title: “HW1_Linear”
author: “Xinyue Lu”
date: “2020/1/21”
output: html_document

9. This exercise involves the Auto data set studied in the lab. Make sure that the missing values have been removed from the data.

Load data and remove missing values.

```
auto <- read.csv("D:/luxinyve/00 Linear graph/HW1/Auto.csv",header=T,na.strings ="?")
dim(auto)
```

```
## [1] 397   9
```

```
auto = na.omit(auto)
dim(auto)
```

```
## [1] 392   9
```

a. Which of the predictors are quantitative, and which are qualitative?

Name and origin are quantitative. The rest are qualitative.

```
str(auto)
```

```
## 'data.frame':    392 obs. of  9 variables:
##  $ mpg         : num  18 15 18 16 17 15 14 14 14 15 ...
##  $ cylinders   : int  8 8 8 8 8 8 8 8 8 8 ...
##  $ displacement: num  307 350 318 304 302 429 454 440 455 390 ...
##  $ horsepower  : int  130 165 150 150 140 198 220 215 225 190 ...
##  $ weight      : int  3504 3693 3436 3433 3449 4341 4354 4312 4425 3850 ...
##  $ acceleration: num  12 11.5 11 12 10.5 10 9 8.5 10 8.5 ...
##  $ year        : int  70 70 70 70 70 70 70 70 70 70 ...
##  $ origin      : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ name        : Factor w/ 304 levels "amc ambassador brougham",..: 49 36 231 14 161 141 54 223 241 2 ...
##  - attr(*, "na.action")= 'omit' Named int  33 127 331 337 355
##   ..- attr(*, "names")= chr  "33" "127" "331" "337" ...
```

```
summary(auto)
```

```
##       mpg          cylinders      displacement     horsepower
##  Min.   : 9.00   Min.   :3.000   Min.   : 68.0   Min.   : 46.0
##  1st Qu.:17.00   1st Qu.:4.000   1st Qu.:105.0   1st Qu.: 75.0
##  Median :22.75   Median :4.000   Median :151.0   Median : 93.5
##  Mean   :23.45   Mean   :5.472   Mean   :194.4   Mean   :104.5
##  3rd Qu.:29.00   3rd Qu.:8.000   3rd Qu.:275.8   3rd Qu.:126.0
##  Max.   :46.60   Max.   :8.000   Max.   :455.0   Max.   :230.0
##
##      weight      acceleration        year           origin
##  Min.   :1613   Min.   : 8.00   Min.   :70.00   Min.   :1.000
##  1st Qu.:2225   1st Qu.:13.78   1st Qu.:73.00   1st Qu.:1.000
##  Median :2804   Median :15.50   Median :76.00   Median :1.000
##  Mean   :2978   Mean   :15.54   Mean   :75.98   Mean   :1.577
##  3rd Qu.:3615   3rd Qu.:17.02   3rd Qu.:79.00   3rd Qu.:2.000
##  Max.   :5140   Max.   :24.80   Max.   :82.00   Max.   :3.000
##
##                  name
##  amc matador       :  5
##  ford pinto        :  5
##  toyota corolla    :  5
##  amc gremlin       :  4
##  amc hornet        :  4
##  chevrolet chevette:  4
##  (Other)           :365
```

b. What is the range of each quantitative predictor? You can answer this using the range() function.

```
attach(auto)
col = colnames(auto)
for (i in 1:8){
cat('The range of',col[i],'is [',range(auto[,i])[1],',',range(auto[,i])[2],'].\n')}
```

```
## The range of mpg is [ 9 , 46.6 ].
## The range of cylinders is [ 3 , 8 ].
## The range of displacement is [ 68 , 455 ].
## The range of horsepower is [ 46 , 230 ].
## The range of weight is [ 1613 , 5140 ].
## The range of acceleration is [ 8 , 24.8 ].
## The range of year is [ 70 , 82 ].
## The range of origin is [ 1 , 3 ].
```

c. What is the mean and standard deviation of each quantitative predictor?

```
for (i in 1:8){
cat('The mean of',col[i],'is',mean(auto[,i]),'. The standard deviation is',sd(auto[,i]),'\n')}
```

```
## The mean of mpg is 23.44592 . The standard deviation is 7.805007
## The mean of cylinders is 5.471939 . The standard deviation is 1.705783
## The mean of displacement is 194.412 . The standard deviation is 104.644
## The mean of horsepower is 104.4694 . The standard deviation is 38.49116
## The mean of weight is 2977.584 . The standard deviation is 849.4026
## The mean of acceleration is 15.54133 . The standard deviation is 2.758864
## The mean of year is 75.97959 . The standard deviation is 3.683737
## The mean of origin is 1.576531 . The standard deviation is 0.8055182
```

d. Now remove the 10th through 85th observations. What is the range, mean, and standard deviation of each predictor in the subset of the data that remains?

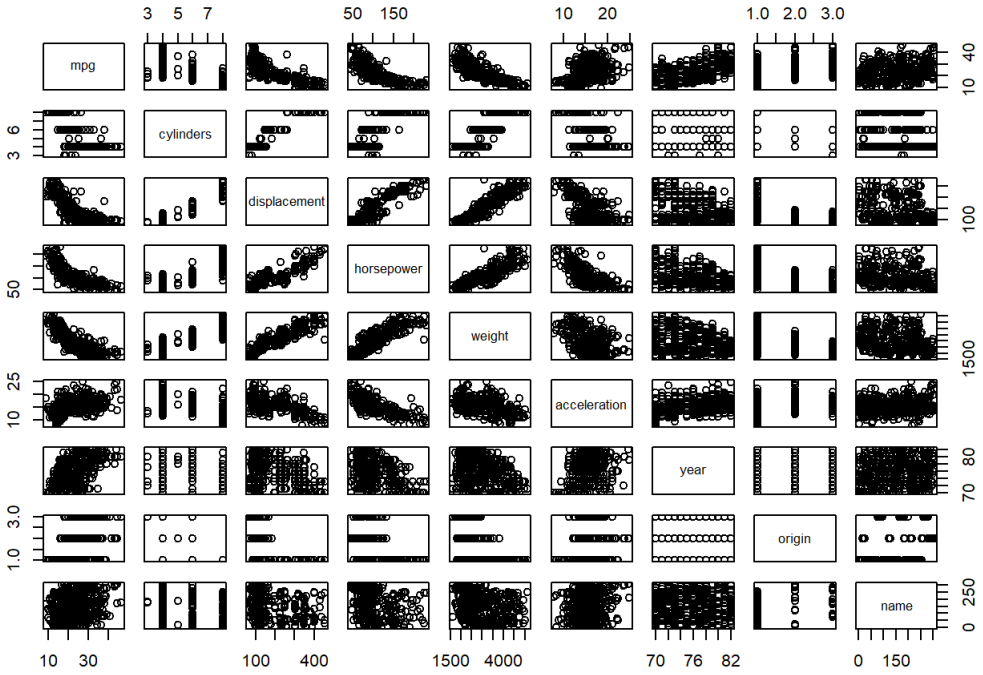
```
auto.2 <- auto[-c(10:85),]
for (i in 1:8){
cat('The range of',col[i],'is [',range(auto.2[,i])[1],',',range(auto[,i])[2],']. The mean is',mean(auto.2[,i]),'. The standard dev
iation is',sd(auto.2[,i]),'\n')}
```

```
## The range of mpg is [ 11 , 46.6 ]. The mean is 24.40443 . The standard deviation is 7.867283
## The range of cylinders is [ 3 , 8 ]. The mean is 5.373418 . The standard deviation is 1.654179
## The range of displacement is [ 68 , 455 ]. The mean is 187.2405 . The standard deviation is 99.67837
## The range of horsepower is [ 46 , 230 ]. The mean is 100.7215 . The standard deviation is 35.70885
## The range of weight is [ 1649 , 5140 ]. The mean is 2935.972 . The standard deviation is 811.3002
## The range of acceleration is [ 8.5 , 24.8 ]. The mean is 15.7269 . The standard deviation is 2.693721
## The range of year is [ 70 , 82 ]. The mean is 77.14557 . The standard deviation is 3.106217
## The range of origin is [ 1 , 3 ]. The mean is 1.601266 . The standard deviation is 0.81991
```

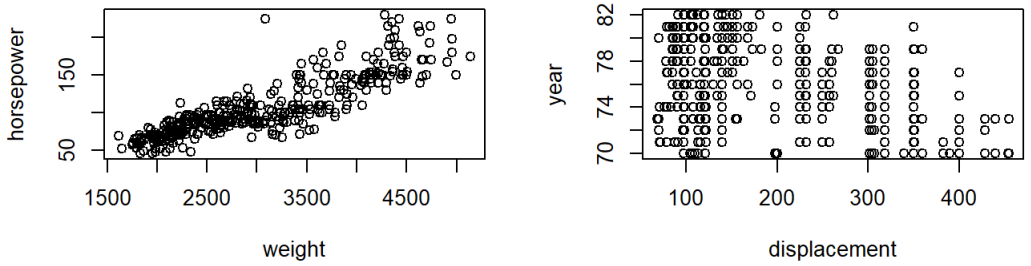
e. Using the full data set, investigate the predictors graphically, using scatterplots or other tools of your choice. Create some plots highlighting the relationships among the predictors. Comment on your findings.

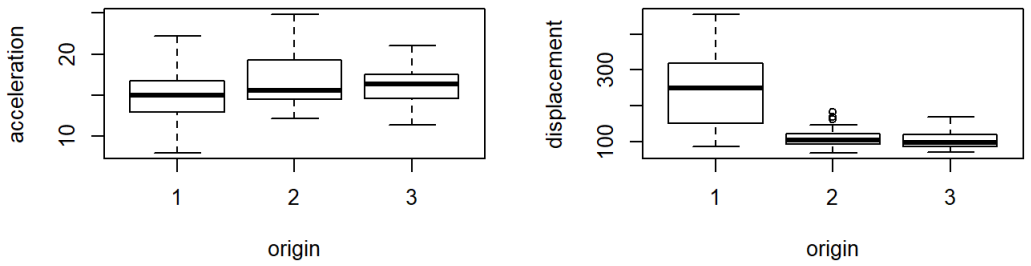
The scatterplot matrix of all the variables showed that some variables had clear linear relashionships and others did not. For example, there was clear linear relashionship between weight and horsepower while the linear relashionship between displacement and year was not significant. Besides, different origins had different displacement distributions while the distributions of acceleration were not so different.

```
plot(auto)
```




```
par(mfrow=c(2,2))
plot(weight,horsepower)
plot(displacement,year)
plot(as.factor(origin),acceleration, xlab='origin',ylab='acceleration' )
plot(as.factor(origin),displacement,xlab='origin',ylab='displacement')
```





f. Suppose that we wish to predict gas mileage (mpg) on the basis of the other variables. Do your plots suggest that any of the other variables might be useful in predicting mpg? Justify your answer.

Accroding to the scatterplot matix, displacement, horsepower and weight might be useful. So I built a linear model containing all the three variables. The model showed that the effects of horsepower and weight were significant, but the effect of displacement was not. However, displacement along was significant and the model R-squared was 0.648, which was not low.

```
model.1 <- lm(mpg~displacement+acceleration+horsepower+weight)
summary(model.1)
```

```
##
## Call:
## lm(formula = mpg ~ displacement + acceleration + horsepower +
##     weight)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -11.378  -2.793  -0.333   2.193  16.256
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  45.2511397  2.4560447  18.424  < 2e-16 ***
## displacement -0.0060009  0.0067093  -0.894  0.37166
## acceleration -0.0231480  0.1256012  -0.184  0.85388
## horsepower   -0.0436077  0.0165735  -2.631  0.00885 **
## weight       -0.0052805  0.0008109  -6.512  2.3e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.247 on 387 degrees of freedom
## Multiple R-squared:  0.707,  Adjusted R-squared:  0.704
## F-statistic: 233.4 on 4 and 387 DF,  p-value: < 2.2e-16
```

```
model.2 <- lm(mpg~displacement)
summary(model.2)
```

```
##
## Call:
## lm(formula = mpg ~ displacement)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.9170  -3.0243  -0.5021   2.3512  18.6128
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  35.12064    0.49443   71.03   <2e-16 ***
## displacement -0.06005    0.00224  -26.81   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.635 on 390 degrees of freedom
## Multiple R-squared:  0.6482, Adjusted R-squared:  0.6473
## F-statistic: 718.7 on 1 and 390 DF,  p-value: < 2.2e-16
```

8. This question involves the use of simple linear regression on the Auto data set.

a. Use the lm() function to perform a simple linear regression with mpg as the response and horsepower as the predictor. Use the summary() function to print the results. Comment on the output. For example:

i. Is there a relationship between the predictor and the response?
ii. How strong is the relationship between the predictor and the response?
iii. Is the relationship between the predictor and the response positive or negative?
iv. What is the predicted mpg associated with a horsepower of 98? What are the associated 95% confidence and prediction intervals?

The linear relationship between mpg and horsepower was very significant strong and negetive, accroding to the correlation coefficient -0.778. The p-value of horsepower was significant and the R-squared was not low.

The predicted mpg with horsepower 98 was 24.46708. The 95% confidence interval was [23.97308, 24.96108]. The 95% prediction interval was [14.8094, 34.12476].

```
cor(mpg,horsepower)
```

```
## [1] -0.7784268
```

```
model.a <- lm(mpg~horsepower)
summary(model.a)
```

```
##
## Call:
## lm(formula = mpg ~ horsepower)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.5710  -3.2592  -0.3435   2.7630  16.9240
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 39.935861   0.717499   55.66   <2e-16 ***
## horsepower  -0.157845   0.006446  -24.49   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.906 on 390 degrees of freedom
## Multiple R-squared:  0.6059, Adjusted R-squared:  0.6049
## F-statistic: 599.7 on 1 and 390 DF,  p-value: < 2.2e-16
```

```
predict(model.a, data.frame(horsepower=98),interval = "confidence")
```

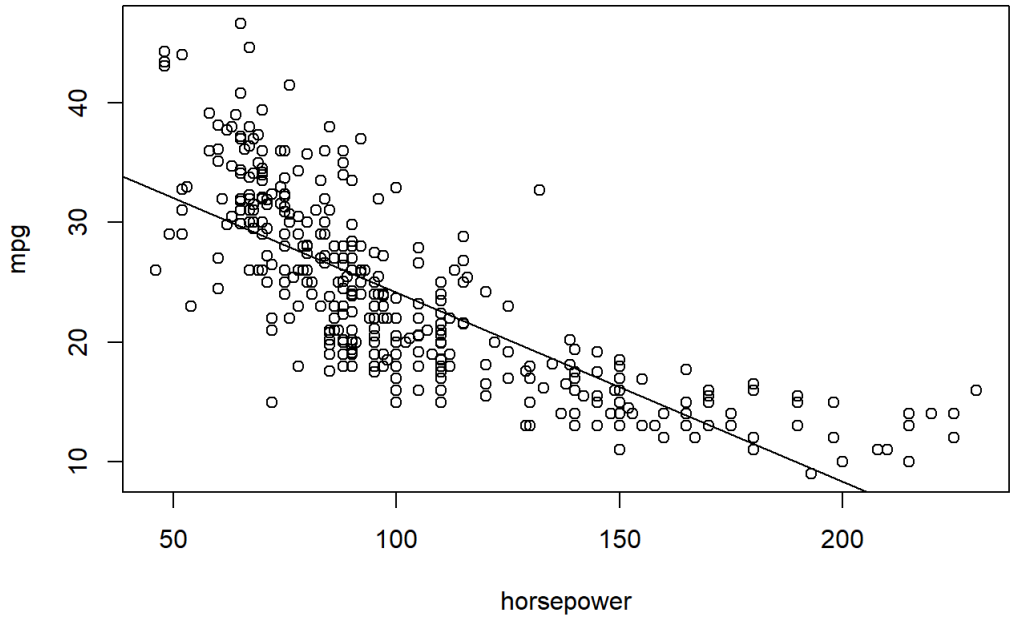
```
##        fit      lwr      upr
## 1 24.46708 23.97308 24.96108
```

```
predict(model.a, data.frame(horsepower=98),interval = "prediction")
```

```
##        fit     lwr      upr
## 1 24.46708 14.8094 34.12476
```

b. Plot the response and the predictor. Use the abline() function to display the least squares regression line.

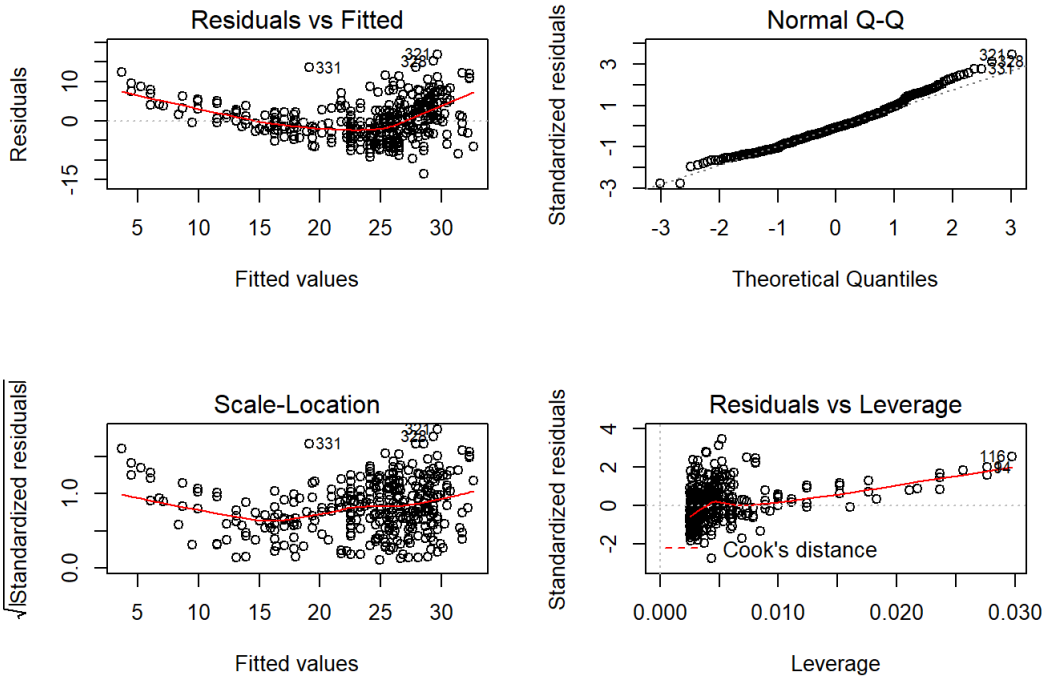
```
s=summary(model.a)
plot(horsepower,mpg)
abline(s$coef[1],s$coef[2])
```



c. Use the plot() function to produce diagnostic plots of the least squares regression fit. Comment on any problems you see with the fit.

The 'Residuals vs Fitted' plot showed a 'curve', which implied that there were higher order relashionship between mpg and horsepower. The 'QQ-plot' showed that the residuals were not very normal. The 'Residuals vs Leverage' plot showed that there were many observations had strong influence on the model and might be abnormal.

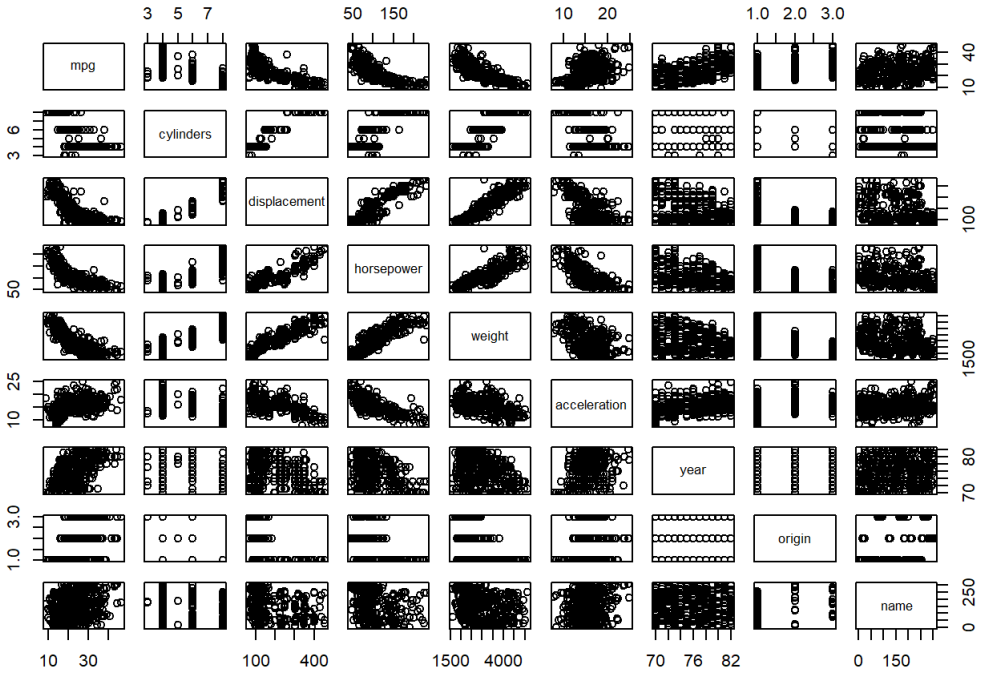
```
par(mfrow=c(2,2))
plot(model.a)
```



9. This question involves the use of multiple linear regression on the Auto data set.

a. Produce a scatterplot matrix which includes all of the variables in the data set.

```
plot(auto)
```



b. Compute the matrix of correlations between the variables using the function cor(). You will need to exclude the name variable, which is qualitative.

```
cor(auto[,1:8])
```

```
##                     mpg  cylinders displacement horsepower     weight
## mpg           1.0000000 -0.7776175   -0.8051269 -0.7784268 -0.8322442
## cylinders    -0.7776175  1.0000000    0.9508233  0.8429834  0.8975273
## displacement -0.8051269  0.9508233    1.0000000  0.8972570  0.9329944
## horsepower   -0.7784268  0.8429834    0.8972570  1.0000000  0.8645377
## weight       -0.8322442  0.8975273    0.9329944  0.8645377  1.0000000
## acceleration  0.4233285 -0.5046834   -0.5438005 -0.6891955 -0.4168392
## year          0.5805410 -0.3456474   -0.3698552 -0.4163615 -0.3091199
## origin        0.5652088 -0.5689316   -0.6145351 -0.4551715 -0.5850054
##              acceleration       year     origin
## mpg             0.4233285  0.5805410  0.5652088
## cylinders      -0.5046834 -0.3456474 -0.5689316
## displacement   -0.5438005 -0.3698552 -0.6145351
## horsepower     -0.6891955 -0.4163615 -0.4551715
## weight         -0.4168392 -0.3091199 -0.5850054
## acceleration    1.0000000  0.2903161  0.2127458
## year            0.2903161  1.0000000  0.1815277
## origin          0.2127458  0.1815277  1.0000000
```

c. Use the lm() function to perform a multiple linear regression with mpg as the response and all other variables except name as the predictors. Use the summary() function to print the results. Comment on the output. For instance:

i. Is there a relationship between the predictors and the response?
ii. Which predictors appear to have a statistically significant relationship to the response?
iii. What does the coeffcient for the year variable suggest?

The model was significant. Significant predictors were displacement, weight, year and origin. The coeffcient for the year suggested that the later the car was manufactured, the more mpg it had. However, horsepower was not significant in this model.

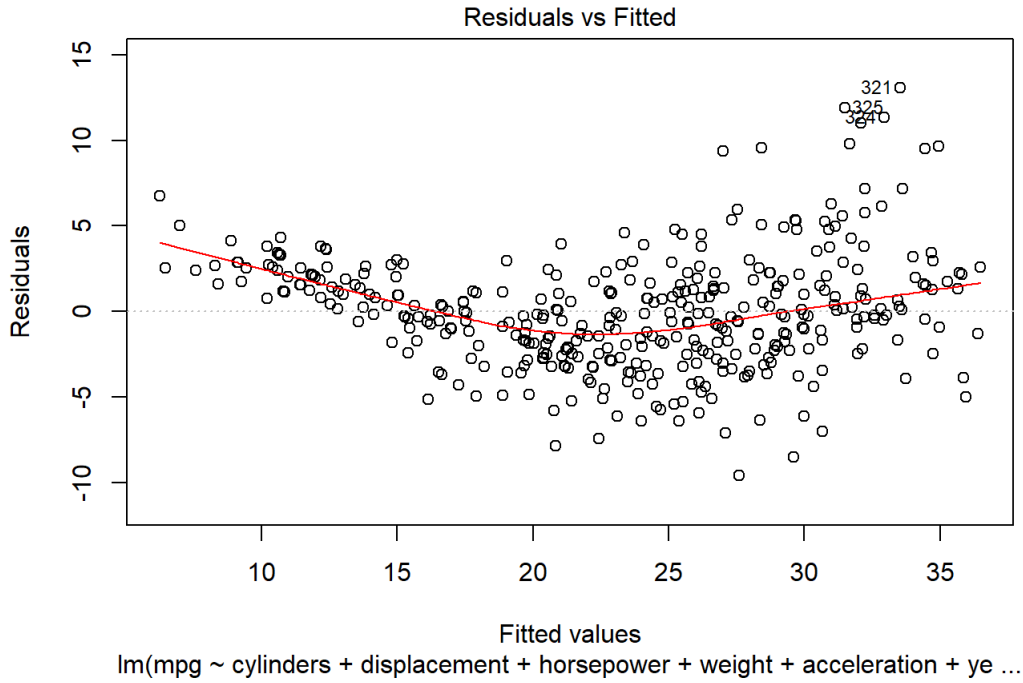
```
model.c <- lm(mpg~cylinders+displacement+horsepower+weight+acceleration+year+origin)
summary(model.c)
```

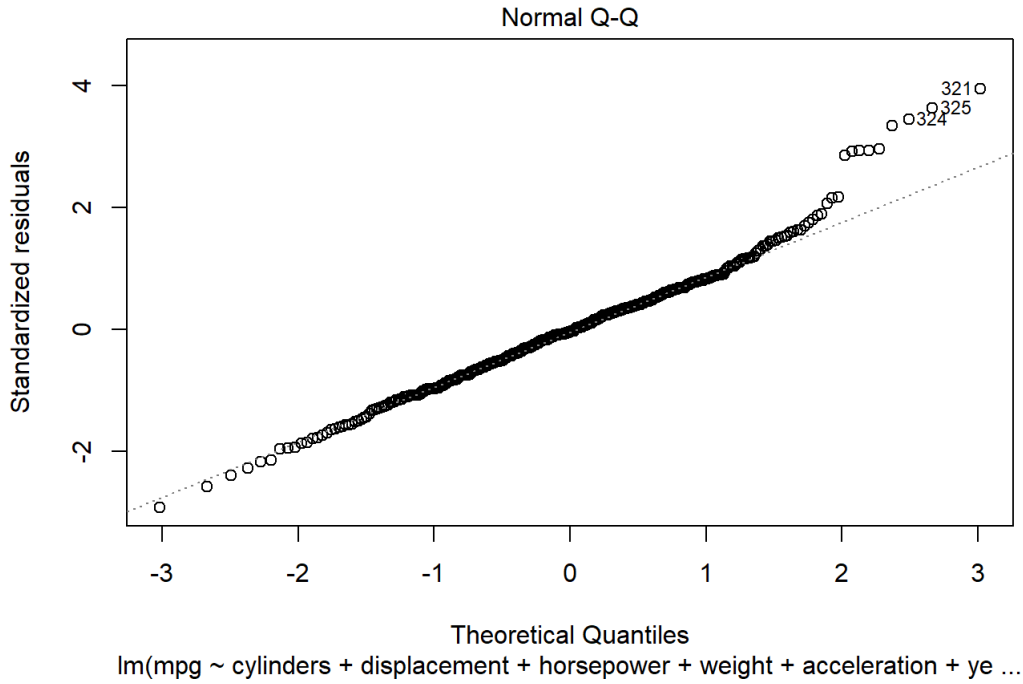
```
##
## Call:
## lm(formula = mpg ~ cylinders + displacement + horsepower + weight +
##     acceleration + year + origin)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.5903 -2.1565 -0.1169  1.8690 13.0604
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -17.218435   4.644294  -3.707  0.00024 ***
## cylinders     -0.493376   0.323282  -1.526  0.12780
## displacement   0.019896   0.007515   2.647  0.00844 **
## horsepower    -0.016951   0.013787  -1.230  0.21963
## weight        -0.006474   0.000652  -9.929  < 2e-16 ***
## acceleration   0.080576   0.098845   0.815  0.41548
## year           0.750773   0.050973  14.729  < 2e-16 ***
## origin         1.426141   0.278136   5.127 4.67e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.328 on 384 degrees of freedom
## Multiple R-squared:  0.8215, Adjusted R-squared:  0.8182
## F-statistic: 252.4 on 7 and 384 DF,  p-value: < 2.2e-16
```

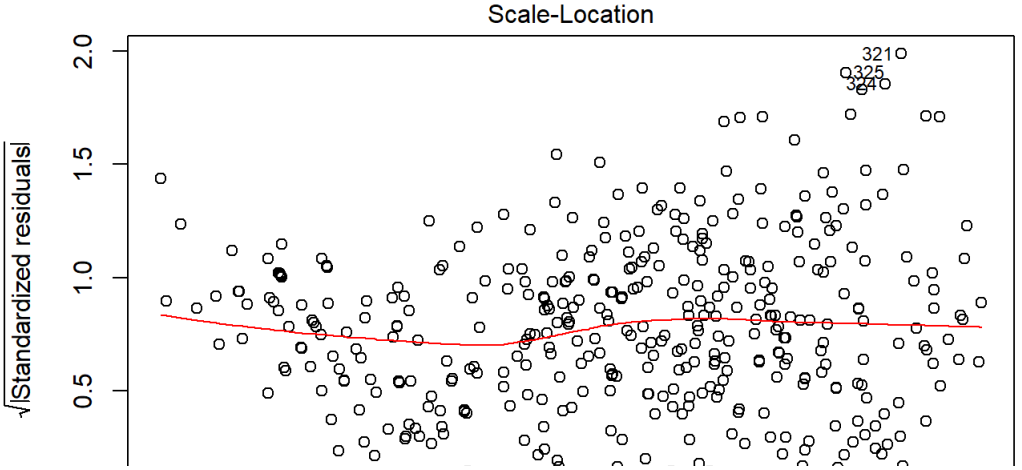
d. Use the plot() function to produce diagnostic plots of the linear regression fit. Comment on any problems you see with the fit. Do the residual plots suggest any unusually large outliers? Does the leverage plot identify any observations with unusually high leverage?

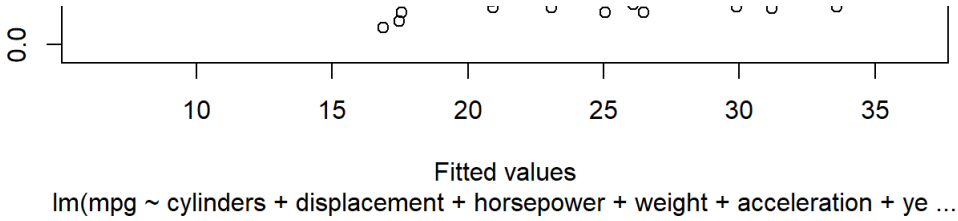
The residuals were positive when the fitted values were relative small or large. And The residuals were pretty large when fitted values were large. The 321, 325, 324 observations' residuals were extreamly large. The leverage plot identified the 14th ovservation as a very abnormal one.

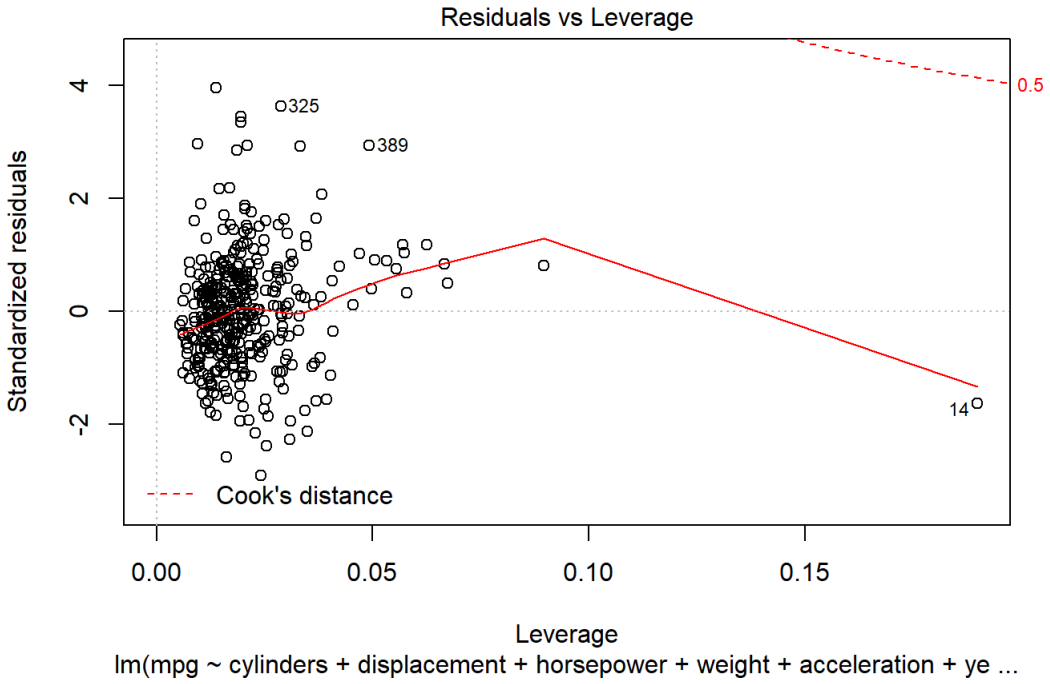
```
plot(model.c)
```


Residuals vs Fitted
Residuals
Fitted values
lm(mpg ~ cylinders + displacement + horsepower + weight + acceleration + ye ...
321
325
324


Normal Q-Q
Standardized residuals
Theoretical Quantiles
lm(mpg ~ cylinders + displacement + horsepower + weight + acceleration + ye ...
321
325
324


Scale-Location
√|Standardized residuals|
321
325
324





e. Use the * and : symbols to fit linear regression models with interaction effects. Do any interactions appear to be statistically significant?

The interaction effect of acceleration and origin was very significant. The interaction effect of displacement and year, and acceleration and year were also relatively significant.

```
model.e <- lm(mpg~cylinders+ displacement+ horsepower+ weight+ acceleration+ year+ origin+ cylinders:displacement+ cylinders:horsepower+ displacement:horsepower+ cylinders:weight+ displacement:weight+ horsepower:weight+ cylinders:acceleration+ displacement:acceleration+ horsepower:acceleration+ weight:acceleration+ cylinders:year+ displacement:year+ horsepower:year+ weight:year+ acceleration:year+ cylinders:origin+ displacement:origin+ horsepower:origin+ weight:origin+ acceleration:origin+ year:origin)
summary(model.e)
```

```
##
## Call:
## lm(formula = mpg ~ cylinders + displacement + horsepower + weight +
##     acceleration + year + origin + cylinders:displacement + cylinders:horsepower +
##     displacement:horsepower + cylinders:weight + displacement:weight +
##     horsepower:weight + cylinders:acceleration + displacement:acceleration +
##     horsepower:acceleration + weight:acceleration + cylinders:year +
##     displacement:year + horsepower:year + weight:year + acceleration:year +
##     cylinders:origin + displacement:origin + horsepower:origin +
##     weight:origin + acceleration:origin + year:origin)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.6303 -1.4481  0.0596  1.2739 11.1386
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 3.548e+01  5.314e+01   0.668  0.50475
## cylinders                   6.989e+00  8.248e+00   0.847  0.39738
## displacement               -4.785e-01  1.894e-01  -2.527  0.01192 *
## horsepower                  5.034e-01  3.470e-01   1.451  0.14769
## weight                      4.133e-03  1.759e-02   0.235  0.81442
## acceleration               -5.859e+00  2.174e+00  -2.696  0.00735 **
## year                        6.974e-01  6.097e-01   1.144  0.25340
## origin                     -2.090e+01  7.097e+00  -2.944  0.00345 **
## cylinders:displacement     -3.383e-03  6.455e-03  -0.524  0.60051
## cylinders:horsepower        1.161e-02  2.420e-02   0.480  0.63157
## displacement:horsepower    -8.491e-05  2.885e-04  -0.294  0.76867
## cylinders:weight            3.575e-04  8.955e-04   0.399  0.69000
## displacement:weight         2.472e-05  1.470e-05   1.682  0.09342 .
## horsepower:weight          -1.968e-05  2.924e-05  -0.673  0.50124
## cylinders:acceleration      2.779e-01  1.664e-01   1.670  0.09584 .
## displacement:acceleration -3.479e-03  3.342e-03  -1.041  0.29853
## horsepower:acceleration    -7.213e-03  3.719e-03  -1.939  0.05325 .
## weight:acceleration         2.346e-04  2.289e-04   1.025  0.30596
## cylinders:year             -1.741e-01  9.714e-02  -1.793  0.07389 .
## displacement:year           5.934e-03  2.391e-03   2.482  0.01352 *
## horsepower:year            -5.838e-03  3.938e-03  -1.482  0.13916
## weight:year                -2.245e-04  2.127e-04  -1.056  0.29182
## acceleration:year           5.562e-02  2.558e-02   2.174  0.03033 *
## cylinders:origin            4.022e-01  4.926e-01   0.816  0.41482
## displacement:origin         2.398e-02  1.947e-02   1.232  0.21875
## horsepower:origin           2.233e-03  2.930e-02   0.076  0.93931
## weight:origin              -5.789e-04  1.591e-03  -0.364  0.71623
## acceleration:origin         4.583e-01  1.567e-01   2.926  0.00365 **
## year:origin                 1.393e-01  7.399e-02   1.882  0.06062 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.695 on 363 degrees of freedom
## Multiple R-squared:  0.8893, Adjusted R-squared:  0.8808
## F-statistic: 104.2 on 28 and 363 DF,  p-value: < 2.2e-16
```

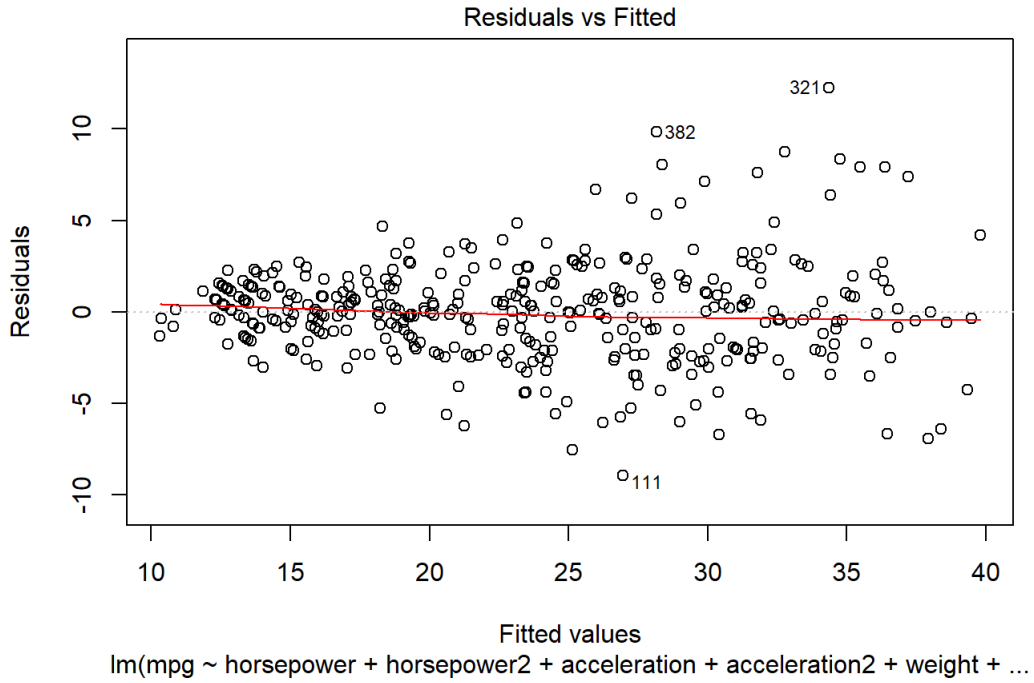
f. Try a few different transformations of the variables, such as log(X), √X, X2. Comment on your findings.

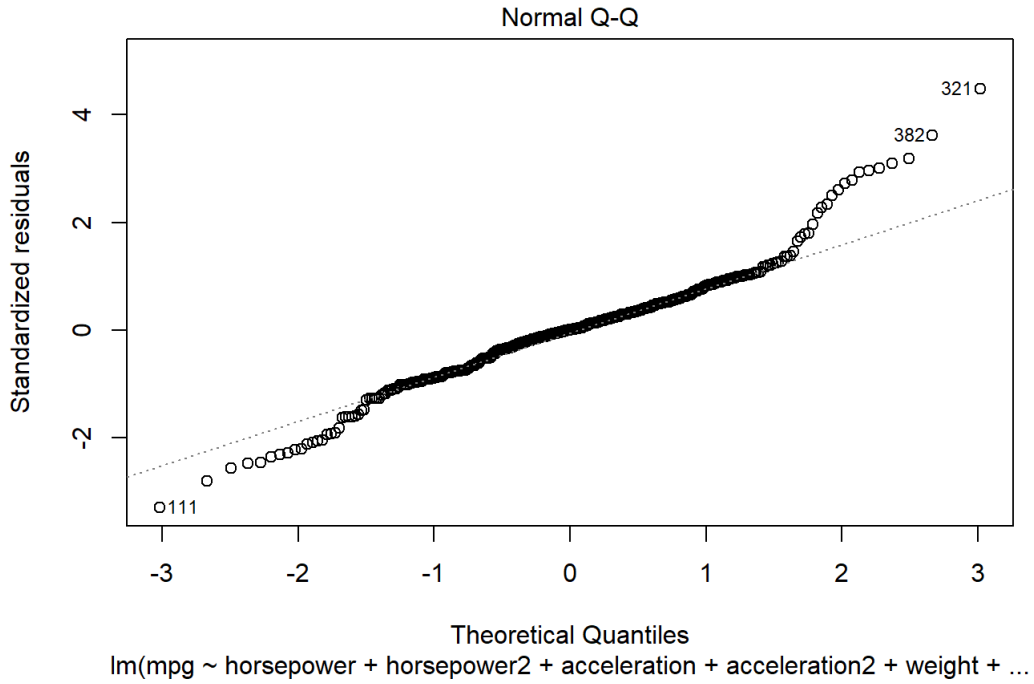
The squared horsepower, squared acceleration, squared weight and log year were significant.

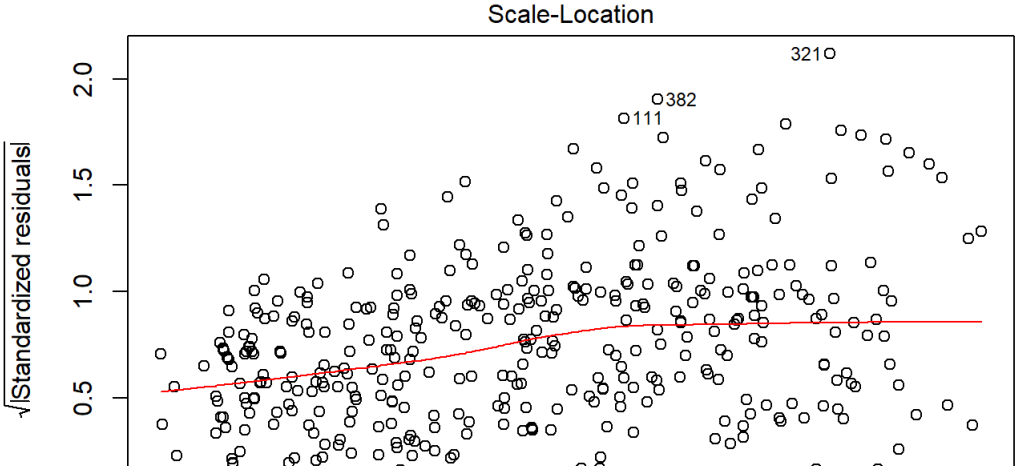
```
cylinders2 = cylinders^2
horsepower2 = horsepower^2
weight2 = weight^2
acceleration2 = acceleration^2
displacement2 = displacement^2
logyear = log(year)
model.f <- lm(mpg~cylinders+cylinders2+horsepower+horsepower2+displacement+displacement2+acceleration+acceleration2+weight+weight2
+logyear+year+origin)
model.f <- lm(mpg~horsepower+horsepower2+acceleration+acceleration2+weight+weight2+logyear+year+origin)
summary(model.f)
```

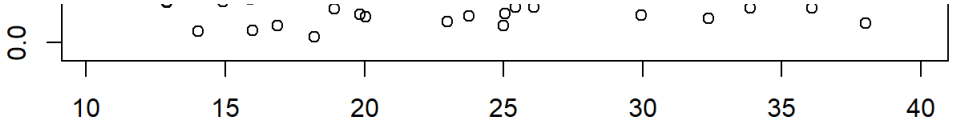
```
##
## Call:
## lm(formula = mpg ~ horsepower + horsepower2 + acceleration +
##     acceleration2 + weight + weight2 + logyear + year + origin)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.9520 -1.6226  0.0225  1.3652 12.2436
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)    2.534e+03  4.573e+02   5.542 5.59e-08 ***
## horsepower    -1.560e-01  3.751e-02  -4.159 3.95e-05 ***
## horsepower2    3.551e-04  1.221e-04   2.909 0.003842 **
## acceleration  -1.854e+00  4.929e-01  -3.761 0.000196 ***
## acceleration2  5.096e-02  1.478e-02   3.448 0.000627 ***
## weight        -1.492e-02  1.929e-03  -7.739 9.06e-14 ***
## weight2        1.741e-06  2.714e-07   6.417 4.13e-10 ***
## logyear       -7.555e+02  1.375e+02  -5.495 7.14e-08 ***
## year           1.072e+01  1.809e+00   5.928 6.87e-09 ***
## origin         7.795e-01  2.257e-01   3.453 0.000616 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.749 on 382 degrees of freedom
## Multiple R-squared:  0.8788, Adjusted R-squared:  0.876
## F-statistic: 307.8 on 9 and 382 DF,  p-value: < 2.2e-16
```

```
plot(model.f)
```

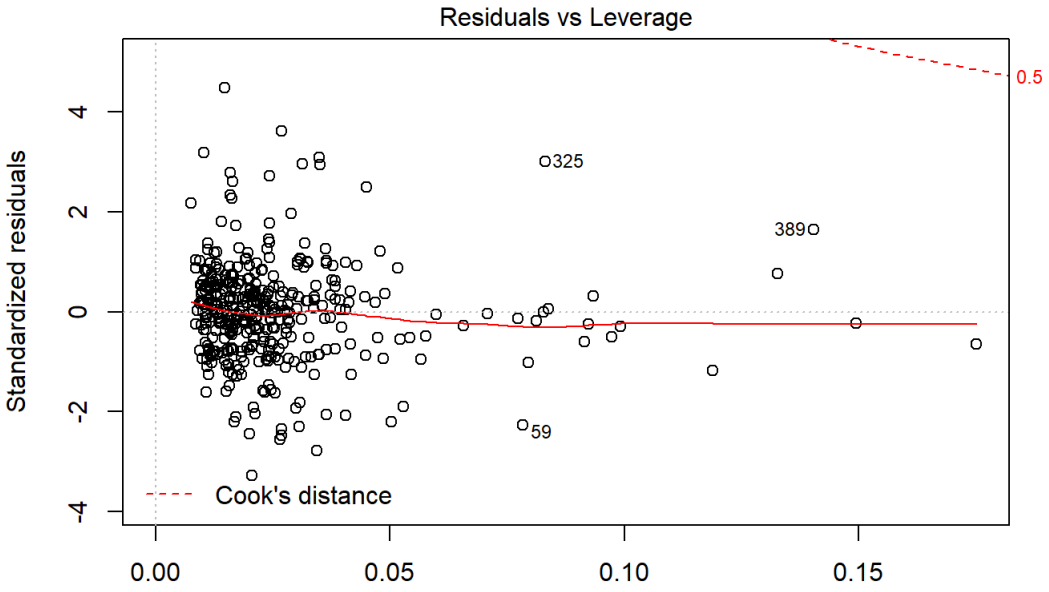

Residuals vs Fitted
Residuals
Fitted values
lm(mpg ~ horsepower + horsepower2 + acceleration + acceleration2 + weight + ...
321
382
111
10
15
20
25
30
35
40
-10
-5
0
5
10


Normal Q-Q
Standardized residuals
Theoretical Quantiles
lm(mpg ~ horsepower + horsepower2 + acceleration + acceleration2 + weight + ...
321
382
111
-3
-2
-1
0
1
2
3
-2
0
2
4


Scale-Location
√|Standardized residuals|
321
382
111
0.5
1.0
1.5
2.0



Fitted values
lm(mpg ~ horsepower + horsepower2 + acceleration + acceleration2 + weight + ...



Leverage
lm(mpg ~ horsepower + horsepower2 + acceleration + acceleration2 + weight + ...