Project 3(Manjeet Singh)

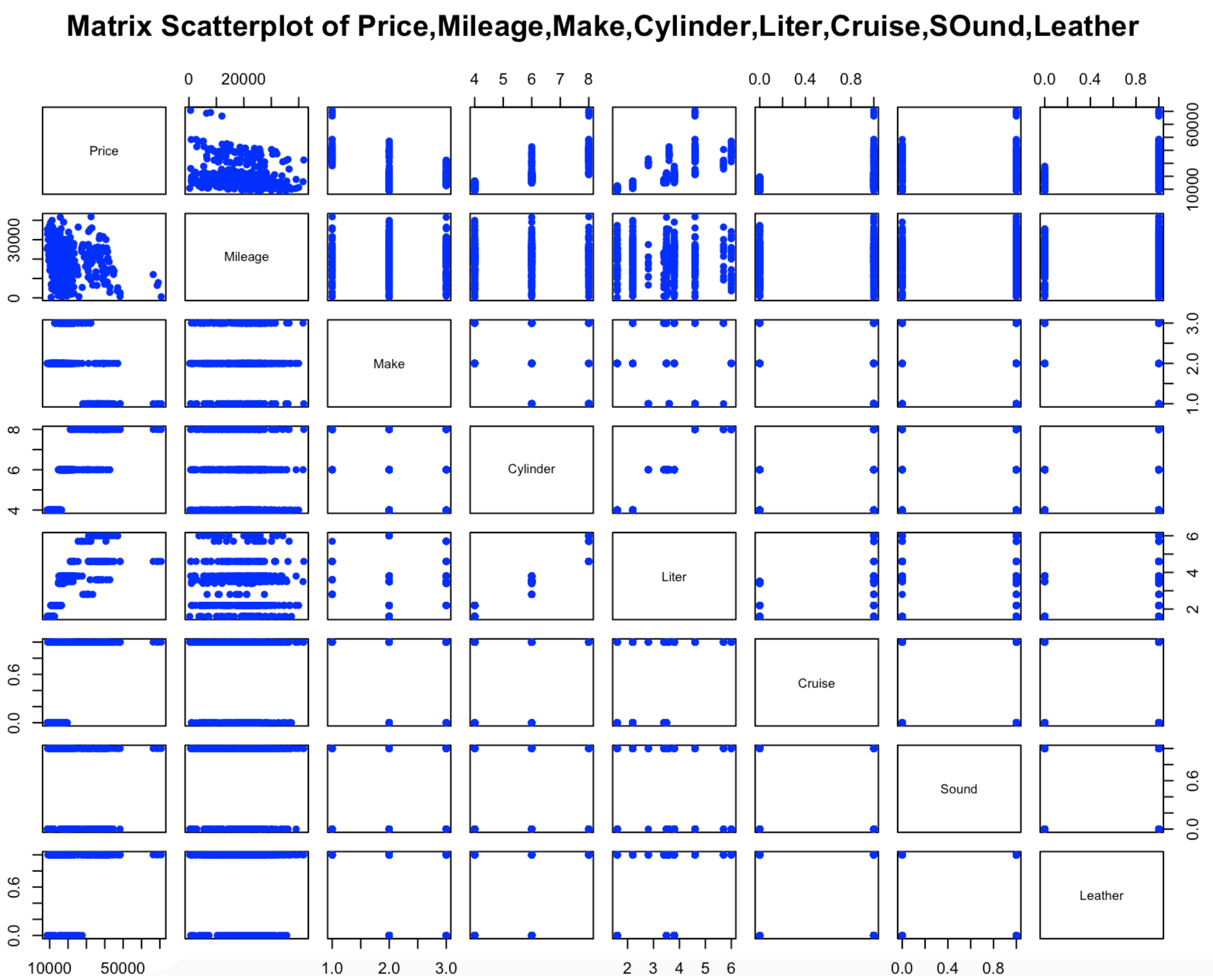
Create a Multiple Linear Regression Model for General Motors (GM) Data set

Lets read the data first.

data <- read.csv("~/Desktop/data.csv", header= T, sep =",")
colnames(data)

Before fitting our regression model we want to investigate how the variables are related to one another. We can do this graphically by constructing scatter plots of all pair-wise combinations of variables in the data frame. This can be done by typing:

plot(data, pch=16, col="blue", main="Matrix Scatterplot of Price, Mileage, Make, Cylinder, Liter, Cruise, Sound, Leather")



The matrix plot above allows us to visualise the relationship among all variables in one single image. For example, we can see how Mileage and Price are related (see first column, second row top to bottom graph).

Model no 1 - "**7 predictor model**"

To fit a multiple linear regression model with price as the response variable and mileage, make, cylinder, litre, cruise, sound and leather as the explanatory variables, use the command :-

lfit <- lm(Price ~ Mileage + Make + Cylinder + Litre + Cruise + Sound + Leather, data = data)

We can access the results of this test by typing

summary(lfit)

```
Call:
lm(formula = Price ~ Mileage + Make + Cylinder + Liter + Cruise +
    Sound + Leather, data = data)

Residuals:
    Min      1Q  Median      3Q     Max
-8420.7 -1743.5  -150.6  1315.7 26563.5

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)    2.612e+04  1.815e+03  14.392  < 2e-16 ***
Mileage       -2.058e-01  1.857e-02 -11.084  < 2e-16 ***
MakeChevrolet -1.706e+04  7.247e+02 -23.538  < 2e-16 ***
MakePontiac   -1.851e+04  7.005e+02 -26.423  < 2e-16 ***
Cylinder      -2.220e+03  5.013e+02  -4.430 1.17e-05 ***
Liter          7.691e+03  5.693e+02  13.509  < 2e-16 ***
Cruise         1.024e+02  4.007e+02   0.256    0.798
Sound          2.279e+02  3.877e+02   0.588    0.557
Leather        2.472e+02  4.198e+02   0.589    0.556
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3430 on 491 degrees of freedom
Multiple R-squared:  0.8823,    Adjusted R-squared:  0.8803
F-statistic: 459.9 on 8 and 491 DF,  p-value: < 2.2e-16
```

From the multiple regression model output above Cruise, Sound, Leather no longer displays a significant p-value. Here, Cruise, Sound, Leather represents the average effect while holding the other variables mileage, Make, Cylinder and litre constant.

Model no 2 - "**4 Predictor Model**"

lfit1 <- lm(Price ~ Mileage + Make + Cylinder + Liter , data = data)

```
Call:
lm(formula = Price ~ Mileage + Make + Cylinder + Liter, data = data)

Residuals:
    Min      1Q  Median      3Q     Max 
-8298.6 -1721.5  -111.9  1264.4 26679.7 

Coefficients:
                Estimate Std. Error t value Pr(>|t|)    
(Intercept)    2.688e+04  1.609e+03  16.703  < 2e-16 ***
Mileage       -2.061e-01  1.847e-02 -11.158  < 2e-16 ***
MakeChevrolet -1.717e+04  6.897e+02 -24.902  < 2e-16 ***
MakePontiac   -1.866e+04  6.661e+02 -28.021  < 2e-16 ***
Cylinder      -2.326e+03  4.849e+02  -4.798 2.13e-06 ***
Liter          7.811e+03  5.425e+02  14.398  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3423 on 494 degrees of freedom
Multiple R-squared:  0.882,	Adjusted R-squared:  0.8808 
F-statistic: 738.7 on 5 and 494 DF,  p-value: < 2.2e-16
```

In R we can perform partial F-tests by fitting both the models separately and thereafter comparing them using the anova function
Lets compare these two models via Anova-

```
> anova(lfit, lfit1)
Analysis of Variance Table

Model 1: Price ~ Mileage + Make + Cylinder + Liter + Cruise + Sound + 
    Leather
Model 2: Price ~ Mileage + Make + Cylinder + Liter
  Res.Df        RSS Df Sum of Sq      F Pr(>F)
1    491 5777986849                           
2    494 5788711339 -3 -10724491 0.3038 0.8227
```

The output shows the results of the partial F-test. Since F= 0.3038 (p-value=0.8227)
It appears that the variables Cruise, sound and Leather do not contribute significant information to the price once the variables Mileage, Make, Cylinder and Litre have been taken into consideration.

The model excluding Cruise, Sound and Leather has in fact improved our F-Statistic from 459.9 to 738.7 but no substantial improvement was achieved in residual standard error and adjusted R-square value. This is possibly due to the presence of outlier points in the data.

Multiple R-squared is 0.882 and Adjusted R Square is 0.8808. This value is quite good. Both of these models are predicting 88% accuracy.

We often use our regression models to estimate the mean response or predict future values of the response variable for certain values of the response variables. The function predict() can be used to make both confidence intervals for the mean response and prediction intervals. To make confidence intervals for the mean response use the option interval=”confidence”. To make a prediction interval use the option interval=”prediction”. By default this makes 95% confidence and prediction intervals. If you instead want to make a 99% confidence or prediction interval use the option level=0.99.

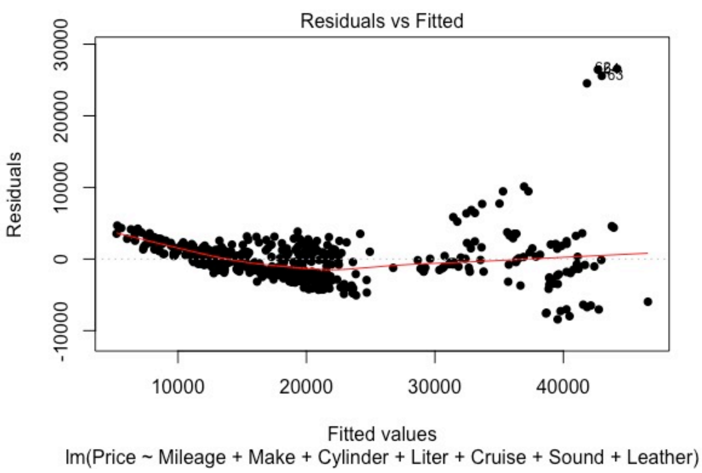
Predicted values are obtained using the function predict()

Obtain the confidence bands.
predict(lfit, interval =“confidence")

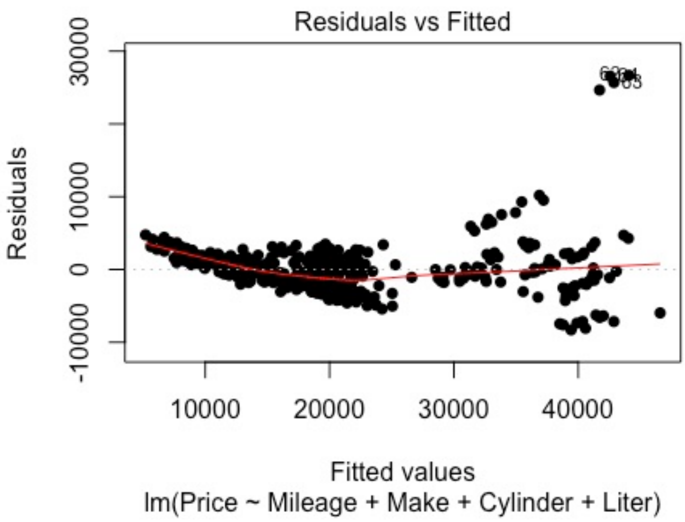
Obtain the prediction bands
predict(lfit, interval =“prediction")

Let’s plot the model’s residuals for both models.

Model 1



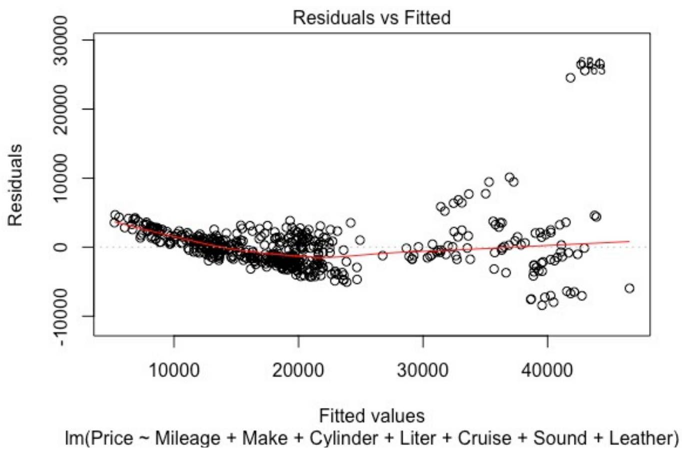
Model 2

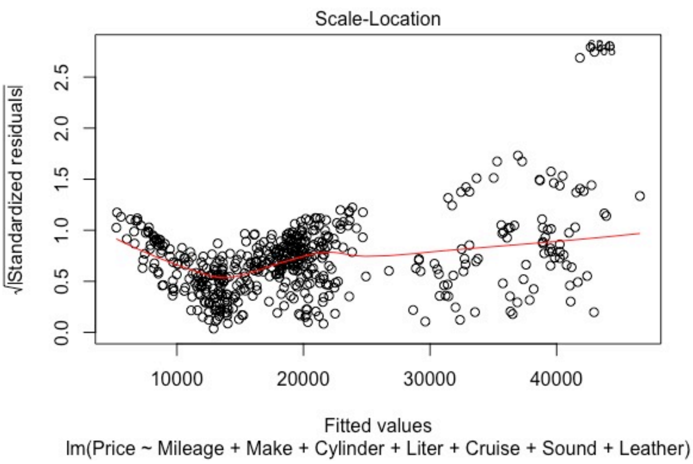


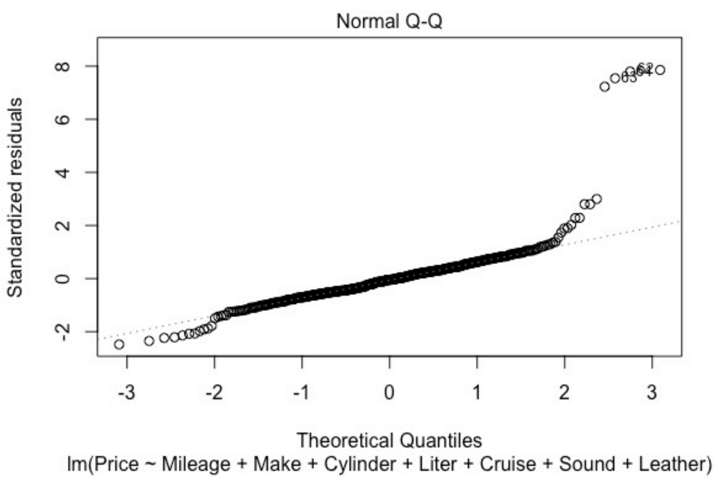
Note how the residuals plot of this model 1 shows some important points still lying far away from the middle area of the graph while Model 2 looks more balanced.

Let plot both the models-

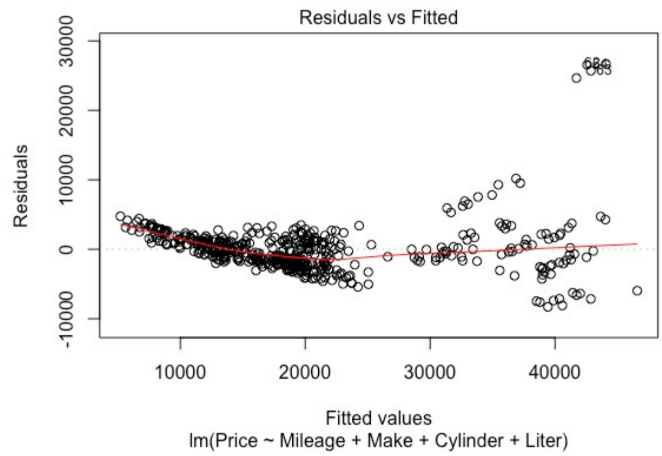
plot(lfit)

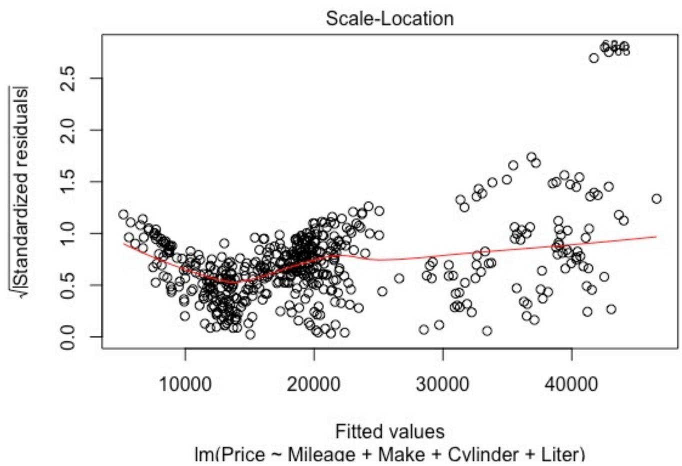


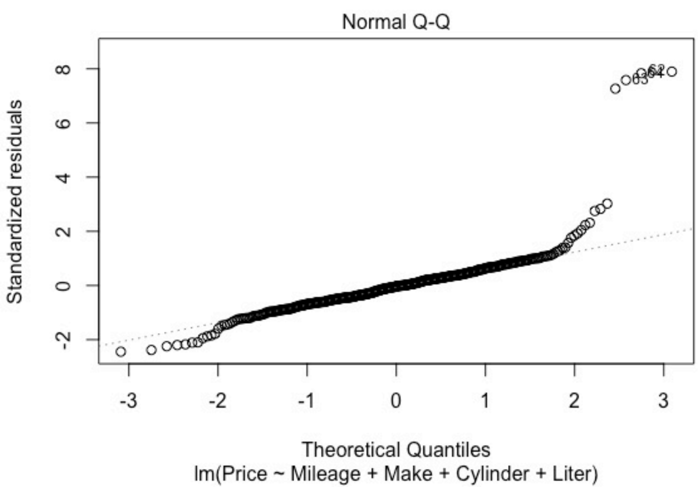


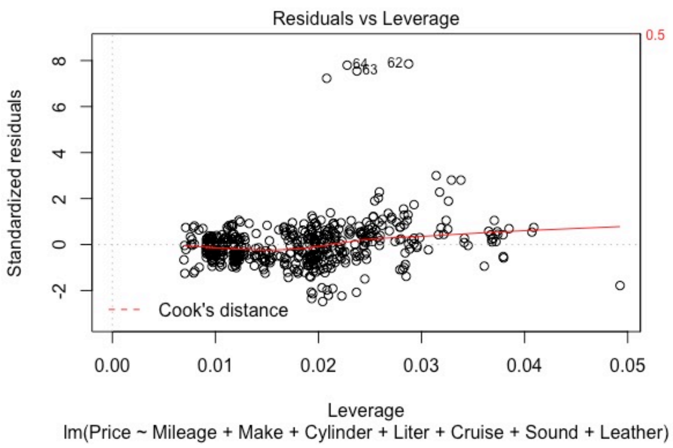


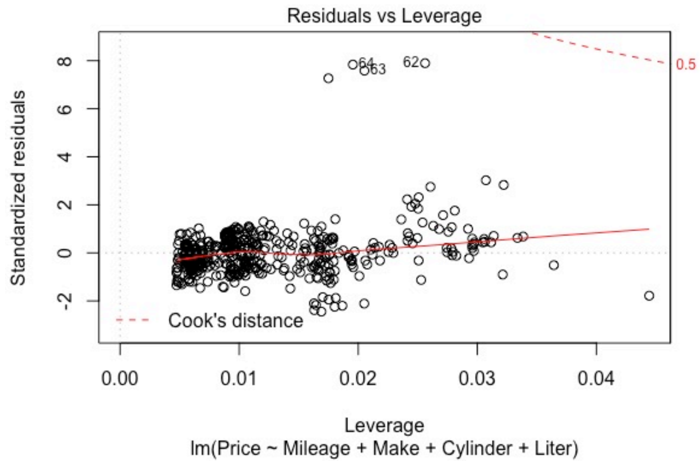
plot(lfit1)











Note now that this updated **model(lfit1)** yields a little bit better R-square measure of 0.8808(not much different from **lfit(0.8808))**, with all predictor p-values highly significant and improved F-Statistic value (738.7). The residuals plot also shows a randomly scattered plot indicating a relatively good fit given the transformations applied due to the non-linearity nature of the data.

We saw a two different multiple linear regression models applied to this dataset and we find out that second model is better than first one by little margin.

But this doesn't mean we cant improve this model more. We can get better predictions by using log and 1/squre function.

I will show one by one. Lets start with log. If i use log function on first model

Model no 3 - lfit2 <- lm(log(Price) ~ Mileage + Make + Cylinder + Liter , data = data).
summary(lfit2)

```
Call:
lm(formula = log(Price) ~ Mileage + Make + Cylinder + Liter,
    data = data)

Residuals:
     Min       1Q   Median       3Q      Max
-0.31353 -0.06646  0.00039  0.06252  0.42301

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)    9.926e+00  4.548e-02 218.269   <2e-16 ***
Mileage       -8.859e-06  5.220e-07 -16.973   <2e-16 ***
MakeChevrolet -6.463e-01  1.949e-02 -33.157   <2e-16 ***
MakePontiac   -6.544e-01  1.883e-02 -34.760   <2e-16 ***
Cylinder      -1.017e-01  1.370e-02  -7.424    5e-13 ***
Liter          3.648e-01  1.533e-02  23.794   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.09675 on 494 degrees of freedom
Multiple R-squared:  0.9462,    Adjusted R-squared:  0.9457
F-statistic:  1738 on 5 and 494 DF,  p-value: < 2.2e-16
```

Adjusted R-square(0.9457) and Multiple R-square (0.9462) in this model are way better than previous two models. It predicts 94% accuracy in compare to 88% in case of last two model.

Model no -4
lfit3 <- lm(1/sqrt(Price) ~ Mileage + Make + Cylinder + Liter , data = data)
summary(lfit3)

```
Call:
lm(formula = 1/sqrt(Price) ~ Mileage + Make + Cylinder + Liter,
    data = data)

Residuals:
       Min         1Q     Median         3Q        Max 
-9.798e-04 -2.224e-04 -1.778e-05  2.471e-04  1.201e-03 

Coefficients:
                Estimate Std. Error t value Pr(>|t|)    
(Intercept)    7.418e-03  1.672e-04  44.358  < 2e-16 ***
Mileage        3.201e-08  1.919e-09  16.678  < 2e-16 ***
MakeChevrolet  2.074e-03  7.167e-05  28.940  < 2e-16 ***
MakePontiac    2.001e-03  6.922e-05  28.907  < 2e-16 ***
Cylinder       3.686e-04  5.039e-05   7.315 1.05e-12 ***
Liter         -1.328e-03  5.638e-05 -23.552  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.0003557 on 494 degrees of freedom
Multiple R-squared:  0.9408,	Adjusted R-squared:  0.9402 
F-statistic:  1570 on 5 and 494 DF,  p-value: < 2.2e-16
```

Adjusted R-square(0.9402) and Multiple R-square (0.9408) in this model are way better than previous two models. But it’s not better than model no 3 by little margin. It predicts 94% accuracy in compare to 88% in case of last two model.

If we compare all the four models, Model no 3 seems better in predicting the price.

	R-Square	Adjusted R-Square
Model no 1	0.8803	0.8823
model no 2	0.8808	0.8830
Model no 3	**0.9457**	**0.9462**
Model no 4	0.9402	0.9408

Here I will go with model 3. F statics is almost double in model no 3(1570) than model 1 and 2.