Machine Learning | Homework 2

Liana Harutyunyan

February 23, 2019

```
library(ggplot2)
library(ISLR)
```

```
## Warning: package 'ISLR' was built under R version 3.5.2
```

```
library(GGally)
```

```
## Warning: package 'GGally' was built under R version 3.5.2
```

```
library(ggcorrplot)
```

```
## Warning: package 'ggcorrplot' was built under R version 3.5.2
```

```
library(dplyr)
library(car)
```

```
## Warning: package 'car' was built under R version 3.5.2
```

```
## Warning: package 'carData' was built under R version 3.5.2
```

Problem 1: Use set.seed(1) in random functions:

a) Create a random vector X containing 100 observations drawn from a N(0, 1) distribution (score=3).

```
X <- rnorm(100, mean = 0, sd = 1)
```

b) Create a vector e containing 100 observations drawn from a (0, 0.25) distribution (score=3).

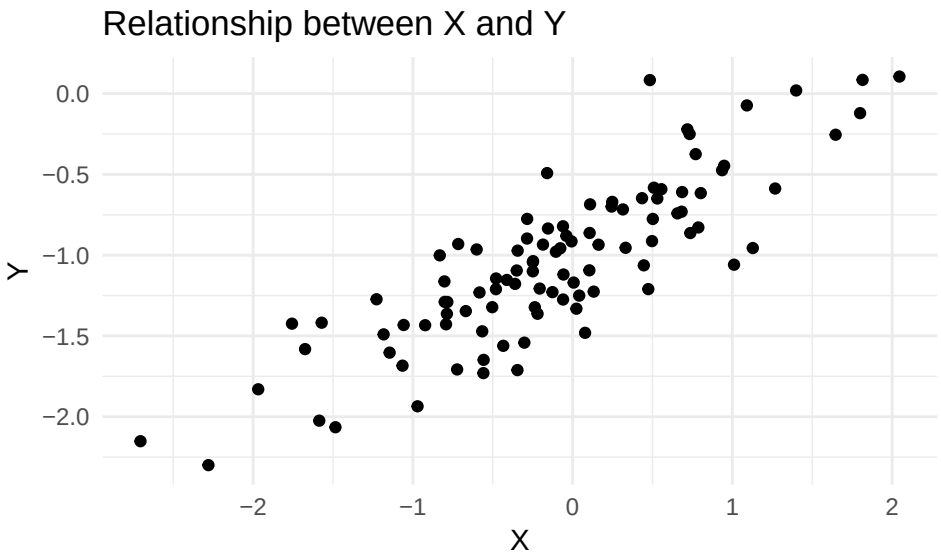
```
e <- rnorm(100, mean = 0, sd = 0.25)
```

c) Generate a vector Y according to the model $Y = -1 + 0.5X + e$ (score=3).

```
Y <- -1 + 0.5*X + e
```

d) Create a scatterplot displaying the relationship between X and Y. Comment on what you observe (score=3).

```
ggplot() +
  geom_point(aes(x = X, y = Y)) +
  theme_minimal() +
  labs(title = "Relationship between X and Y")
```



As it can be seen from the graph, there is a relationship between X and Y, which is linear and moreover their correlation is positive.

e) Fit a least squares linear model to predict Y using X. Use summary(x) and comment on the model obtained. How do $\hat{\beta}_0$ and $\hat{\beta}_1$ compare to β_0 and β_1 (score=3)?

```
model_1 <- lm(Y ~ X)
summary(model_1)
```

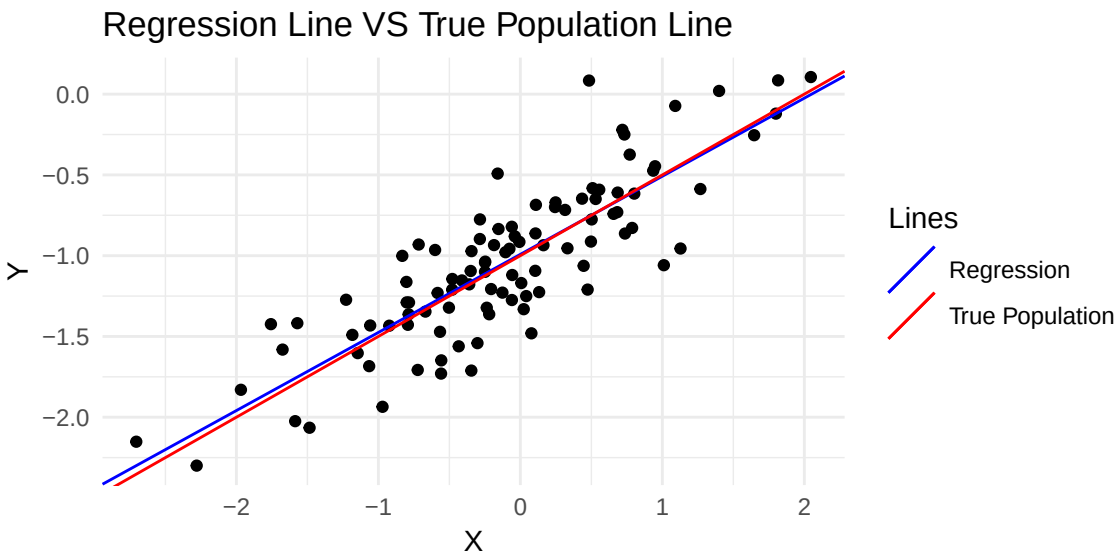
```
##
## Call:
## lm(formula = Y ~ X)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.55560 -0.18708  0.04018  0.15208  0.84201
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -0.99180    0.02675  -37.08   <2e-16 ***
## X            0.48346    0.03055   15.82   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2652 on 98 degrees of freedom
## Multiple R-squared:  0.7187, Adjusted R-squared:  0.7159
## F-statistic: 250.4 on 1 and 98 DF,  p-value: < 2.2e-16
```

As we can see as expected their relationship is really strong. Also $\beta_0 = -1$ and $\hat{\beta}_0 = -1.00807$, so they are almost the same, and same about $\beta_1 = 0.5$ and $\hat{\beta}_1 = 0.51220$.

f) Display the least squares line on the scatterplot obtained in d). Draw the population regression line on the plot, in a different color. Use the legend() command to create an appropriate legend (score=4).

```
ggplot() +
  geom_point(aes(x = X, y = Y)) +
  geom_abline(aes(intercept = model_1$coefficients[1], slope = model_1$coefficients[2], color = "Regress
```

```
  geom_abline(aes(intercept = -1, slope = 0.5, color = "True Population")) +
  scale_color_manual(values = c("blue", "red")) +
  labs(title = "Regression Line VS True Population Line", colour = "Lines") +
  theme_minimal()
```



g) Fit a polynomial regression model that predicts Y using X and X^2. Is there evidence that the quadratic term improves the model fit? Explain your answer (score=6).

```
model_1_2 <- lm(Y ~ poly(X, 2))
summary(model_1_2)
```

```
##
## Call:
## lm(formula = Y ~ poly(X, 2))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.56967 -0.18156  0.02808  0.15075  0.84884
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.04690    0.02654 -39.451   <2e-16 ***
## poly(X, 2)1  4.19705    0.26537  15.816   <2e-16 ***
## poly(X, 2)2  0.25098    0.26537   0.946    0.347
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2654 on 97 degrees of freedom
## Multiple R-squared:  0.7213, Adjusted R-squared:  0.7155
## F-statistic: 125.5 on 2 and 97 DF,  p-value: < 2.2e-16
```

Now we can compare the model statistics: RSE, R^2. The model with quadratic term has RSE of 0.2342, while the model with only linear term has RSE of 0.2373. Moreover, both adjusted and multiple R^2 also decreased a little in the model with quadratic term.

h) What are the confidence intervals for β_0 and β_1 based on the original data set, the noisier data set, and the less noisy data set? Comment on your results (score=5).

```r
e_much_noise <- rnorm(100, mean = 0, sd = 0.5)
e_less_noise <- rnorm(100, mean = 0, sd = 0.125)

Y_much_noise <- -1 + 0.5*X + e_much_noise
Y_less_noise <- -1 + 0.5*X + e_less_noise

model_1_much_noise <- lm(Y_much_noise ~ X)
model_1_less_noise <- lm(Y_less_noise ~ X)
```

So for the original data, where the error term was from N(0, 0.25), the confidence intervals for coefficients are:

```r
confint(model_1)
```

```
##                 2.5 %     97.5 %
## (Intercept) -1.0448845 -0.9387148
## X            0.4228317  0.5440887
```

For the data with much noise (N(0, 0.5)):

```r
confint(model_1_much_noise)
```

```
##                 2.5 %     97.5 %
## (Intercept) -1.0785149 -0.8852107
## X            0.4041576  0.6249315
```

And for the data with less noise from N(0, 0.125):

```r
confint(model_1_less_noise)
```

```
##                 2.5 %     97.5 %
## (Intercept) -1.0414329 -0.9949293
## X            0.4758742  0.5289863
```

As we can see, the lesser the noise, the smaller is the confidence interval for coefficients.

Problem 2: Use "Auto" data set ("ISLR" package).

First lets read the data:

```r
data("Auto")
```

a) Perform a simple linear regression with "mpg" as the response and "horsepower" as the predictor (score=2). Use the summary() function to print the results. Explain the output:

```r
model_2 <- lm(mpg ~ horsepower, data=Auto)
summary(model_2)
```

```
##
## Call:
## lm(formula = mpg ~ horsepower, data = Auto)
##
## 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
```

I. Is there a relationship between the predictor and the response (score=2)? As we can see the p-value for horsepower is very small which gives strong evidence that our predictor and response, mpg and horsepower respectively have a relationship.

II. How strong is the relationship between the predictor and the response (score=2)? We can answer to this question using residual error relative to response and R^2 statistic.
To calculate the first one, we need to calculate mean of response:

```
mean(Auto$mpg)
```

```
## [1] 23.44592
```

And we have the RSE from model summary: 4.906. So RSE relative to the response variable will be 0.2092475, so about 20%:

```
4.906 / 23.44592
```

```
## [1] 0.2092475
```

What about R^2, it is approximately 0.6 which says that about 60% of variance in MPG variable can be explained using the Horsepower.

III. What is the predicted "mpg" associated with a "horsepower" of 98 (score=2)?

```
predict(model_2, newdata = data.frame(horsepower = 98))
```

```
##        1
## 24.46708
```

IV. What are the associated 95% confidence and prediction intervals (score=2)?
By default the function computes 95% prediction and cinfidence intervals, so we do not have to specify that.

```
predict(model_2, newdata = data.frame(horsepower = 98),
      interval = "confidence")
```

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

So the Confidence interval is [23.97308; 24, 96108].

```
predict(model_2, newdata = data.frame(horsepower = 98),
      interval = "prediction")
```

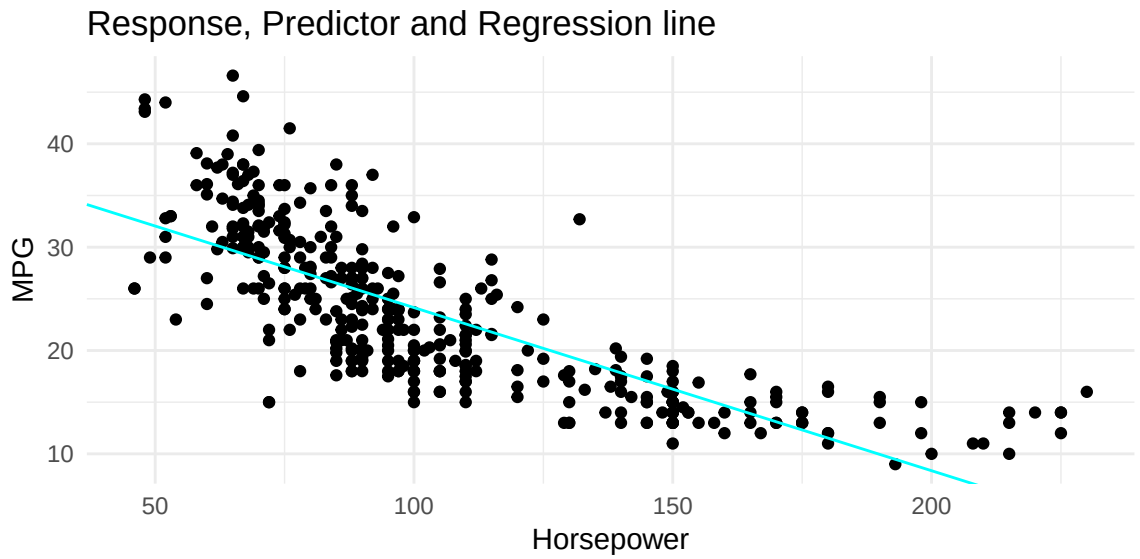
```
##        fit     lwr      upr
## 1 24.46708 14.8094 34.12476
```

And Prediction interval is [14.8094; 34.12476].

b) Plot the response and the predictor. Use the abline() function to display the least squares regression line (score=2).

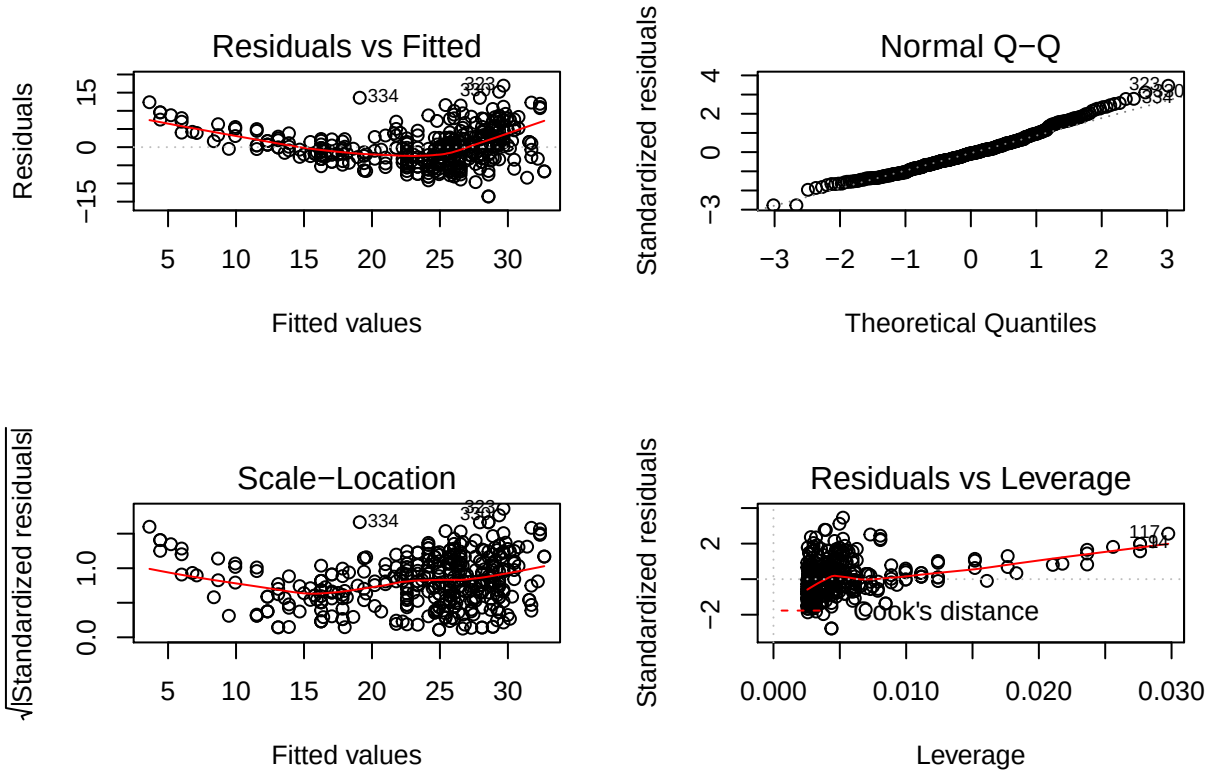
```
ggplot(Auto, aes(x = horsepower, y = mpg)) +
  geom_point() +
  geom_abline(intercept = model_2$coefficients[1],
```

```
              slope = model_2$coefficients[2],
              color = "cyan") +
  theme_minimal() +
  labs(x = "Horsepower", y = "MPG",
       title = "Response, Predictor and Regression line")
```



c) Produce diagnostic plots of the least squares regression fit. Comment on any problems you see with the fit (score=4).

```
par(mfrow=c(2,2))
plot(model_2)
```



The plot of *Residuals vs Fitted* has a pattern which looks like U-shape, which means non-linear relationship between the response and predictor variables.
The second *Normal Q-Q* plot showing if residuals are normally distributed. In our case they mainly are lined well on the straight dashed line, but also there are some resudials that do not follow that pattern.
From the third plot, we can see that the assumption about the constant variance of error terms is not proved.

d) Do the residual plots suggest any unusually large outliers? Does the leverage plot identify any observations with unusually high leverage (score=4)?

The last plot titled as *Residuals vs Leverage* shows the presense of a few outliers (bigger than 2 or smaller than -2) and few high leverage points (for example 117). Those points are influential cases, and omitting these observations from the data may lead to different result of regression model.

Problem 3: Use "Carseats" data set ("ISLR" package).

First lets see the structure of the data:

```
data("Carseats")
str(Carseats)
```

```
## 'data.frame':    400 obs. of  11 variables:
##  $ Sales      : num  9.5 11.22 10.06 7.4 4.15 ...
##  $ CompPrice  : num  138 111 113 117 141 124 115 136 132 132 ...
##  $ Income     : num  73 48 35 100 64 113 105 81 110 113 ...
##  $ Advertising: num  11 16 10 4 3 13 0 15 0 0 ...
##  $ Population : num  276 260 269 466 340 501 45 425 108 131 ...
```

```
##  $ Price      : num  120 83 80 97 128 72 108 120 124 124 ...
##  $ ShelveLoc  : Factor w/ 3 levels "Bad","Good","Medium": 1 2 3 3 1 1 3 2 3 3 ...
##  $ Age        : num  42 65 59 55 38 78 71 67 76 76 ...
##  $ Education  : num  17 10 12 14 13 16 15 10 10 17 ...
##  $ Urban      : Factor w/ 2 levels "No","Yes": 2 2 2 2 2 1 2 2 1 1 ...
##  $ US         : Factor w/ 2 levels "No","Yes": 2 2 2 2 1 2 1 2 1 2 ...
```

a) Fit a multiple regression model to predict "Sales" using "Price", "Urban", and "US" (score=2).

```
model_3 <- lm(Sales ~ Price + Urban + US, data = Carseats)
```

b) For which of the predictors can you reject the null hypothesis H_0: $\beta_j = 0$ (score=2)?

To answer the question above, let's see the summary of the model and p-values for our predictors.

```
summary(model_3)
```

```
##
## Call:
## lm(formula = Sales ~ Price + Urban + US, data = Carseats)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.9206 -1.6220 -0.0564  1.5786  7.0581
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 13.043469   0.651012  20.036  < 2e-16 ***
## Price       -0.054459   0.005242 -10.389  < 2e-16 ***
## UrbanYes    -0.021916   0.271650  -0.081    0.936
## USYes        1.200573   0.259042   4.635 4.86e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.472 on 396 degrees of freedom
## Multiple R-squared:  0.2393, Adjusted R-squared:  0.2335
## F-statistic: 41.52 on 3 and 396 DF,  p-value: < 2.2e-16
```

As we can see the p-values for Price and US are small enough, so we reject the null hypothesis for these two variables.

c) On the basis of your response to the previous question, fit a smaller model that only uses the predictors for which there is evidence of association with the outcome (score=4).?

```
model_3_1 <- lm(Sales ~ Price + US, data = Carseats)
```

d) How well do the models in a) and c) fit the data (score=2)?

For comparison do the summary of the second model, and compare model fitting statistics.

```
summary(model_3_1)
```

```
##
## Call:
## lm(formula = Sales ~ Price + US, data = Carseats)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
```

```
## -6.9269 -1.6286 -0.0574  1.5766  7.0515
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 13.03079    0.63098  20.652  < 2e-16 ***
## Price       -0.05448    0.00523 -10.416  < 2e-16 ***
## USYes        1.19964    0.25846   4.641 4.71e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.469 on 397 degrees of freedom
## Multiple R-squared:  0.2393, Adjusted R-squared:  0.2354
## F-statistic: 62.43 on 2 and 397 DF,  p-value: < 2.2e-16
```

As we can see the RSE of the first fitted model was 2.472, and one from this model is 2.469. So we have a small but still important improvement in the model.

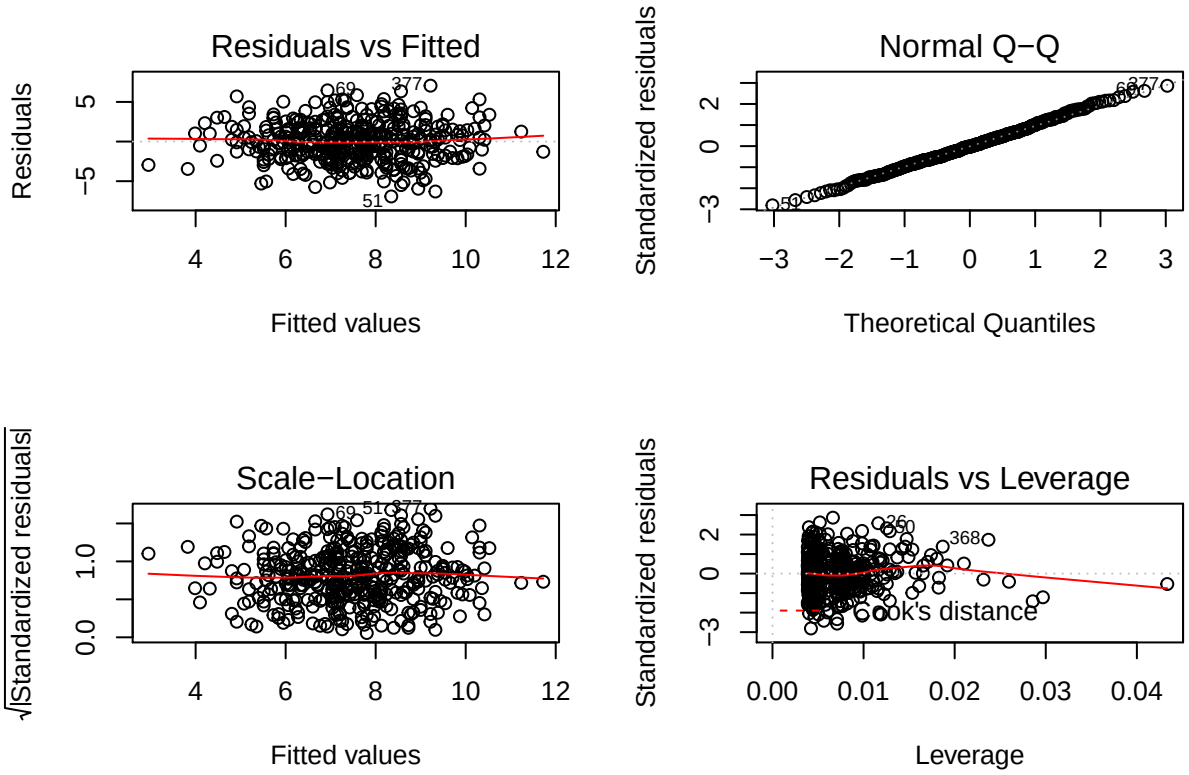
e) Using the model from c), obtain 95% confidence intervals for the coefficient(s) (score=2).

```
confint(model_3_1)
```

```
##                   2.5 %      97.5 %
## (Intercept) 11.79032020 14.27126531
## Price       -0.06475984 -0.04419543
## USYes        0.69151957  1.70776632
```

f) Produce diagnostic plots of the linear regression fit. Comment on any problems you see with the fit (score=4).

```
par(mfrow=c(2,2))
plot(model_3_1)
```



As we can see from the first graph the relationship between predictors and response is linear and the fit is quite good.
The *Normal Q-Q* graph also shows quite good results. The resudials are lined properly. So they are normally distributed.
The *Scale Location* graph shows that residuals are spreaded equally on the range of predictors, as the line is horizontal, and points are distributed equally.
So, the fit was quite a good one.

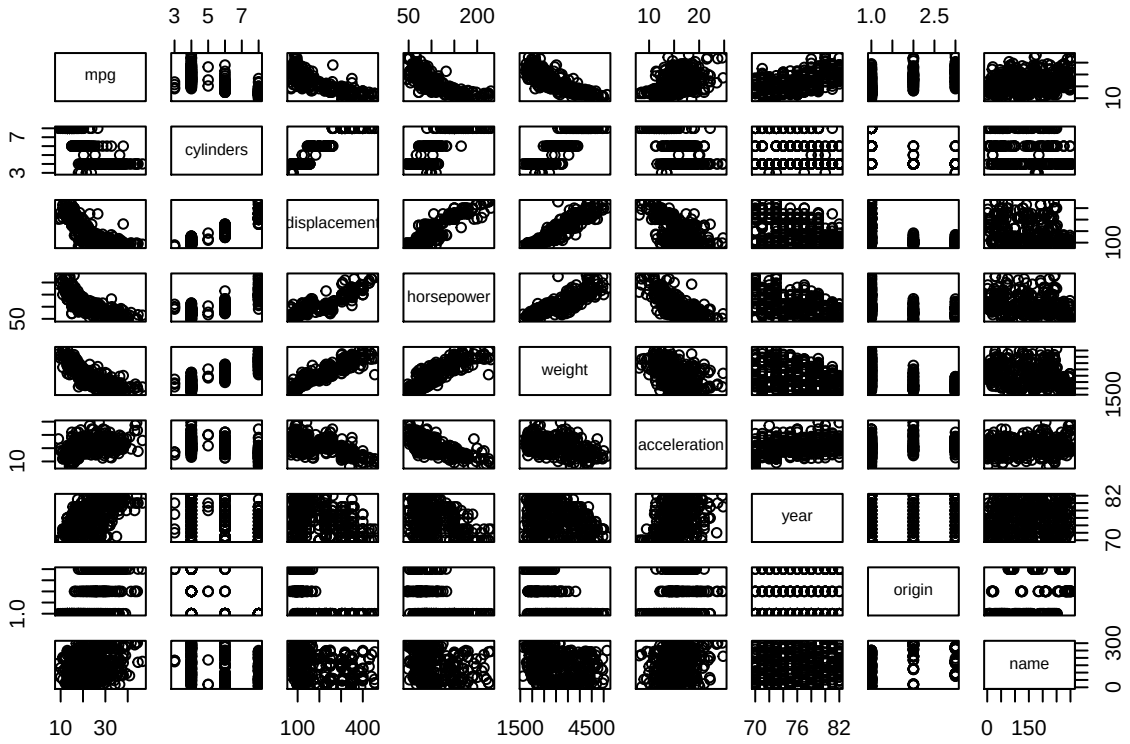
g) Do the residual plots suggest any unusually large outliers? Does the leverage plot identify any observations with unusually high leverage (score=4)?

From the last graph we can see that there is not much evidence for leverage points in the data (although 368 can be considered as one). But we can see certainly several outliers (near 3 and -3).

Problem 4: Use "Auto" data set ("ISLR" package).

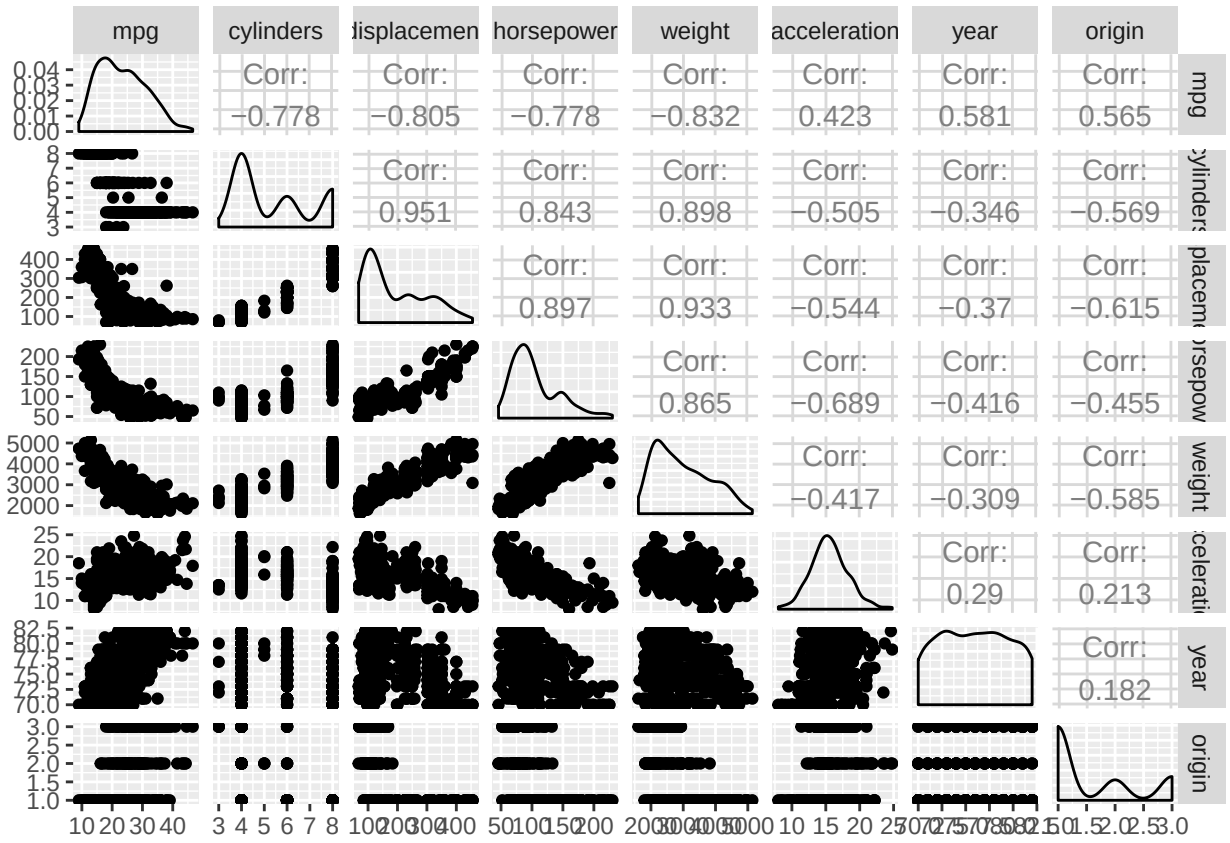
a) Produce a scatterplot matrix which includes all of the variables in the data set (score=2).

```
pairs(Auto)
```



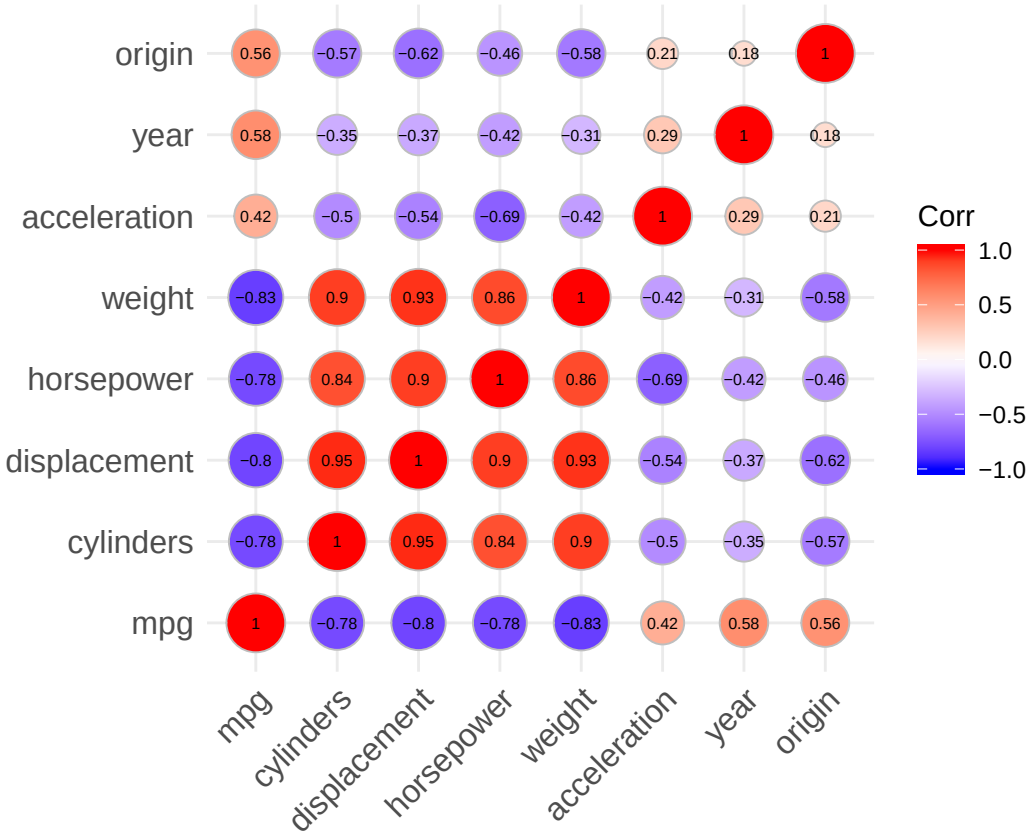
Or using the ggplot library:

```
Auto %>%
   select(-name) %>%
   ggpairs()
```



b) Compute the matrix of correlations between the variables. You will need to exclude the "name" variable, which is qualitative (score=2).

```
Auto_num <- Auto %>% select(-name)
corr <- round(cor(Auto_num), 3)
ggcorrplot(corr, method = "circle", lab = TRUE, lab_size = 2)
```



c) Perform a multiple linear regression with "mpg" as the response and all other variables except "name" as the predictors (score=2). Use the summary() function to print the results. Comment on the output:
I. Is there a relationship between the predictors and the response (score=2)? II. Which predictors appear to have a statistically significant relationship to the response (score=2)? III. What does the coefficient for the "year" variable suggest (score=2)?

```
model_4 <- lm(mpg ~ . , data = Auto_num)
summary(model_4)
```

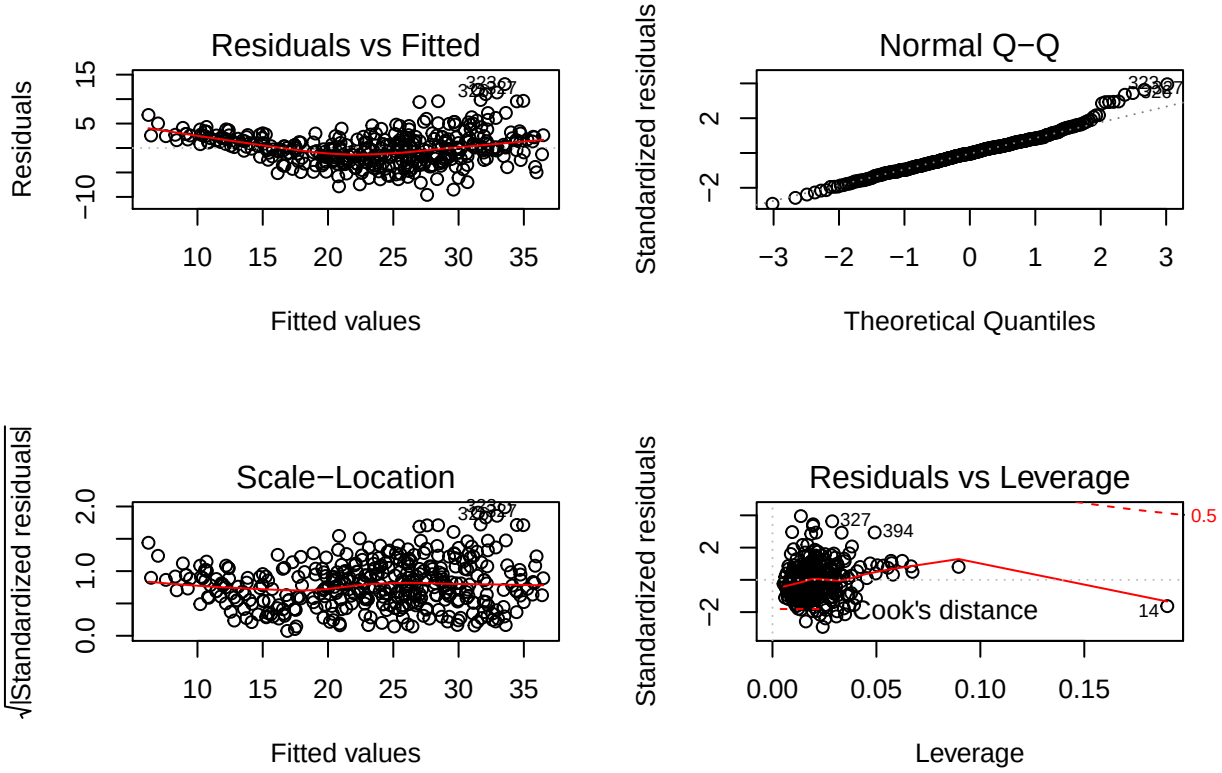
```
##
## Call:
## lm(formula = mpg ~ ., data = Auto_num)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.5903 -2.1565 -0.1169  1.8690 13.0604
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -17.218435   4.644294  -3.707  0.00024 ***
## cylinders     -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
```

I. Yes, there is a relationship between the response variable and some of the predictors.
II. As we can see from the p-values, all variables except **cylinders, horsepower and accelaration** have significant relationship with the response.
III. The coefficient for the **year** variable is 0.750773, which means that if other variables are constant, 1 year increase will increase mpg by 0.750773.

d) Produce diagnostic plots of the linear regression fit. Comment on any problems you see with the fit (score=4).

```
par(mfrow=c(2,2))
plot(model_4)
```

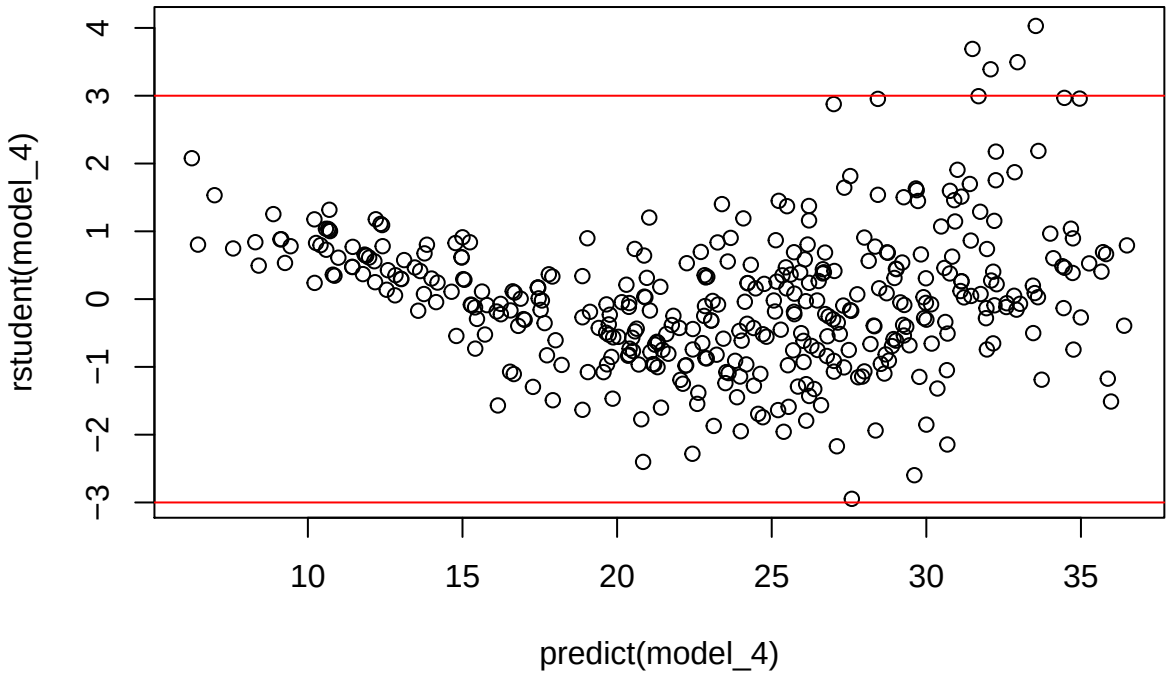


From the *Residuals vs Fitted* graph, we can see that the red line has a kind of "U-shape", which as indicated before is a sign for non-linear relationship between predictor and response variables.
The second graph shows that mainly the residualls are distributed normally.
The *Scale-Location* graph shows that the residuals are distributed equally among the range of predictors.
So the results are mainly good.

e) Do the residual plots suggest any unusually large outliers? Does the leverage plot identify any observations with unusually high leverage (score=4)?

What about the Leverages and Outliers, the plot of standardized Residuals VS Leverage shows the presence of a few outliers (for example the points 327) and a high leverage point (point 14). To make sure, we are right, for this exercise lets consider other plots as well, besides the ones above:
(keeping in mind that in our case n = 392, and p = 7)

```
plot(predict(model_4), rstudent(model_4))  # Studentized Residuals vs Fitted values
abline(h = -3, col = 'red')
abline(h = 3, col = 'red')
```



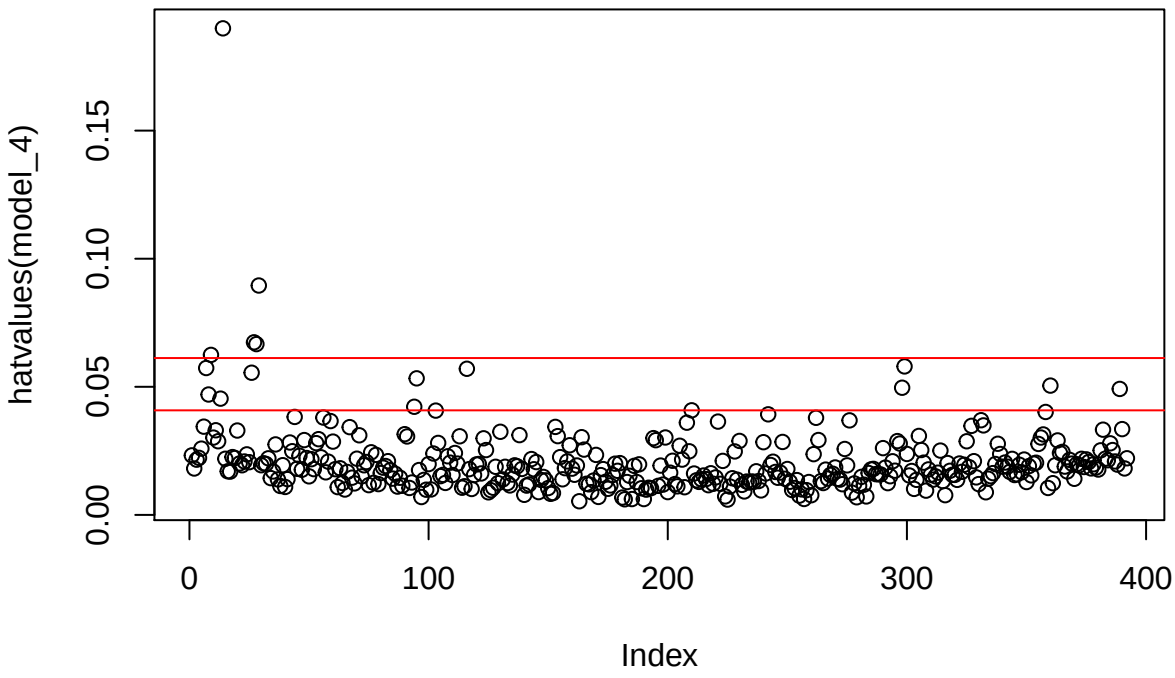

```
which(abs(rstudent(model_4))>3)           # for identifying outliers.
```

```
## 245 323 326 327
## 243 321 324 325
```

```
plot(hatvalues(model_4))
abline(h=2*8/392, col = 'red')           # Often use 2(p+1)/n or 3(p+1)/n threshold
which(hatvalues(model_4)>2*8/392)
```

```
##   7   8   9  13  14  26  27  28  29  95  96 117 212 300 301 365 394
##   7   8   9  13  14  26  27  28  29  94  95 116 210 298 299 360 389
```

```
abline(h=3*8/392, col = 'red')           # to determine high leverage points.
```




```
which(hatvalues(model_4)>3*8/392)
```

```
##  9 14 27 28 29
##  9 14 27 28 29
```

As we see we were right and we have several both leverage and outlier points.

f) Use the "*" and ":" symbols to fit linear regression models with interaction effects. Do any interactions appear to be statistically significant (score=5)?

```
model_4_2 <- lm(mpg ~ . + displacement:year, data = Auto_num)
summary(model_4_2)
```

```
##
## Call:
## lm(formula = mpg ~ . + displacement:year, data = Auto_num)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.9549 -1.9978 -0.0345  1.6232 12.2793
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)       -6.241e+01  7.811e+00  -7.990 1.60e-14 ***
## cylinders          1.200e-02  3.133e-01   0.038   0.9695
## displacement       2.710e-01  3.663e-02   7.399 8.78e-13 ***
## horsepower        -3.201e-02  1.318e-02  -2.429   0.0156 *
## weight            -5.860e-03  6.211e-04  -9.435  < 2e-16 ***
```

```
## acceleration       1.059e-01  9.328e-02   1.136   0.2568
## year               1.334e+00  9.636e-02  13.848  < 2e-16 ***
## origin             1.230e+00  2.638e-01   4.662 4.34e-06 ***
## displacement:year -3.491e-03  4.997e-04  -6.988 1.25e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.138 on 383 degrees of freedom
## Multiple R-squared:  0.8417, Adjusted R-squared:  0.8384
## F-statistic: 254.5 on 8 and 383 DF,  p-value: < 2.2e-16
```

```
model_4_3 <- lm(mpg ~.+displacement:weight + acceleration:cylinders + horsepower*year, data = Auto_num)
summary(model_4_3)
```

```
##
## Call:
## lm(formula = mpg ~ . + displacement:weight + acceleration:cylinders +
##     horsepower * year, data = Auto_num)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.7523 -1.4688 -0.0706  1.3285 11.2701
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)              -7.266e+01  1.002e+01  -7.252 2.32e-12 ***
## cylinders                 1.538e+00  6.824e-01   2.254   0.0248 *
## displacement             -6.433e-02  1.038e-02  -6.200 1.47e-09 ***
## horsepower                5.733e-01  9.166e-02   6.255 1.07e-09 ***
## weight                   -8.481e-03  7.716e-04 -10.991  < 2e-16 ***
## acceleration              3.217e-01  2.105e-01   1.529   0.1271
## year                      1.581e+00  1.242e-01  12.726  < 2e-16 ***
## origin                    5.081e-01  2.463e-01   2.063   0.0398 *
## displacement:weight       1.791e-05  2.248e-06   7.966 1.92e-14 ***
## cylinders:acceleration   -7.010e-02  4.026e-02  -1.741   0.0825 .
## horsepower:year          -8.431e-03  1.244e-03  -6.780 4.60e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.782 on 381 degrees of freedom
## Multiple R-squared:  0.8762, Adjusted R-squared:  0.873
## F-statistic: 269.7 on 10 and 381 DF,  p-value: < 2.2e-16
```

As we can see the second model is better with both R^2 values and RSE has also got decreased/improved. Moreover, all interaction terms, have small p-values (except for cylinders:accelareation), so they all are statistically significant.

f) Try a few different transformations of the variables, such as $log(x)$, $\sqrt{(x)}$, x^2. Comment on your findings (score=5).?

```
model_4_4 <- lm(mpg ~ . + log(horsepower) + log(acceleration) + log(displacement) + log(weight), data =
summary(model_4_4)
```

```
##
## Call:
## lm(formula = mpg ~ . + log(horsepower) + log(acceleration) +
```

```
##     log(displacement) + log(weight), data = Auto_num)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -9.338 -1.510 -0.096  1.441 12.365
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)       263.971424  48.763889   5.413  1.1e-07 ***
## cylinders          -0.138713   0.295410  -0.470 0.638937
## displacement       -0.002810   0.014820  -0.190 0.849714
## horsepower          0.057768   0.034135   1.692 0.091397 .
## weight              0.005097   0.002458   2.073 0.038820 *
## acceleration        1.278126   0.527141   2.425 0.015789 *
## year                0.781778   0.044993  17.376  < 2e-16 ***
## origin              0.630661   0.265868   2.372 0.018186 *
## log(horsepower)   -12.549058   3.989164  -3.146 0.001787 **
## log(acceleration) -23.788371   8.262259  -2.879 0.004213 **
## log(displacement)  -0.259132   2.916338  -0.089 0.929244
## log(weight)       -27.251957   7.887532  -3.455 0.000612 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.885 on 380 degrees of freedom
## Multiple R-squared:  0.8672, Adjusted R-squared:  0.8634
## F-statistic: 225.6 on 11 and 380 DF,  p-value: < 2.2e-16
```

```
model_4_5 <- lm(mpg ~ . + sqrt(horsepower) + sqrt(acceleration):sqrt(displacement), data = Auto_num)
summary(model_4_5)
```

```
##
## Call:
## lm(formula = mpg ~ . + sqrt(horsepower) + sqrt(acceleration):sqrt(displacement),
##     data = Auto_num)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.9708 -1.5645 -0.0645  1.5136 11.7488
##
## Coefficients:
##                                       Estimate Std. Error t value
## (Intercept)                          23.276792   8.958391   2.598
## cylinders                             0.231589   0.292036   0.793
## displacement                          0.036031   0.013491   2.671
## horsepower                            0.252979   0.064763   3.906
## weight                               -0.002811   0.000663  -4.239
## acceleration                          0.280613   0.193979   1.447
## year                                  0.759837   0.044993  16.888
## origin                                0.648652   0.259216   2.502
## sqrt(horsepower)                     -7.005198   1.403327  -4.992
## sqrt(acceleration):sqrt(displacement) -0.359996   0.098855  -3.642
##                                       Pr(>|t|)
## (Intercept)                           0.009731 **
## cylinders                             0.428261
## displacement                          0.007892 **
```

```
## horsepower                              0.000111 ***
## weight                                  2.82e-05 ***
## acceleration                            0.148825
## year                                     < 2e-16 ***
## origin                                  0.012754 *
## sqrt(horsepower)                        9.11e-07 ***
## sqrt(acceleration):sqrt(displacement) 0.000308 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.914 on 382 degrees of freedom
## Multiple R-squared:  0.8638, Adjusted R-squared:  0.8606
## F-statistic: 269.1 on 9 and 382 DF,  p-value: < 2.2e-16
```

```
model_4_6 <- lm(mpg ~ .-horsepower+ poly(horsepower,5), data = Auto_num)
summary(model_4_6)
```

```
##
## Call:
## lm(formula = mpg ~ . - horsepower + poly(horsepower, 5), data = Auto_num)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.4524 -1.7204 -0.0091  1.3752 11.7028
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)          -1.765e+01  3.801e+00  -4.642 4.75e-06 ***
## cylinders            -1.404e-01  3.397e-01  -0.413 0.679521
## displacement         -3.573e-03  7.325e-03  -0.488 0.626004
## weight               -3.589e-03  6.727e-04  -5.335 1.65e-07 ***
## acceleration         -2.607e-01  1.004e-01  -2.595 0.009817 **
## year                  7.362e-01  4.549e-02  16.185  < 2e-16 ***
## origin                8.596e-01  2.534e-01   3.392 0.000768 ***
## poly(horsepower, 5)1 -3.864e+01  1.058e+01  -3.651 0.000298 ***
## poly(horsepower, 5)2  3.311e+01  3.815e+00   8.680  < 2e-16 ***
## poly(horsepower, 5)3 -1.315e+01  3.409e+00  -3.856 0.000135 ***
## poly(horsepower, 5)4 -2.290e+00  3.116e+00  -0.735 0.462866
## poly(horsepower, 5)5  4.504e+00  3.103e+00   1.451 0.147515
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.944 on 380 degrees of freedom
## Multiple R-squared:  0.8617, Adjusted R-squared:  0.8577
## F-statistic: 215.3 on 11 and 380 DF,  p-value: < 2.2e-16
```

From these 3 models, we can see that for our data the model with logarithmic terms gave the best result (in terms of R^2 and RSE).
And lets fit a model, with combination of all different transformations.

```
model_4_7 <- lm(log(mpg) ~.+ I(horsepower^3) + I(horsepower^2) +
                  I(displacement^2) + sqrt(year) + log(acceleration),data = Auto_num)
summary(model_4_7)
```

```
##
## Call:
```

```
## lm(formula = log(mpg) ~ . + I(horsepower^3) + I(horsepower^2) +
##     I(displacement^2) + sqrt(year) + log(acceleration), data = Auto_num)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.40735 -0.06383  0.00436  0.06143  0.36434
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)         5.251e+01  1.090e+01   4.820 2.09e-06 ***
## cylinders           1.427e-02  1.317e-02   1.084   0.2791
## displacement       -2.573e-03  6.521e-04  -3.945 9.51e-05 ***
## horsepower         -3.993e-03  3.782e-03  -1.056   0.2918
## weight             -1.396e-04  2.568e-05  -5.434 9.90e-08 ***
## acceleration        2.655e-02  1.765e-02   1.505   0.1333
## year                6.811e-01  1.442e-01   4.724 3.26e-06 ***
## origin              1.588e-02  9.970e-03   1.593   0.1121
## I(horsepower^3)     2.709e-08  7.291e-08   0.372   0.7105
## I(horsepower^2)    -3.337e-06  2.993e-05  -0.112   0.9113
## I(displacement^2)   4.428e-06  1.126e-06   3.932   0.0001 ***
## sqrt(year)         -1.136e+01  2.515e+00  -4.518 8.36e-06 ***
## log(acceleration) -5.751e-01  2.836e-01  -2.028   0.0433 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1085 on 379 degrees of freedom
## Multiple R-squared:  0.9014, Adjusted R-squared:  0.8983
## F-statistic: 288.7 on 12 and 379 DF,  p-value: < 2.2e-16
```

As we can see the logarithmic transformation on the response variable and other transformations on the predictors, significantly imporve the fitting of the model. R^2 is improved from 0.86 to 0.9, and RSE is decreased as well.
As a conclusion, we can say that as observed from plots the relationship between predictors and response were not linear, and therefore non-linear transformations helped to improve the model, as expected.