DMPM Assignment 2: Part 2

Name: Saniya S. Inamdar

SRN: 201900913

Roll no. : 17

CODE:

```
library(dplyr)
library(scales)
carsdf <- read.csv("ToyotaCorolla.csv")
head(carsdf)
summary(carsdf)
summary(carsdf$FuelType)
str(carsdf)

carsdf %>%
  count(FuelType)

#lets encode the FuelType, 1=CNG, 2= Deisel, 3= petrol
encode_ordinal <- function(x, order = unique(x)) {
  x <- as.numeric(factor(x, levels = order, exclude = NULL))
  x
}

carsdf[["Fuel_encoded"]] <- encode_ordinal(carsdf[["FuelType"]])
head(carsdf)

carsdf %>%
  count(Fuel_encoded)

model <- lm(Price~Age+KM+HP+MetColor+Automatic+CC+Doors+Weight+Fuel_encoded, data=carsdf

summary(model)

print(model)

pred<-predict(model)
resi <- resid(model)
```

```r
#lets prove the correlations by plotting the scatter plot

plot(carsdf$Price,carsdf$Age,
     main="Age and Price",
     abline(lm(carsdf$Age~carsdf$Price)),
     ylab = "Age(in Years)",
     xlab="Price(in EUROS)")

plot(carsdf$Price,carsdf$KM,
     main="KM and Price",
     abline(lm(carsdf$KM~carsdf$Price)),
     ylab = "Kilometers(KM)",
     xlab="Price(in EUROS)")

plot(carsdf$Price,carsdf$HP,
     main="HorsePower(HP) and Price",
     abline(lm(carsdf$HP~carsdf$Price)), ylab =
       "HorsePower(HP)", xlab="Price(in
EUROS)")

plot(carsdf$Price,carsdf$CC,
     main="Cylinder volume(in cc) and Price",
     abline(lm(carsdf$CC~carsdf$Price)), ylab =
       "Cylinder Volume(in cc)", xlab="Price(in
EUROS)")

plot(carsdf$Price,carsdf$Weight,
     main="Weight and Price",
     abline(lm(carsdf$Weight~carsdf$Price)), ylab
     = "Weight(in kg)", xlab="Price(in
EUROS)")
```

```r
plot(carsdf$Age,resi,
     main = "Residual Plot(Age and Price)",
     abline(0,0), ylab = "Residuals", xlab
     = "Age(in years)")
plot(carsdf$KM,resi,
     main = "Residual Plot(KM and Price)",
     abline(0,0), ylab = "Residuals",
     xlab = "KM")
plot(carsdf$HP,resi,
     main = "Residual Plot(HP and Price)",
     abline(0,0), ylab = "Residuals", xlab =
       "HP")
plot(carsdf$CC,resi,
     main = "Residual Plot(CC and Price)",
     abline(0,0), ylab = "Residuals", xlab =
       "CC")
plot(carsdf$Weight,resi,
     main = "Residual Plot(Weight and Price)",
     abline(0,0), ylab = "Residuals", xlab =
       "Weight(in kg)")


#evaluation
x<-cbind(carsdf$Price,pred)
x<-data.matrix(x)
x<-rescale(x)
x<-as.data.frame(x)
mae<-mae(x$V1,x$pred)
mse<-mse(x$V1,x$pred)
rmse<-rmse(x$V1,x$pred)
cat("\nMAE:",mae,"\n\nMSE:",mse,"\n\nRMSE:",rmse,"\n\n")
```

OUTPUT:

```
> head(carsdf)
  Price Age    KM FuelType HP MetColor Automatic   CC Doors Weight
1 13500  23 46986   Diesel 90        1         0 2000     3   1165
2 13750  23 72937   Diesel 90        1         0 2000     3   1165
3 13950  24 41711   Diesel 90        1         0 2000     3   1165
4 14950  26 48000   Diesel 90        0         0 2000     3   1165
5 13750  30 38500   Diesel 90        0         0 2000     3   1170
6 12950  32 61000   Diesel 90        0         0 2000     3   1170
> summary(carsdf)
     Price            Age              KM            FuelType              HP
 Min.   : 4350   Min.   : 1.00   Min.   :     1   Length:1436        Min.   : 69.0
 1st Qu.: 8450   1st Qu.:44.00   1st Qu.: 43000   Class :character   1st Qu.: 90.0
 Median : 9900   Median :61.00   Median : 63390   Mode  :character   Median :110.0
 Mean   :10731   Mean   :55.95   Mean   : 68533                      Mean   :101.5
 3rd Qu.:11950   3rd Qu.:70.00   3rd Qu.: 87021                      3rd Qu.:110.0
 Max.   :32500   Max.   :80.00   Max.   :243000                      Max.   :192.0
    MetColor        Automatic             CC           Doors           Weight
 Min.   :0.0000   Min.   :0.00000   Min.   :1300   Min.   :2.000   Min.   :1000
 1st Qu.:0.0000   1st Qu.:0.00000   1st Qu.:1400   1st Qu.:3.000   1st Qu.:1040
 Median :1.0000   Median :0.00000   Median :1600   Median :4.000   Median :1070
 Mean   :0.6748   Mean   :0.05571   Mean   :1567   Mean   :4.033   Mean   :1072
 3rd Qu.:1.0000   3rd Qu.:0.00000   3rd Qu.:1600   3rd Qu.:5.000   3rd Qu.:1085
 Max.   :1.0000   Max.   :1.00000   Max.   :2000   Max.   :5.000   Max.   :1615
```

There are 10 columns in total.

There is no missing record in the dataset.

There are three categorical variable (FuelType, MetColor and Automatic) and rest are numerical.

I check the datatypes of the variables.

```
> str(carsdf)
'data.frame':   1436 obs. of  10 variables:
 $ Price    : int  13500 13750 13950 14950 13750 12950 16900 18600 21500 12950 ...
 $ Age      : int  23 23 24 26 30 32 27 30 27 23 ...
 $ KM       : int  46986 72937 41711 48000 38500 61000 94612 75889 19700 71138 ...
 $ FuelType : chr  "Diesel" "Diesel" "Diesel" "Diesel" ...
 $ HP       : int  90 90 90 90 90 90 90 90 192 69 ...
 $ MetColor : int  1 1 1 0 0 0 1 1 0 0 ...
 $ Automatic: int  0 0 0 0 0 0 0 0 0 0 ...
 $ CC       : int  2000 2000 2000 2000 2000 2000 2000 2000 1800 1900 ...
 $ Doors    : int  3 3 3 3 3 3 3 3 3 3 ...
 $ Weight   : int  1165 1165 1165 1165 1170 1170 1245 1245 1185 1105 ...
```

```
> carsdf %>%
+   count(FuelType)
  FuelType    n
1      CNG   17
2   Diesel  155
3   Petrol 1264
> #lets encode the FuelType, 1=CNG, 2= Deisel, 3= petrol
> encode_ordinal <- function(x, order = unique(x)) {
+   x <- as.numeric(factor(x, levels = order, exclude = NULL))
+   x
+ }
> carsdf[["Fuel_encoded"]] <- encode_ordinal(carsdf[["FuelType"]])
> head(carsdf)
  Price Age    KM FuelType HP MetColor Automatic   CC Doors Weight Fuel_encoded
1 13500  23 46986   Diesel 90        1         0 2000     3   1165            1
2 13750  23 72937   Diesel 90        1         0 2000     3   1165            1
3 13950  24 41711   Diesel 90        1         0 2000     3   1165            1
4 14950  26 48000   Diesel 90        0         0 2000     3   1165            1
5 13750  30 38500   Diesel 90        0         0 2000     3   1170            1
6 12950  32 61000   Diesel 90        0         0 2000     3   1170            1
> carsdf %>%
+   count(Fuel_encoded)
  Fuel_encoded    n
1            1  155
2            2 1264
3            3   17
```

Above I applied Ordinal encoding to the FuelType variable, which I later realise that it’s not necessary.

```
> model <- lm(Price~Age+KM+HP+MetColor+Automatic+CC+Doors+Weight+Fuel_encoded, data=carsdf)
> summary(model)

Call:
lm(formula = Price ~ Age + KM + HP + MetColor + Automatic + CC +
    Doors + Weight + Fuel_encoded, data = carsdf)

Residuals:
     Min       1Q   Median       3Q      Max
-11209.6   -748.0      8.9    735.9   6374.1

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  -8.036e+02  1.300e+03  -0.618   0.5364
Age          -1.226e+02  2.589e+00 -47.336  < 2e-16 ***
KM           -1.567e-02  1.285e-03 -12.190  < 2e-16 ***
HP            5.279e+01  4.084e+00  12.926  < 2e-16 ***
MetColor      5.563e+01  7.501e+01   0.742   0.4584
Automatic     2.905e+02  1.560e+02   1.863   0.0627 .
CC           -3.446e+00  4.024e-01  -8.565  < 2e-16 ***
Doors        -2.535e+01  3.910e+01  -0.648   0.5169
Weight        2.099e+01  1.096e+00  19.151  < 2e-16 ***
Fuel_encoded -1.555e+03  2.497e+02  -6.225 6.31e-10 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1317 on 1426 degrees of freedom
Multiple R-squared:  0.869,     Adjusted R-squared:  0.8681
F-statistic:  1051 on 9 and 1426 DF,  p-value: < 2.2e-16
```

Significant columns: Age, KM, HP, CC, Weight

```
> print(model)

Call:
lm(formula = Price ~ Age + KM + HP + MetColor + Automatic + CC +
    Doors + Weight + Fuel_encoded, data = carsdf)

Coefficients:
 (Intercept)           Age            KM            HP      MetColor     Automatic
  -8.036e+02    -1.226e+02    -1.567e-02     5.279e+01     5.563e+01     2.905e+02
          CC         Doors        Weight  Fuel_encoded
  -3.447e+00    -2.535e+01     2.099e+01    -1.555e+03
```

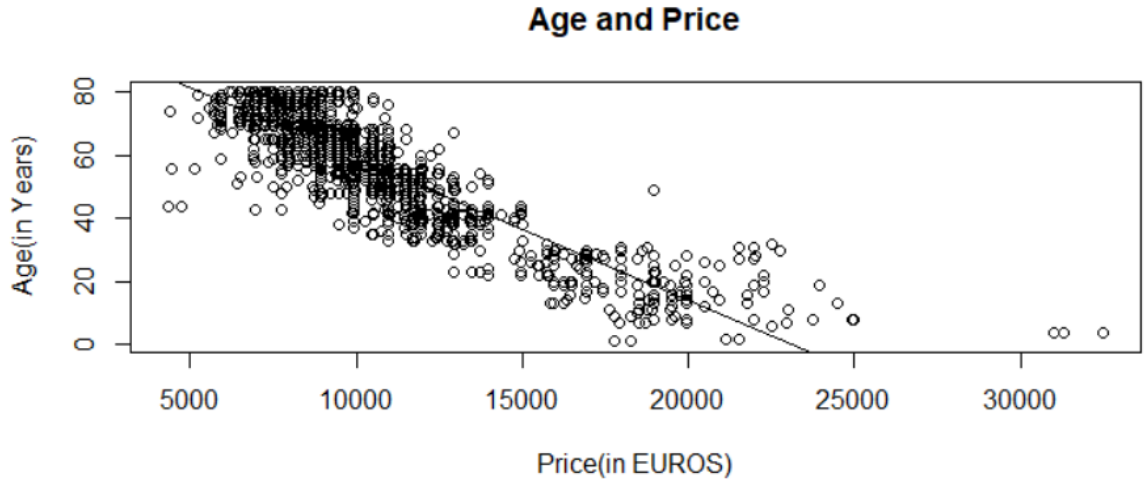
Above, we can observe the coefficients of correlation with the feature variables.

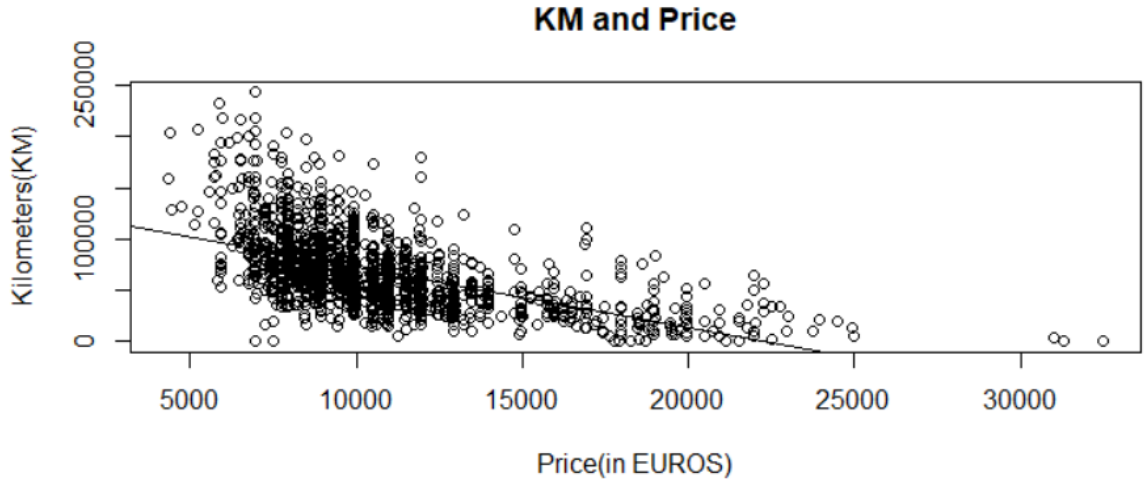
For example: For “Age” the coefficient is negative, so the”Age” is negatively correlated with the “Price”, so we can conclude that, More the Age of the car less is it’s resell Price.

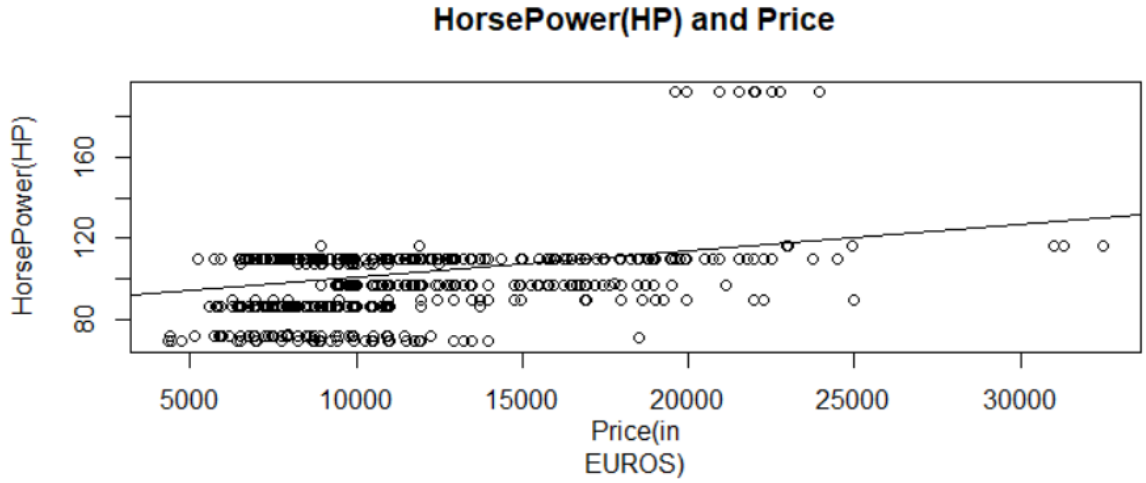
Similarly for other features.

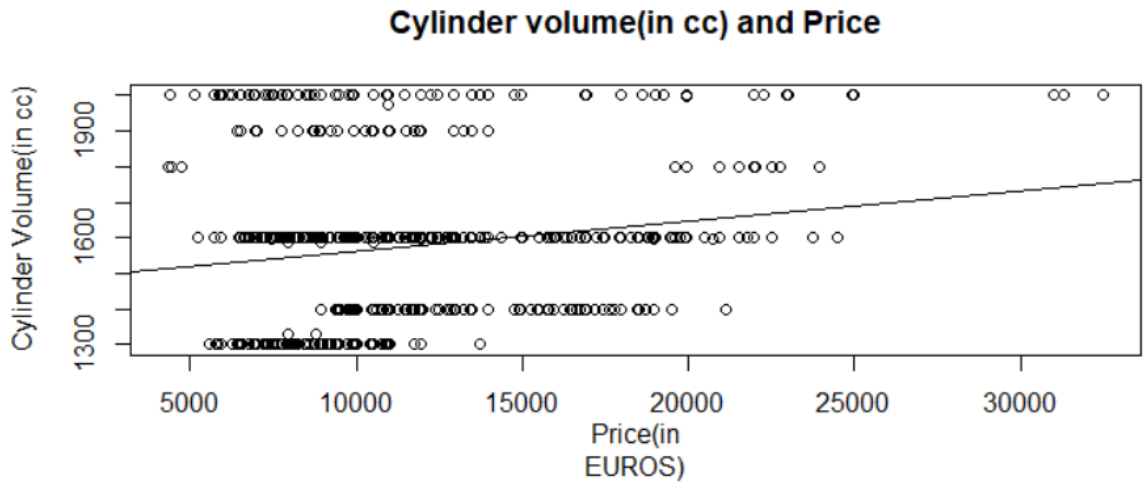
Lets prove it by plotting some scatter plots with slope:

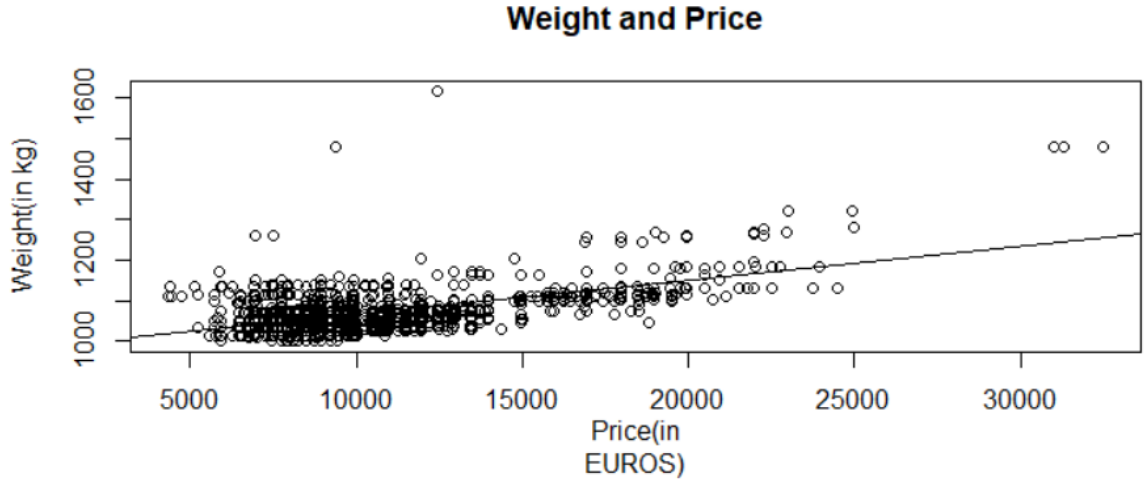
```
> plot(carsdf$Price,carsdf$Age,
+      main="Age and Price",
+      abline(lm(carsdf$Age~carsdf$Price)),
+      ylab = "Age(in Years)",
+      xlab="Price(in EUROS)")
> plot(carsdf$Price,carsdf$KM,
+      main="KM and Price",
+      abline(lm(carsdf$KM~carsdf$Price)),
+      ylab = "Kilometers(KM)",
+      xlab="Price(in EUROS)")
> plot(carsdf$Price,carsdf$HP,
+      main="HorsePower(HP) and Price",
+      abline(lm(carsdf$HP~carsdf$Price)), ylab =
+        "HorsePower(HP)", xlab="Price(in
+ EUROS)")
> plot(carsdf$Price,carsdf$CC,
+      main="Cylinder volume(in cc) and Price",
+      abline(lm(carsdf$CC~carsdf$Price)), ylab =
+        "Cylinder Volume(in cc)", xlab="Price(in
+ EUROS)")
> plot(carsdf$Price,carsdf$Weight,
+      main="Weight and Price",
+      abline(lm(carsdf$Weight~carsdf$Price)), ylab
+      = "Weight(in kg)", xlab="Price(in
+ EUROS)")
```


Age and Price
Age(in Years)
0
20
40
60
80
5000
10000
15000
20000
25000
30000
Price(in EUROS)


KM and Price
Kilometers(KM)
0
100000
250000
5000
10000
15000
20000
25000
30000
Price(in EUROS)





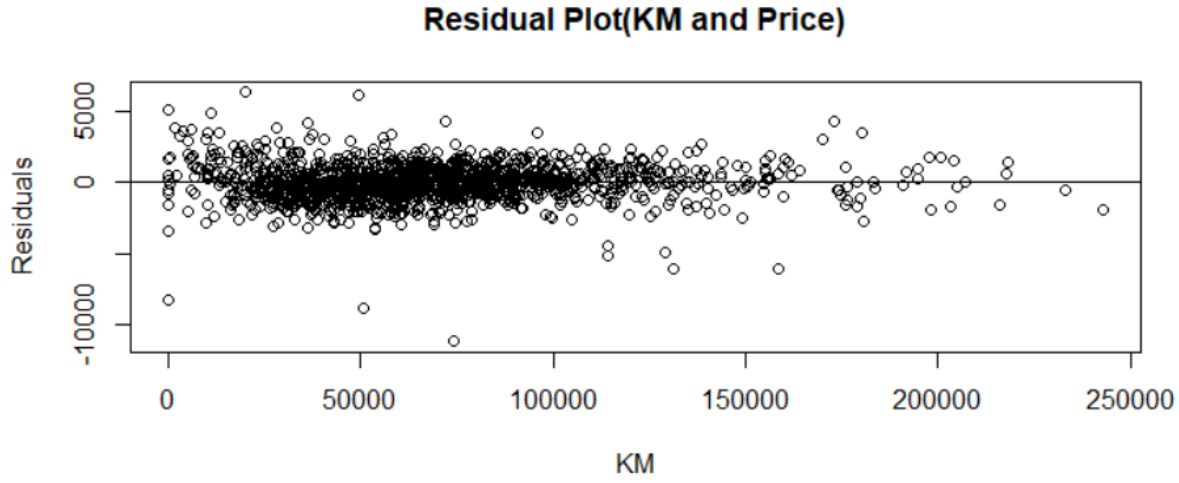


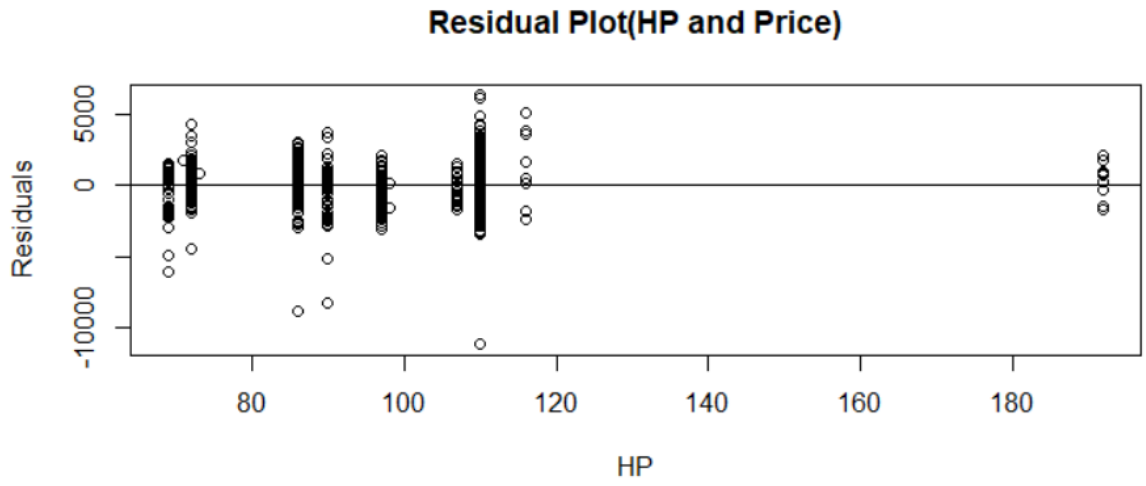

```
> pred<-predict(model)
> resi <- resid(model)
```

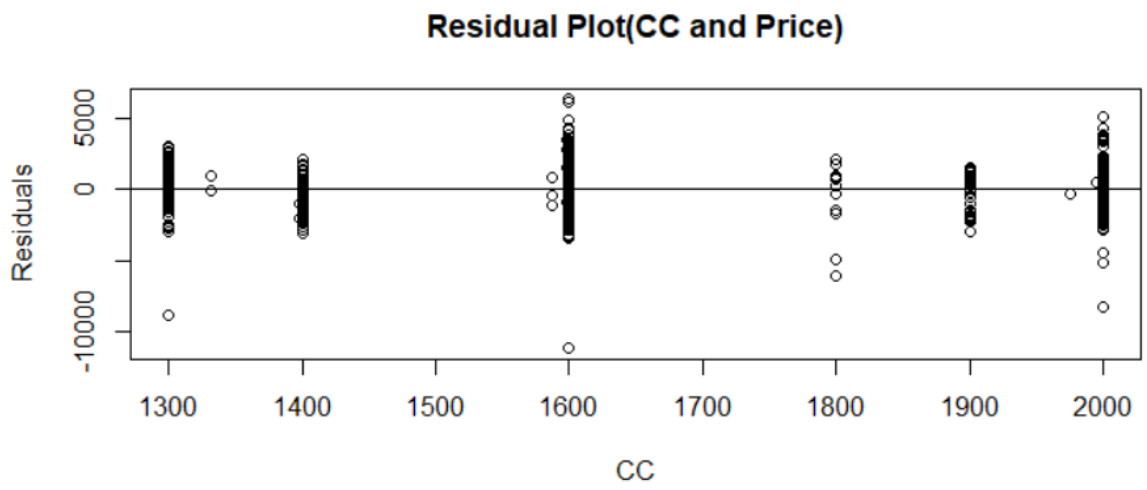
Computed the Predictions and residuals

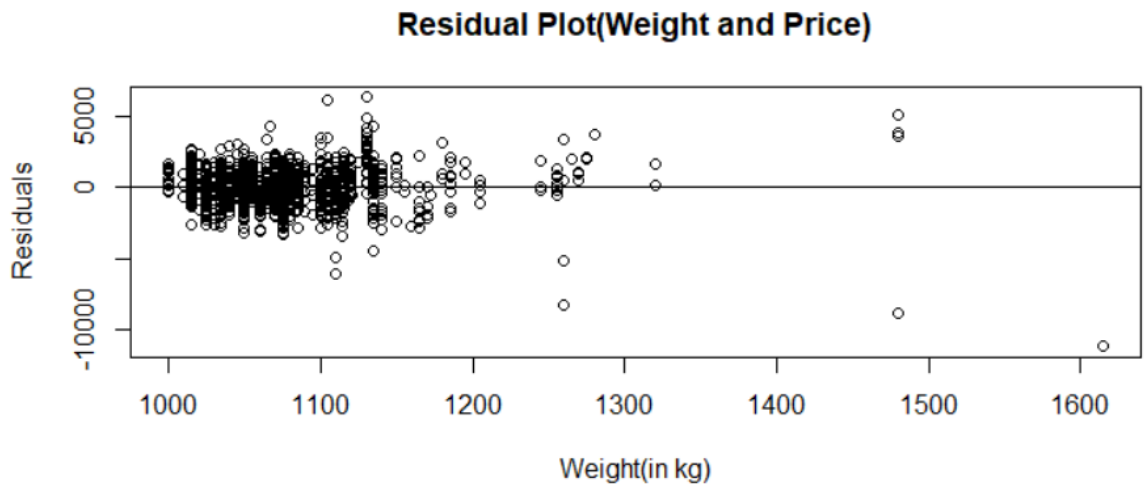
Let’s plot some residual plots:

```
> plot(carsdf$Age,resi,
+      main = "Residual Plot(Age and Price)",
+      abline(0,0), ylab = "Residuals", xlab
+      = "Age(in years)")
> plot(carsdf$KM,resi,
+      main = "Residual Plot(KM and Price)",
+      abline(0,0), ylab = "Residuals",
+      xlab = "KM")
> plot(carsdf$HP,resi,
+      main = "Residual Plot(HP and Price)",
+      abline(0,0), ylab = "Residuals", xlab =
+        "HP")
> plot(carsdf$CC,resi,
+      main = "Residual Plot(CC and Price)",
+      abline(0,0), ylab = "Residuals", xlab =
+        "CC")
> plot(carsdf$Weight,resi,
+      main = "Residual Plot(Weight and Price)",
+      abline(0,0), ylab = "Residuals", xlab =
+        "Weight(in kg)")
```


Residual Plot(KM and Price)
Residuals
5000
0
-10000
0
50000
100000
150000
200000
250000
KM


Residual Plot(HP and Price)
Residuals
5000
0
-10000
80
100
120
140
160
180
HP





Now it's time to evaluate the Model I created

```
> x<-cbind(carsdf$Price,pred)
> x<-data.matrix(x)
> x<-rescale(x)
> x<-as.data.frame(x)
> mae<-mae(x$V1,x$pred)
> mse<-mse(x$V1,x$pred)
> rmse<-rmse(x$V1,x$pred)
> cat("\nMAE:",mae,"\n\nMSE:",mse,"\n\nRMSE:",rmse,"\n\n")

MAE: 0.03374678

MSE: 0.002173777

RMSE: 0.04662378
```

The metrics I used to judge the model performance are: Mean Absolute error(MAE), Mean Squared Error(MSE) and Root Mean Squared Error(RMSE).

The less the value of these metrics the more good the model is. Here we get :

MAE: 0.03374678

MSE: 0.002173777

RMSE: 0.04662378

We can condlude that the model performance if up to the mark.

I also tried experimenting with different features, by removing some insignificant or less correlated features, But there was no increase in the performance metrics.

END
