Touray_Assignment2

Sheikh-Sedat Touray

2023-09-16

Auto Dataset

In this problem before we begin we must convert origin(1 = American, 2 = European, 3 = Japanese) to the factor (categorical) Variable.

So for npw we install the **ISLR** package

```
library(ISLR)
?Auto
library(plyr)
Auto$origin <- as.factor(Auto$origin)
Auto$origin <- revalue(Auto$origin, c('1' = 'American', '2' = 'European', '3' = 'Japanese'))
head(Auto)
```

```
##   mpg cylinders displacement horsepower weight acceleration year   origin
## 1  18         8          307        130   3504         12.0   70 American
## 2  15         8          350        165   3693         11.5   70 American
## 3  18         8          318        150   3436         11.0   70 American
## 4  16         8          304        150   3433         12.0   70 American
## 5  17         8          302        140   3449         10.5   70 American
## 6  15         8          429        198   4341         10.0   70 American
##                        name
## 1 chevrolet chevelle malibu
## 2         buick skylark 320
## 3        plymouth satellite
## 4             amc rebel sst
## 5               ford torino
## 6          ford galaxie 500
```

a) Now we can get the summary statitics of each variable by using the *summary* function.

```
summary(Auto)
```

```
##       mpg          cylinders      displacement     horsepower        weight    
##  Min.   : 9.00   Min.   :3.000   Min.   : 68.0   Min.   : 46.0   Min.   :1613  
##  1st Qu.:17.00   1st Qu.:4.000   1st Qu.:105.0   1st Qu.: 75.0   1st Qu.:2225  
##  Median :22.75   Median :4.000   Median :151.0   Median : 93.5   Median :2804  
##  Mean   :23.45   Mean   :5.472   Mean   :194.4   Mean   :104.5   Mean   :2978  
##  3rd Qu.:29.00   3rd Qu.:8.000   3rd Qu.:275.8   3rd Qu.:126.0   3rd Qu.:3615  
##  Max.   :46.60   Max.   :8.000   Max.   :455.0   Max.   :230.0   Max.   :5140  
##                                                                                
##   acceleration        year            origin                      name    
##  Min.   : 8.00   Min.   :70.00   American:245   amc matador       :  5  
##  1st Qu.:13.78   1st Qu.:73.00   European: 68   ford pinto        :  5  
##  Median :15.50   Median :76.00   Japanese: 79   toyota corolla    :  5  
```

```
##  Mean   :15.54   Mean   :75.98                  amc gremlin       :  4
##  3rd Qu.:17.02   3rd Qu.:79.00                  amc hornet        :  4
##  Max.   :24.80   Max.   :82.00                  chevrolet chevette:  4
##                                                 (Other)           :365
```

b) Describing the data in terms of number of row, columns and data types

```r
str(Auto)
```

```
## 'data.frame':    392 obs. of  9 variables:
##  $ mpg         : num  18 15 18 16 17 15 14 14 14 15 ...
##  $ cylinders   : num  8 8 8 8 8 8 8 8 8 8 ...
##  $ displacement: num  307 350 318 304 302 429 454 440 455 390 ...
##  $ horsepower  : num  130 165 150 150 140 198 220 215 225 190 ...
##  $ weight      : num  3504 3693 3436 3433 3449 ...
##  $ acceleration: num  12 11.5 11 12 10.5 10 9 8.5 10 8.5 ...
##  $ year        : num  70 70 70 70 70 70 70 70 70 70 ...
##  $ origin      : Factor w/ 3 levels "American","European",..: 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 
```

From the **str()** function we can observe that there are 392 rows(observations) and 9 variables(columns)

All the the variables are numerical except the *origin* which we change to a categorical data and the *name* variable is categorical type also. And the ranges of the data can be seen in the summary funtion and it is the difference between the max. and mins.

c) i) Supervised learning question I am interested in is what *Origin* car has the highest *Mpg*.

ii) Unsupervised learning question I am interested in is if we pass the unlabelled auto data in a model are we going to be accurately observe the three clusters based on the feature learning technique of the model due to it's ability to follow patterns and group objects based on their similarities and separate them based on their differences.

---

d) Providing univariate means and variances

```r
by(Auto[,1:7], Auto$origin, colMeans)
```

```
## Auto$origin: American
##          mpg    cylinders displacement   horsepower       weight acceleration
##    20.033469     6.277551   247.512245   119.048980  3372.489796    14.990204
##         year
##    75.591837
## ------------------------------------------------------------
## Auto$origin: European
##          mpg    cylinders displacement   horsepower       weight acceleration
##    27.602941     4.161765   109.632353    80.558824  2433.470588    16.794118
##         year
##    75.676471
## ------------------------------------------------------------
## Auto$origin: Japanese
##          mpg    cylinders displacement   horsepower       weight acceleration
##    30.450633     4.101266   102.708861    79.835443  2221.227848    16.172152
##         year
##    77.443038
```

```r
by(Auto[,1:4], Auto$origin, var)
```

```
## Auto$origin: American
```

```
##                     mpg  cylinders displacement horsepower
## mpg            41.478547  -8.793754    -528.8049 -193.12132
## cylinders      -8.793754   2.742322     152.1400   54.68307
## displacement -528.804920 152.140030    9677.9056 3543.26784
## horsepower   -193.121318  54.683071    3543.2678 1591.83366
## --------------------------------------------------------
## Auto$origin: European
##                    mpg  cylinders displacement horsepower
## mpg          43.298797 -0.9064530   -74.004873  -90.14047
## cylinders    -0.906453  0.2570237     7.567823    4.01273
## displacement -74.004873  7.5678227   514.982221  284.55180
## horsepower   -90.140474  4.0127305   284.551800  406.33977
## --------------------------------------------------------
## Auto$origin: Japanese
##                    mpg  cylinders displacement horsepower
## mpg          37.088685 -0.5026290   -51.581224 -73.044125
## cylinders    -0.502629  0.3485881     9.850373   4.542519
## displacement -51.581224  9.8503733   535.465433 301.079682
## horsepower   -73.044125  4.5425187   301.079682 317.523856
```

Providing Multivariate Covariance and Correlation

```
by(Auto[,1:4], Auto$origin, cor)
```

```
## Auto$origin: American
##                    mpg  cylinders displacement horsepower
## mpg           1.0000000 -0.8245240   -0.8346281 -0.7515703
## cylinders    -0.8245240  1.0000000    0.9338854  0.8276464
## displacement -0.8346281  0.9338854    1.0000000  0.9027437
## horsepower   -0.7515703  0.8276464    0.9027437  1.0000000
## --------------------------------------------------------
## Auto$origin: European
##                    mpg  cylinders displacement horsepower
## mpg           1.0000000 -0.2717195   -0.4955943 -0.6795748
## cylinders    -0.2717195  1.0000000    0.6577915  0.3926528
## displacement -0.4955943  0.6577915    1.0000000  0.6220432
## horsepower   -0.6795748  0.3926528    0.6220432  1.0000000
## --------------------------------------------------------
## Auto$origin: Japanese
##                    mpg  cylinders displacement horsepower
## mpg           1.0000000 -0.1397882   -0.3660203 -0.6730950
## cylinders    -0.1397882  1.0000000    0.7209924  0.4317698
## displacement -0.3660203  0.7209924    1.0000000  0.7301760
## horsepower   -0.6730950  0.4317698    0.7301760  1.0000000
```

```
cov(Auto$mpg,Auto$cylinders, method = 'spearman')
```

```
## [1] -9691.318
```

Before we plot the graphs i want to attach the Auto dataset.

```
attach(Auto)
cylinders <- as.factor(cylinders)
search()
```

```
##  [1] ".GlobalEnv"        "Auto"              "package:plyr"
##  [4] "package:ISLR"      "package:stats"     "package:graphics"
```

```
##  [7] "package:grDevices" "package:utils"     "package:datasets"
## [10] "package:methods"   "Autoloads"         "package:base"
```

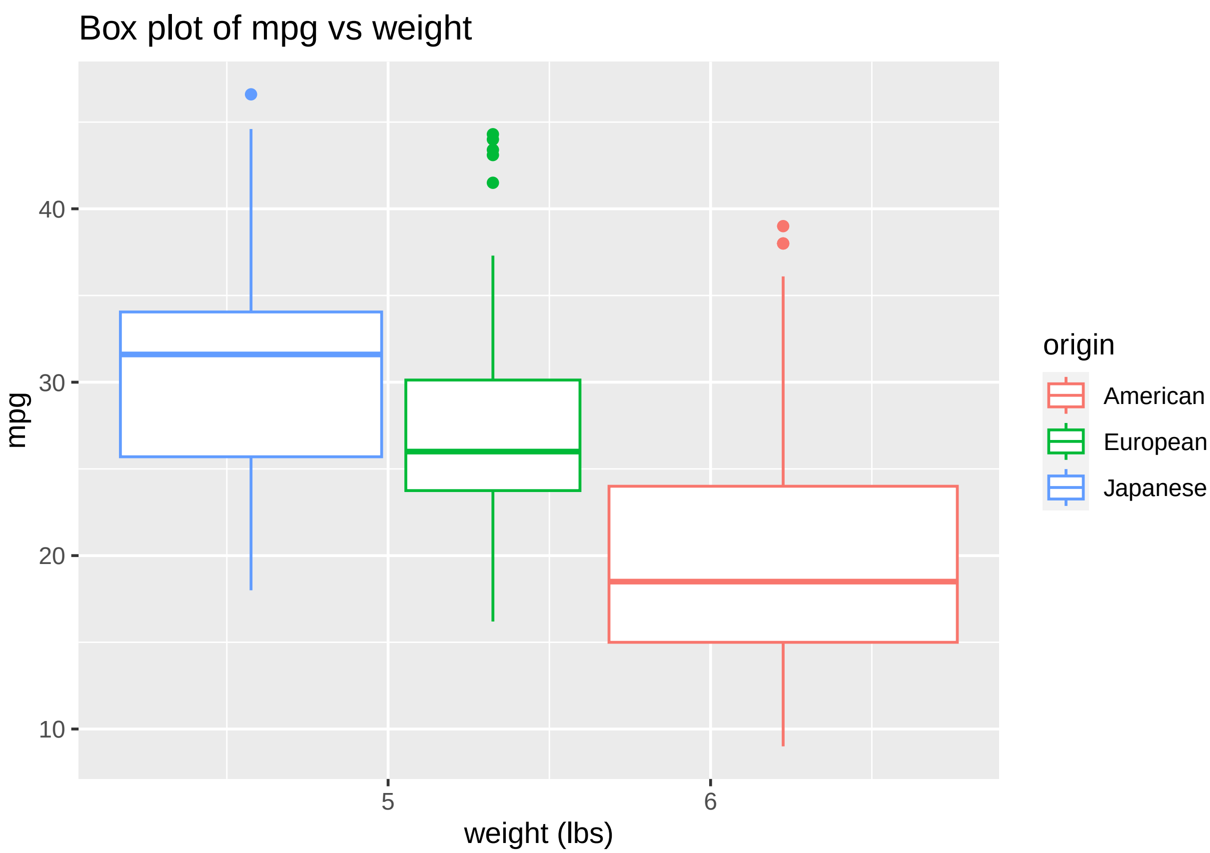
e) **Produce 3 graphical plots**

Box plot of mpg against weight

```r
#library(tidyverse)
library(ggplot2)
```

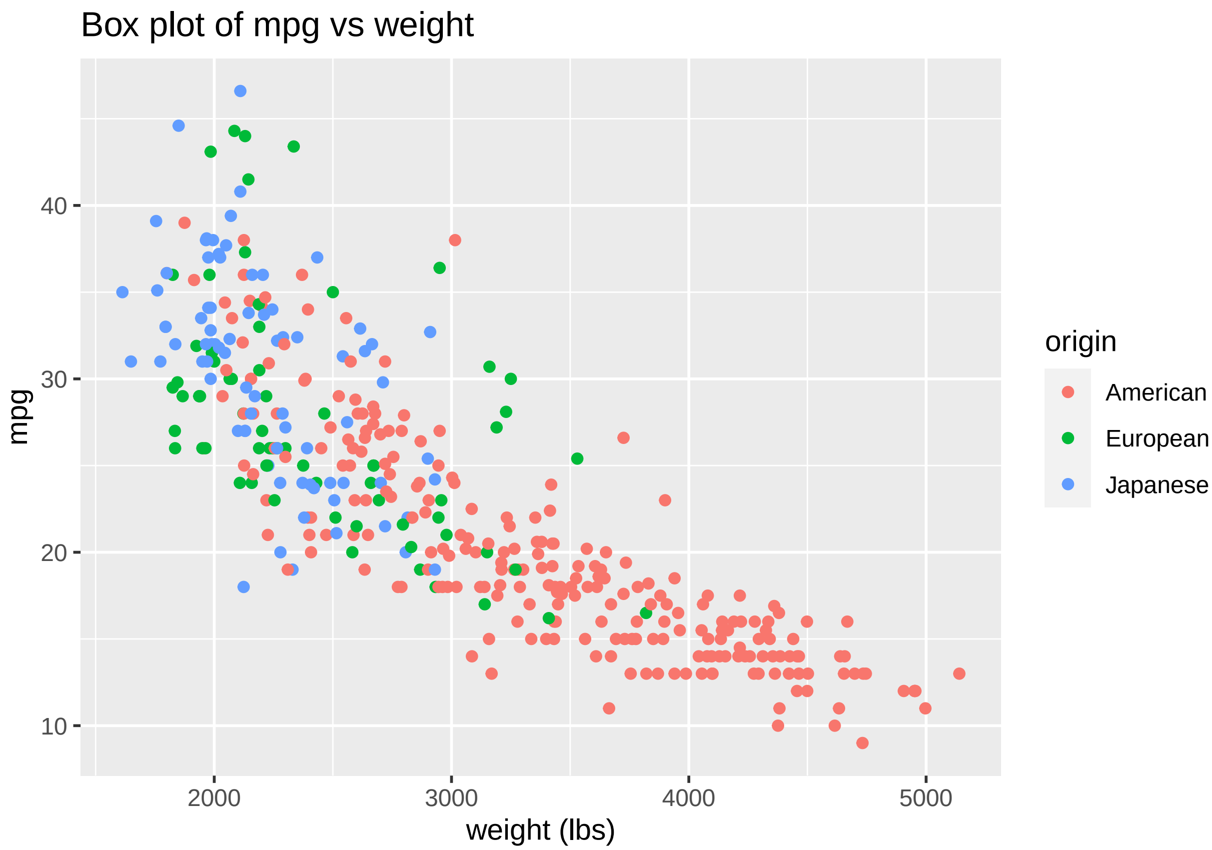
```
##
## Attaching package: 'ggplot2'

## The following object is masked from 'Auto':
##
##     mpg
```

```r
bpwm <- ggplot(Auto, aes(x=cylinders, y = mpg, color=origin)) +
       geom_boxplot()

bpwm + ggtitle("Box plot of mpg vs weight") +
  xlab("weight (lbs)") + ylab("mpg")
```



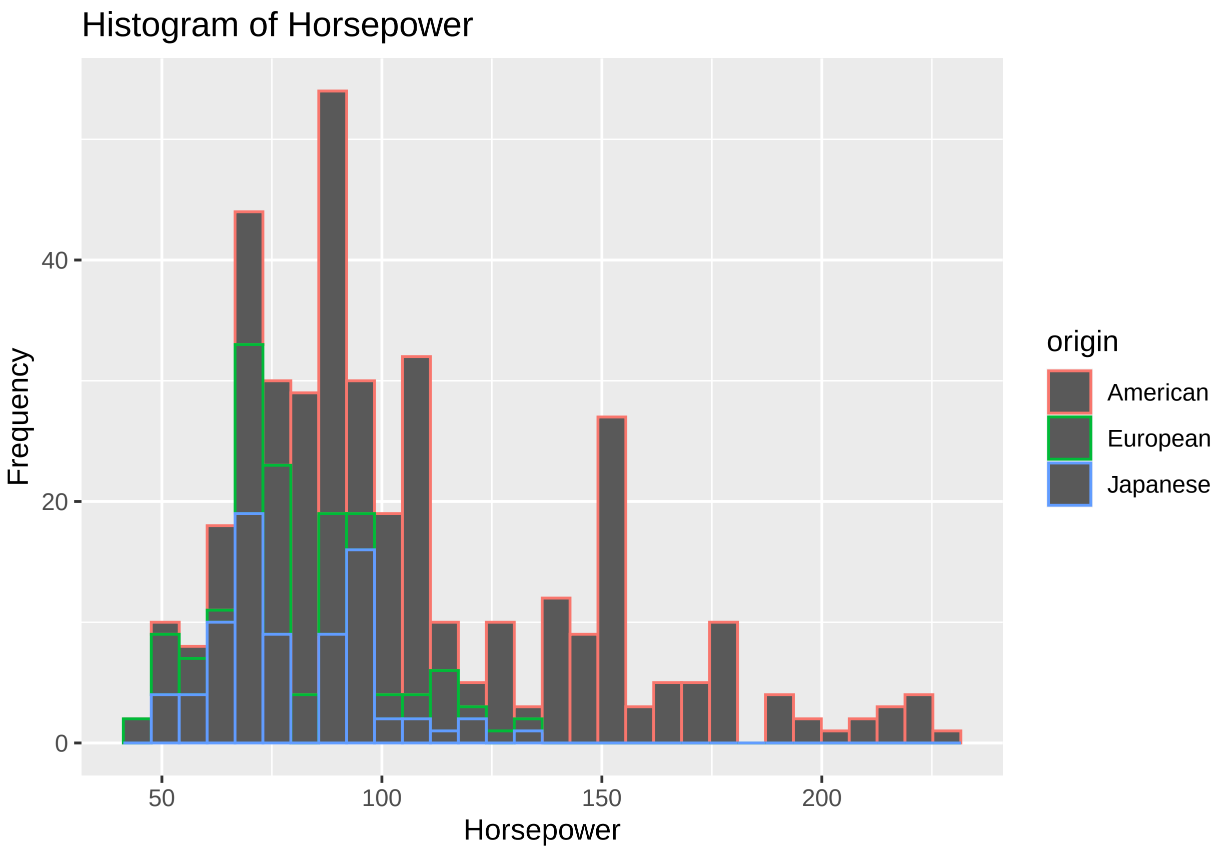
Scatter plot of mpg against weight

```r
scwm <- ggplot(Auto, aes(x=weight, y = mpg, color=origin)) +
geom_point()

scwm + ggtitle("Box plot of mpg vs weight") +
  xlab("weight (lbs)") + ylab("mpg")
```



Histogram of horsepower

```
#library(tidyverse)
library(ggplot2)
hpwr <- ggplot(Auto, aes(x=horsepower, color=origin)) +
       geom_histogram(bins=30)

hpwr + ggtitle("Histogram of Horsepower") +
  xlab("Horsepower") + ylab("Frequency")
```



f) Check univariate and multivariate normality of **horsepower**, **weight**, and **acceleration** variables.

```
shapiro.test(horsepower)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  horsepower
## W = 0.9041, p-value = 5.022e-15
```

```
shapiro.test(weight)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  weight
## W = 0.94147, p-value = 2.602e-11
```

```
shapiro.test(acceleration)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  acceleration
## W = 0.99187, p-value = 0.03053
```

```
shapiro.test(c(horsepower,weight,acceleration))
```

```
##
##  Shapiro-Wilk normality test
```

```
##
## data:  c(horsepower, weight, acceleration)
## W = 0.70565, p-value < 2.2e-16
```

For Multivariate Normaility

```r
library(mvnormtest)
multv <- t(Auto[,4:6])

mshapiro.test(multv)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Z
## W = 0.90096, p-value = 2.744e-15
```

g) Fitting a simple linear regression model with **weight** as predictor and **mpg** as response

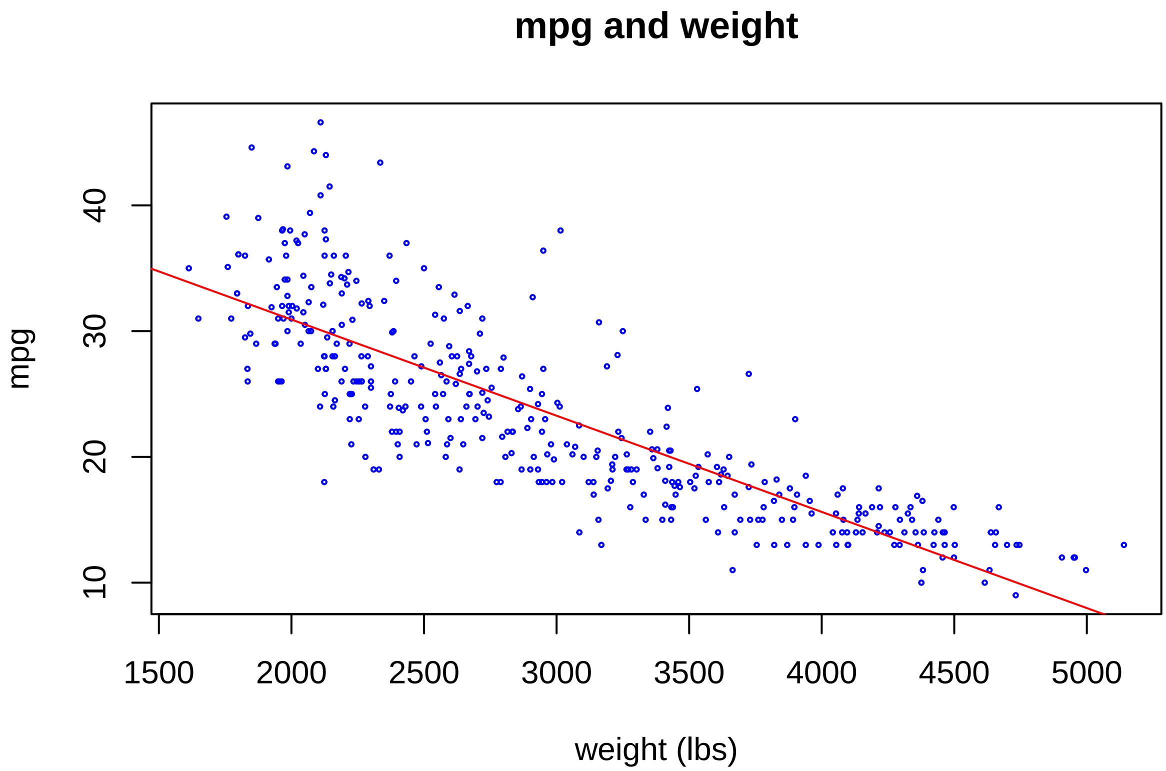
```r
model <- lm(mpg ~ weight, data = Auto)
model
```

```
##
## Call:
## lm(formula = mpg ~ weight, data = Auto)
##
## Coefficients:
## (Intercept)       weight
##   46.216525    -0.007647
```

The negative Coefficient shows that as **weight** increases the **mpg** decreases.

h) Plot **mpg** and **weight** along the regression line (on one plot)

```r
plot(mpg ~ weight, data = Auto, cex=0.3, col = "blue", main='mpg and weight',
     xlab='weight (lbs)',ylab='mpg')

abline(lm(mpg ~ weight, data = Auto), col = 'red')
```



It is clear that mpg and weight do have a negative linear relationship because as weight increases the mpg decreases and when the model was fit with a regression line this proven again as seen in the plot above.

---

i) Fitting multiple variables against **mpg** as response. without interaction.

```
model2 <- lm(mpg ~ weight + origin, data = Auto)
summary(model2)
```

```
##
## Call:
## lm(formula = mpg ~ weight + origin, data = Auto)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.1339  -2.7358  -0.3032   2.4307  15.4544
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)     43.7322362  1.1134286  39.277  < 2e-16 ***
## weight          -0.0070271  0.0003201 -21.956  < 2e-16 ***
## originEuropean   0.9709056  0.6587673   1.474 0.141340
## originJapanese   2.3271499  0.6648043   3.501 0.000518 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.277 on 388 degrees of freedom
## Multiple R-squared:  0.702,  Adjusted R-squared:  0.6997
## F-statistic: 304.7 on 3 and 388 DF,  p-value: < 2.2e-16
```

Fitting multiple variables against **mpg** as response. with interactions.

```r
model2.1 <- lm(mpg ~ weight*displacement + origin*displacement, data = Auto)
summary(model2.1)
```

```
##
## Call:
## lm(formula = mpg ~ weight * displacement + origin * displacement,
##     data = Auto)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -13.566  -2.370  -0.308   1.833  18.053
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  5.579e+01  2.851e+00  19.570  < 2e-16 ***
## weight                      -9.618e-03  1.082e-03  -8.891  < 2e-16 ***
## displacement                -8.543e-02  1.358e-02  -6.289 8.68e-10 ***
## originEuropean              -4.116e+00  3.032e+00  -1.357    0.175
## originJapanese              -2.730e+00  2.558e+00  -1.067    0.286
## weight:displacement          1.967e-05  3.452e-06   5.697 2.42e-08 ***
## displacement:originEuropean  2.973e-02  2.593e-02   1.147    0.252
## displacement:originJapanese  2.843e-02  2.212e-02   1.285    0.200
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.09 on 384 degrees of freedom
## Multiple R-squared:  0.7303, Adjusted R-squared:  0.7254
## F-statistic: 148.6 on 7 and 384 DF,  p-value: < 2.2e-16
```

Comparing these two models with and without interactions shows that the model with interactions performed better that the the one without and the interaction between weight and displacement is statistically significant while the interaction between origin and displacement is not.

h) Fitting more variables against **mpg** as response.

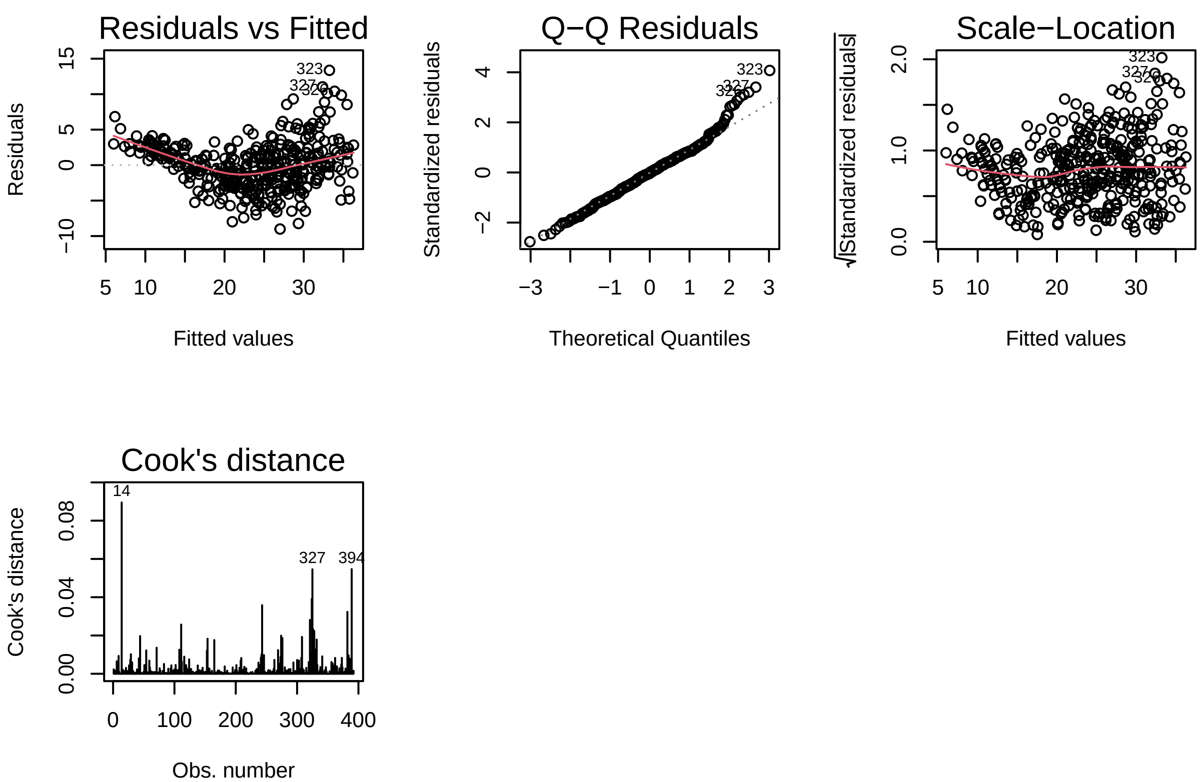
```r
model3 <- lm(mpg ~ cylinders + displacement + horsepower + weight + acceleration + year + origin, data =
summary(model3)
```

```
##
## Call:
## lm(formula = mpg ~ cylinders + displacement + horsepower + weight +
##     acceleration + year + origin, data = Auto)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.0095 -2.0785 -0.0982  1.9856 13.3608
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)    -1.795e+01  4.677e+00  -3.839 0.000145 ***
## cylinders      -4.897e-01  3.212e-01  -1.524 0.128215
## displacement    2.398e-02  7.653e-03   3.133 0.001863 **
## horsepower     -1.818e-02  1.371e-02  -1.326 0.185488
## weight         -6.710e-03  6.551e-04 -10.243  < 2e-16 ***
## acceleration    7.910e-02  9.822e-02   0.805 0.421101
```

```
## year             7.770e-01  5.178e-02  15.005  < 2e-16 ***
## originEuropean   2.630e+00  5.664e-01   4.643 4.72e-06 ***
## originJapanese   2.853e+00  5.527e-01   5.162 3.93e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.307 on 383 degrees of freedom
## Multiple R-squared:  0.8242, Adjusted R-squared:  0.8205
## F-statistic: 224.5 on 8 and 383 DF,  p-value: < 2.2e-16
```

k) Run a full diagnostic on model fit in (j) and report any issues related to model.

```r
par(mfrow=c(2,3))

Diag1 <- plot(model3, which=1:4)
```



Issues Related to Model 3

The **Residuals vs Fitted plot** is useful for checking of linearity and homoscedasticity and values not too far from 0 are the best for this purposes anything below -2 or greater that 2 could be considered problematic. So the issue with this model as we can see from this plot is that it has a high value of about 5.

By looking at the **QQ-plot** and how all the observations lie along the 45-degree line then we may assume linearity.

The **Scale - Loacation plot** is used to check for homoscedascity and we are checking checking to see if there is a pattern in the residuals and in our case, there is somewhat of a pattern which is also an issue with our model.

My cook's **distance** shows that observation 14 has a larger cook's distance than the other data points but it does not mean that this is an issue because outliers maybe or may not be influential and in this plot are not able to tell that.

So therefore **Residuals vs Fitted plot** and **Scale - Loacation plot** clearly show that model3 has an issue.

---

l) Propose a less problematic response than model in j

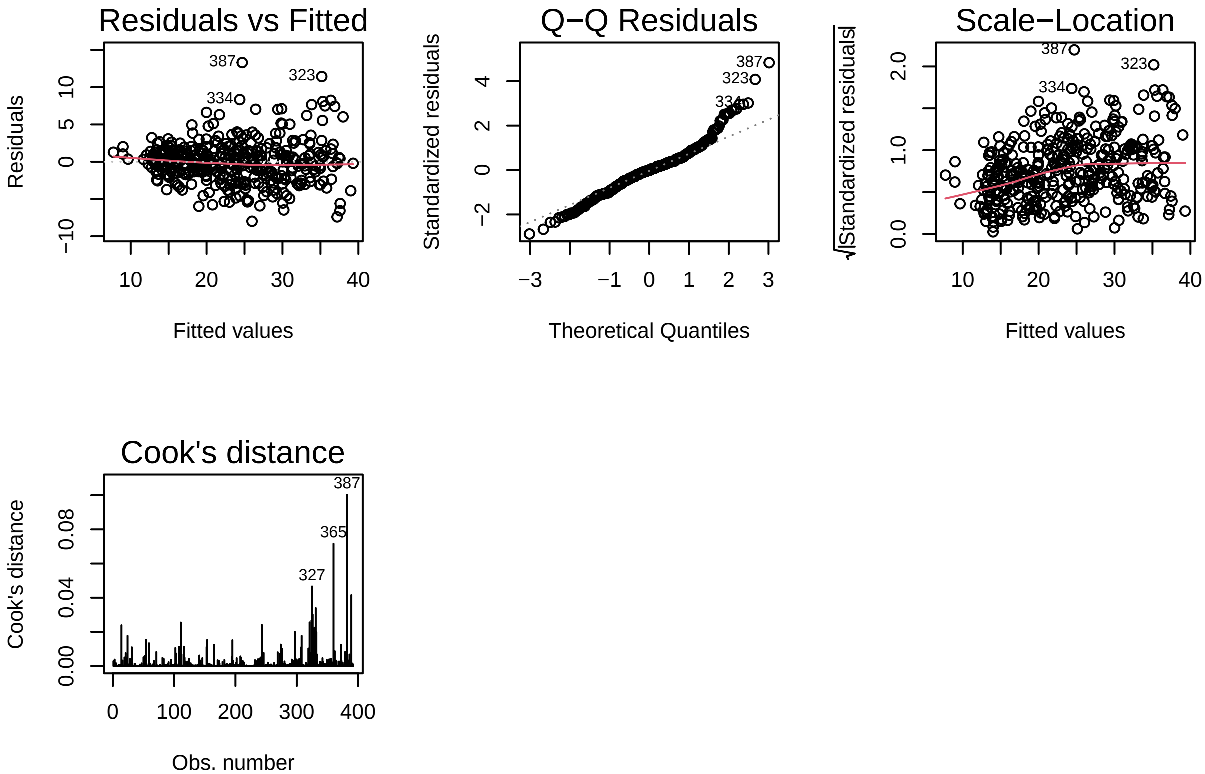
```r
model3.1 <- lm(mpg ~ cylinders + horsepower*displacement + weight*displacement + acceleration*displaceme
summary(model3.1)
```

```
##
## Call:
## lm(formula = mpg ~ cylinders + horsepower * displacement + weight *
##     displacement + acceleration * displacement + year * displacement +
##     origin, data = Auto)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -7.988 -1.552 -0.035  1.318 13.303
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)              -3.681e+01  9.045e+00  -4.070 5.73e-05 ***
## cylinders                 5.534e-01  2.977e-01   1.859 0.063783 .
## horsepower               -8.529e-02  3.257e-02  -2.619 0.009172 **
## displacement              1.072e-01  4.406e-02   2.433 0.015452 *
## weight                   -7.898e-03  1.340e-03  -5.894 8.34e-09 ***
## acceleration              8.790e-02  1.770e-01   0.497 0.619726
## year                      1.160e+00  1.022e-01  11.349  < 2e-16 ***
## originEuropean            1.336e+00  5.166e-01   2.587 0.010059 *
## originJapanese            1.048e+00  5.022e-01   2.088 0.037506 *
## horsepower:displacement   1.265e-04  1.053e-04   1.202 0.230205
## displacement:weight       1.470e-05  3.892e-06   3.776 0.000185 ***
## displacement:acceleration -5.187e-04  8.591e-04  -0.604 0.546334
## displacement:year        -2.213e-03  5.255e-04  -4.212 3.17e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.829 on 379 degrees of freedom
## Multiple R-squared:  0.8726, Adjusted R-squared:  0.8686
## F-statistic: 216.4 on 12 and 379 DF,  p-value: < 2.2e-16
```

In my proposed model it shows that all the variables that had an interaction with displacement are statistically significant and the new model seemed to be performing way better based on the R-sqquared values. However we shall explore this further when we run a diagnostic of the model

My proposed Model Diagnostic

```r
par(mfrow=c(2,3))

Diag2 <- plot(model3.1, which=1:4)
```



As anticipated my proposed model has solved the issues that were present in the previous diagnostic in the ***Residuals vs Fitted plot*** *(now at values almost 0 which we were looking for) and the* ***Scale - Loacation plot*** *(no patterns there).*