostic plots

1.1 Residuals vs Fitted

We should look for two things in this plot.

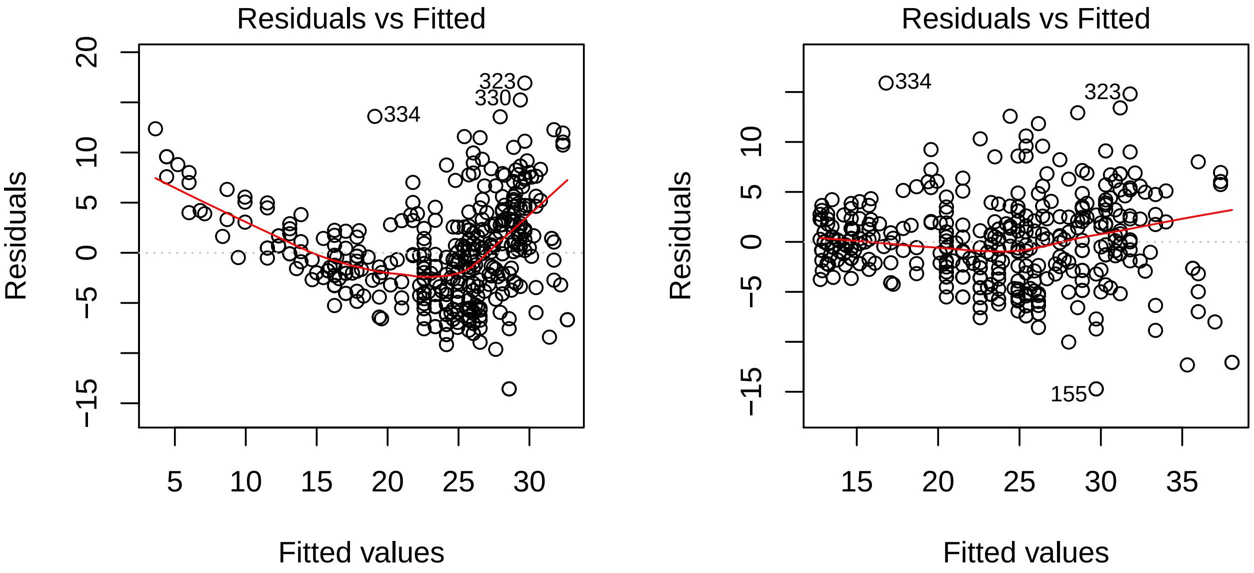
- At any fitted value, the mean of the residuals should be roughly 0. If this is the case, the linearity assumption is valid. For this reason, we generally add a horizontal line at $y = 0$ to emphasize this point.
- At every fitted value, the spread of the residuals should be roughly the same. If this is the case, the constant variance assumption is valid.

```r
library(ISLR)
plot(Auto$horsepower,Auto$mpg)
```



```
fit_linear=lm(mpg~horsepower,data=Auto)
fit_quad=lm(mpg~horsepower+I(horsepower^2),data=Auto)
```

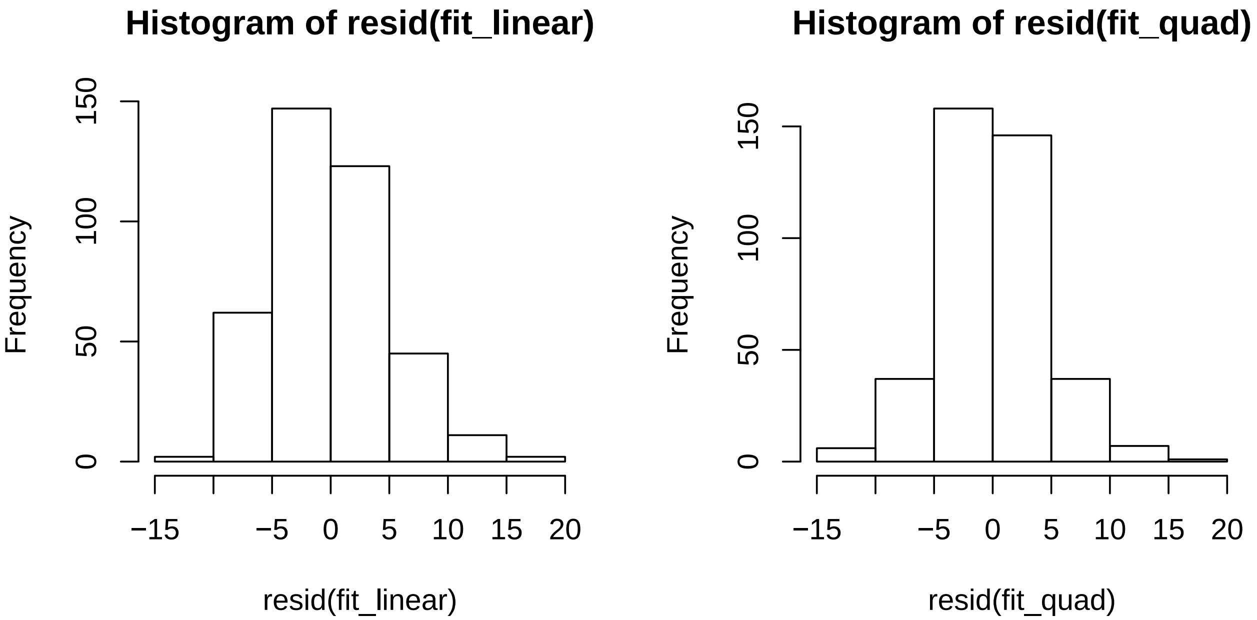
```
par(mfrow=c(1,2))
plot(fit_linear,which=1)
plot(fit_quad,which=1)
```



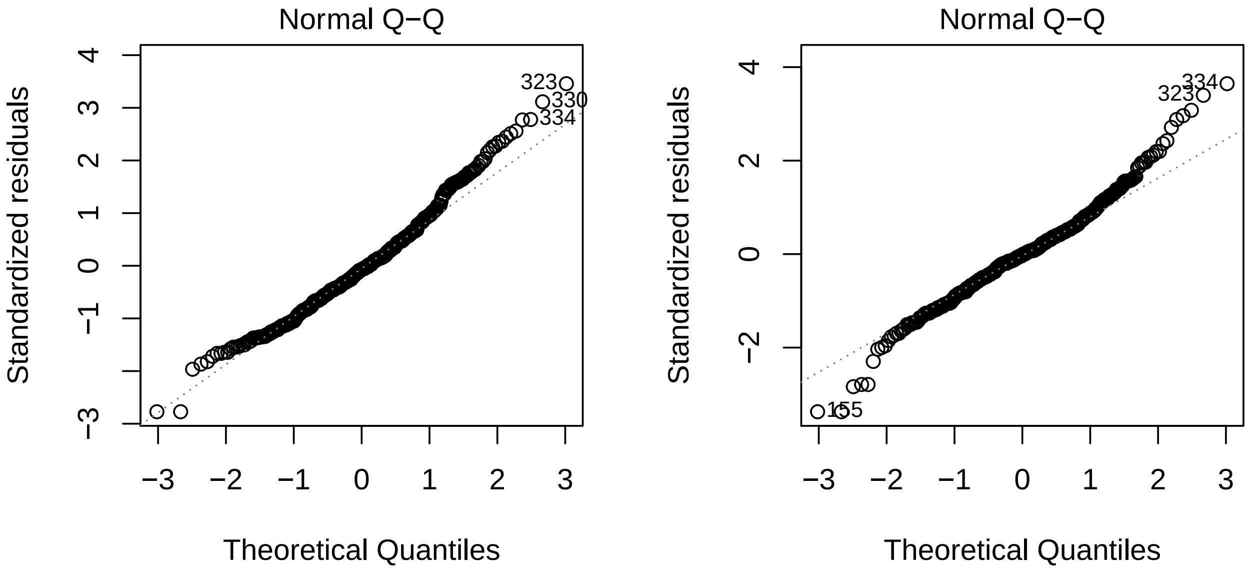
1.2 Normal Q-Q plot

QQ plot(a normal quantile-quantile plot) is a visual method for assessing the normality of errors, which is more powerful than a histogram. If the points of the plot do not closely follow a straight line, this would suggest that the data do not come from a normal distribution.

```
par(mfrow=c(1,2))
hist(resid(fit_linear))
hist(resid(fit_quad))
```



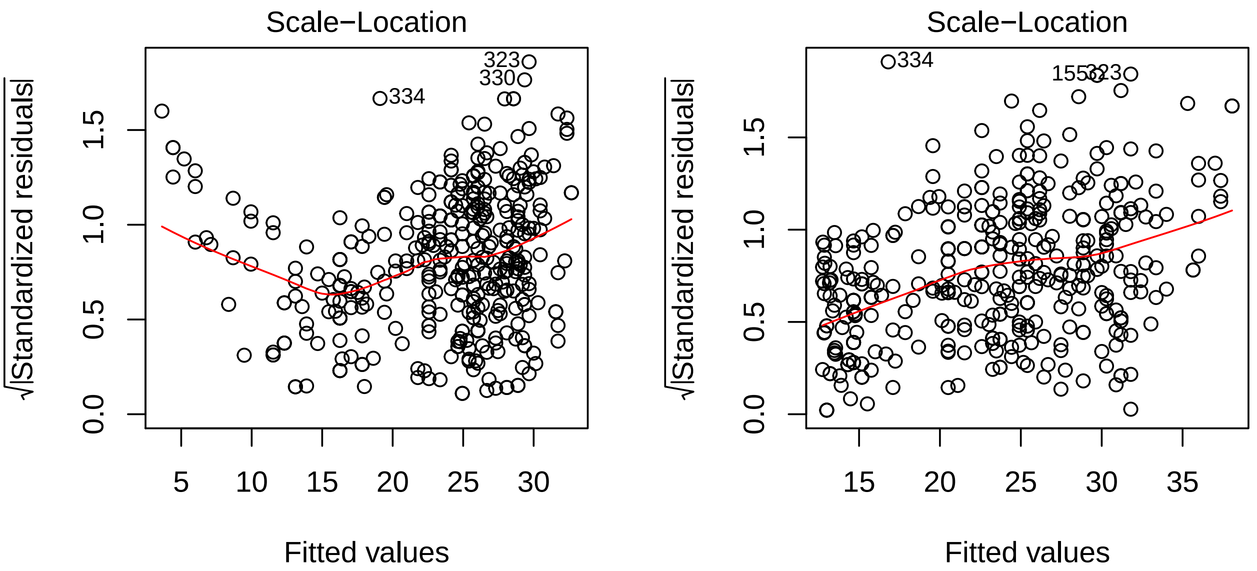
```r
par(mfrow=c(1,2))
plot(fit_linear,which=2)
plot(fit_quad,which=2)
```



1.3 Scale-location

It's also called 'Spread-Location' or 'S-L' plot. S-L plot is similar to the residual-fitted plot in the sense that you can check nonlinearity and the assumption of equal variance. But it takes the square root of the absolute residuals in order to diminish skewness. It's good if you see a horizontal line with equally (randomly) spread points.

```r
par(mfrow=c(1,2))
plot(fit_linear,which=3)
plot(fit_quad,which=3)
```



Some words on independence: To check independence, plot residuals against any time variables present (e.g., order of observation), any spatial variables present, and any variables used in the technique (e.g., factors, regressors). A pattern that is not random suggests lack of independence.

2. Outlier, leverage, collinearity

2.1 Outlier

Outliers are points which do not fit the model well. They may or may not have a large affect on the model. To identify outliers, we will look for observations with large residuals.

Let H to be the projection matrix,

$$e = y - \hat{y} = Iy - Hy = (I - H)y$$

Then, under the assumptions of linear regression,

$$Var(e_i) = (1 - h_i)\sigma^2$$

and thus estimating σ^2 with s_e^2 gives

$$SE[e_i] = s_e\sqrt{1 - h_i}$$

We can then look at the standardized residual for each observation,

$$r_i = \frac{e_i}{s_e\sqrt{1 - h_i}} \overset{\text{approx}}{\sim} N(0, 1)$$

when n is large.

We can use this fact to identify "large" residuals(in another word outlier is point deviated in y). For example, standardized residuals greater than 2 in magnitude should only happen approximately 5 percent of the time.

Returning again to our three plots, we can calculate the residuals and standardized residuals for each. Standardized residuals can be obtained in R by using `rstandard()` where we would normally use `resid()`.

```r
resid(fit_quad)[1:20]
```

```
##           1           2           3           4           5           6
##   0.9084921   1.5198439   3.3412825   1.3412825   1.2479409   2.1635097
##           7           8           9          10          11          12
##   0.1036716   0.4491395  -0.3033231   2.2535767   1.7896441   0.1885169
##          13          14          15          16          17          18
##   0.3412825  -0.3033231   0.2823268  -1.7176732  -5.2578198  -5.1646045
##          19          20
##   1.5953162 -12.0591911
```

```
rstandard(fit_quad)[1:20]
```

```
##           1           2           3           4           5           6
##  0.20838032  0.34904638  0.76677070  0.30780281  0.28631326  0.50133307
##           7           8           9          10          11          12
##  0.02450447  0.10551886 -0.07221688  0.52025309  0.41121750  0.04327939
##          13          14          15          16          17          18
##  0.07831886 -0.07221688  0.06466022 -0.39339205 -1.20425889 -1.18270767
##          19          20
##  0.36532426 -2.78985276
```

```
rstandard(fit_quad)[abs(rstandard(fit_quad)) > 2]
```

```
##        20        60       103       155       156       197       201
## -2.789853 -2.836386 -2.789853 -3.372439 -3.372439 -2.035817 -2.296156
##       248       307       310       321       323       325       330
##  2.087202  2.117633  2.960661  2.879573  3.397282  2.066204  3.076382
##       334       336       358       380       387       391
##  3.646286  2.197308  2.362500  2.426307  2.710336  2.193237
```

2.2 Leverage

A data point with high leverage, is a data point that could have a large influence when fitting the model.

The diagonal elements of projection matrix are called the leverages:

$$H_{ii} = h_i$$

where h_{i} is the leverage for the ith observation. Large values of h_i indicate extreme values in X(deviation in X!) , which may influence regression. Note that leverages only depend on X. Here, p is the number of predictors and is also the trace (and rank) of the hat matrix.

$$\sum_{i=1} h_i = p$$

What is a value of h_i that would be considered large? There is no exact answer to this question. A common heuristic would be to compare each leverage to two times the average leverage. A leverage larger than this is considered an observation to be aware of. That is, if

$$h_i > 2\hat{h}$$

For simple linear regression, the leverage for each point is given by $h_i = \frac{1}{n} + \frac{(x_i-\bar{x})^2}{\sum_{i'=1}^{n}(x_{i'}-\bar{x})^2}$. It suggests that the large leverages occur when x values are far from their mean. To find leverage in R:

```
hatvalues(fit_quad)[1:20]
```

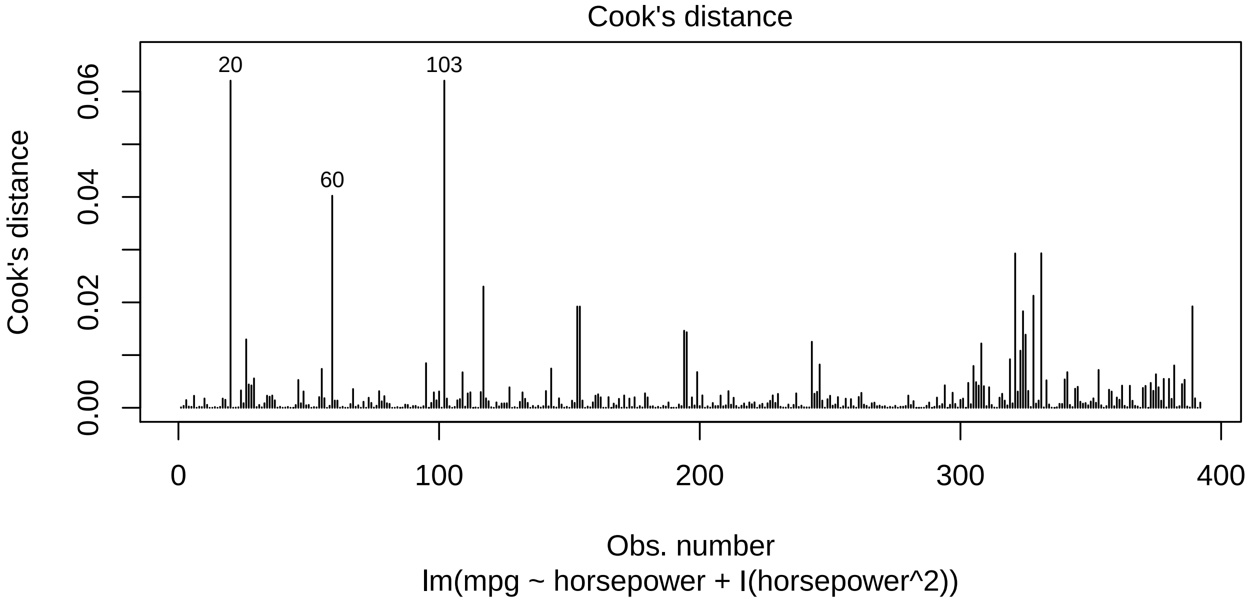
```
##           1           2           3           4           5           6
## 0.006455933 0.008962923 0.007447243 0.007447243 0.006967783 0.026528558
##           7           8           9          10          11          12
## 0.064406709 0.052977076 0.077870991 0.019215522 0.009969958 0.008262817
##          13          14          15          16          17          18
## 0.007447243 0.077870991 0.003475186 0.003475186 0.003608790 0.003269996
##          19          20
## 0.003230513 0.023365619
```

```
hatvalues(fit_quad)[hatvalues(fit_quad)>2*mean(hatvalues(fit_quad))]
```

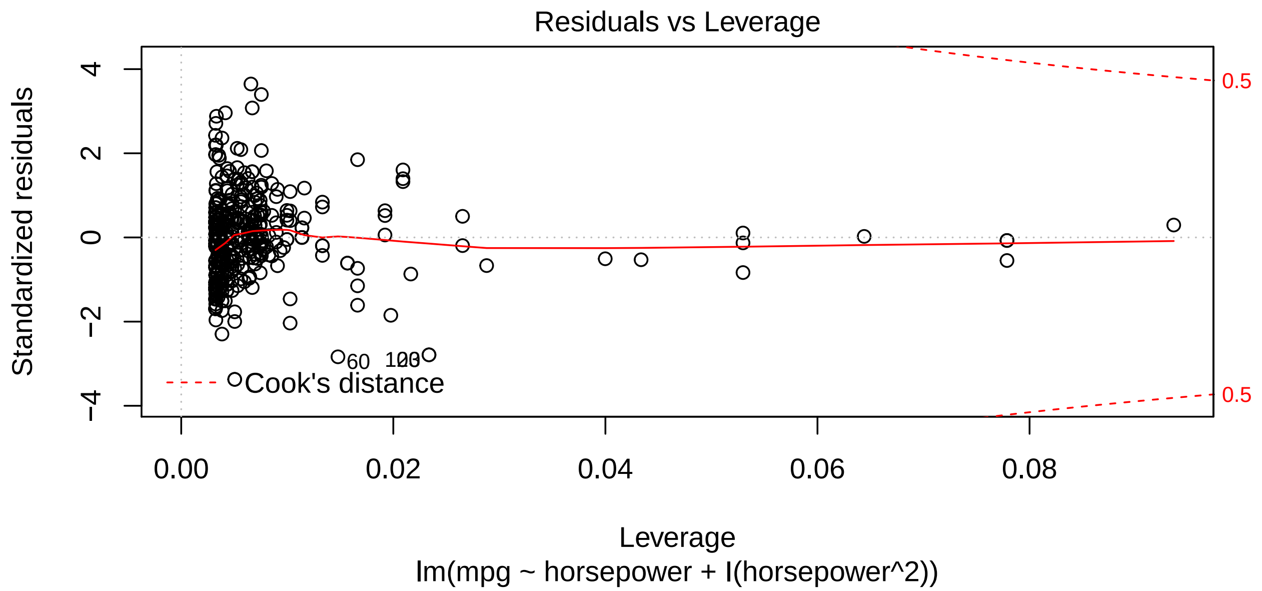
```
##          6          7          8          9         10         14
## 0.02652856 0.06440671 0.05297708 0.07787099 0.01921552 0.07787099
##         20         26         27         28         29         68
## 0.02336562 0.05297708 0.02879961 0.04336660 0.02164870 0.03998573
##         71         91         95         96        103        117
## 0.01921552 0.02652856 0.05297708 0.07787099 0.02336562 0.09359709
##        118        145        182        196        199        232
## 0.01976368 0.01662896 0.01568131 0.01662896 0.01568131 0.01921552
##        245        247        326        327        394
## 0.02091050 0.01662896 0.02091050 0.02091050 0.01662896
```

You can visualize this by residuals vs leverage plot. When cases are outside of the Cook's distance (meaning they have high Cook's distance scores), the cases are influential to the regression results. The regression results will be altered if we exclude those cases.

```
plot(fit_quad,which=4)
```



```
plot(fit_quad,which=5)
```



Collinearity

- The presence of collinearity can pose problems in the regression context, since it can be difficult to separate out the individual effects of collinear variables on the response
- Since collinearity reduces the accuracy of the estimates of the regression coefficients, it causes the standard error for $\hat{\beta}_j$ to grow.
- Recall that the t-statistic for each predictor is calculated by dividing by its standard error. Consequently, collinearity results in a decline in the t-statistic.

```
library(ISLR)
data(Credit)
model1 = lm(Balance ~ Age + Limit, data = Credit)
summary(model1)
```

```
##
## Call:
## lm(formula = Balance ~ Age + Limit, data = Credit)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -696.84 -150.78  -13.01  126.68  755.56
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.734e+02  4.383e+01  -3.957 9.01e-05 ***
## Age         -2.291e+00  6.725e-01  -3.407 0.000723 ***
## Limit        1.734e-01  5.026e-03  34.496  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 230.5 on 397 degrees of freedom
## Multiple R-squared:  0.7498, Adjusted R-squared:  0.7486
## F-statistic:   595 on 2 and 397 DF,  p-value: < 2.2e-16
```

```r
model2 = lm(Balance ~ Rating + Limit, data = Credit)
summary(model2)
```

```
##
## Call:
## lm(formula = Balance ~ Rating + Limit, data = Credit)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -707.8 -135.9   -9.5  124.0  817.6
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -377.53680   45.25418  -8.343 1.21e-15 ***
## Rating         2.20167    0.95229   2.312   0.0213 *
## Limit          0.02451    0.06383   0.384   0.7012
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 232.3 on 397 degrees of freedom
## Multiple R-squared:  0.7459, Adjusted R-squared:  0.7447
## F-statistic: 582.8 on 2 and 397 DF,  p-value: < 2.2e-16
```

The standard error of $\hat{beta}_{Limit}$ increases 12-fold in the second regression, due to collinearity. Also notice the increase of p-values.

A simple way to detect collinearity is to look at the correlation matrix of the predictors. Unfortunately, not all collinearity problems can be detected by inspection of the correlation matrix: it is possible for collinearity to exist between three or more variables even if no pair of variables has a particularly high correlation.

Variance inflation factor(VIF)

The VIF is the ratio of the variance of $\hat{\beta}_j$ when fitting the full model divided by the variance of $\hat{\beta}_j$ if fit on its own.

$$VIF(\hat{\beta}_j) = \frac{1}{1 - R^2_{X_j|X_{-j}}}$$

where $R^2_{X_j|X_{-j}}$ is the R^2 from a regression of X_j onto all of the other predictos.

```r
library(faraway)
vif(model2)
```

```
##   Rating    Limit
## 160.4933 160.4933
```

In the Credit data, a regression of balance on age, rating, and limit indicates that the predictors have VIF values of 160.4933, and 160.4933. As we suspected, there is considerable collinearity in the data!

If you have interest to see more cases, recommond to read https://daviddalpiaz.github.io/appliedstats/model-diagnostics.html and https://daviddalpiaz.github.io/appliedstats/collinearity.html