Regression Project Using R- Car Price Prediction

Motivation:

A Chinese automobile company aspires to enter the US market by setting up their manufacturing unit there and producing car locally to compete with the US and European counterpart. They have contracted an automobile consulting company to understand the factors on which the price of cars depends. the main aim is to understand the factors affecting the pricing of cars in American market, Since those may vary different from the Chinese market. The company want to know following:

1)Which variables are significant in predicting the price of car? 2) How well those variable describe the price of car?

Objective:

We are required to model the price of cars with the available independent variables. It will be used by the management to understand how exactly the prices vary with the indepent variables. The management can accordingly launch the design variants and plan its business strategy. Further, the model will be a good way for management to understand the pricing dynamics of a new market.

Outcome:

By building this model and testing its significant the management will be able to see its usefulness under different hypothesis sceneria. ultimately the designed Multi linear regression model should be able to fit the objective of the Model.

Data Source:

The data Source considered for the project is taken from - "[https://www.kaggle.com/car-price-prediction"](https://www.kaggle.com/car-price-prediction)

Variable in the dataset:

There are 26 variable in the dataset conisdered. out of this 26 variables: Integer type- 1 nos. Categorial type- 11 nos. Numerical type/Continous variable-14nos. The response/dependent variable of the dataset is "Price"

Name	Type	Details
Car_ID	Interger	Unique id of each observation
Symboling	Categorical	Its assigned insurance risk rating, A value of +3 indicates that the auto is risky, -3 that it is probably pretty safe.
carCompany	Categorical	Name of car company
fueltype	Categorical	Car fuel type i.e gas or diesel
aspiration	Categorical	Aspiration used in a car
doornumber	Categorical	Number of doors in a car
carbody	Categorical	body of car
drivewheel	Categorical	type of drive wheel
enginelocation	Categorical	Location of car engine
wheelbase	Numeric	Weelbase of car
carlength	Numeric	Length of car
carwidth	Numeric	Width of car
carheight	Numeric	height of car
curbweight	Numeric	The weight of a car without occupants or baggage.
enginetype	Categorical	Type of engine.
cylindernumber	Categorical	cylinder placed in the car
enginesize	Numeric	Size of car
fuelsystem	Categorical	Fuel system of car
boreratio	Numeric	Boreratio of car
stroke	Numeric	Stroke or volume inside the engine
compressionratio	Numeric	compression ratio of car
horsepower	Numeric	Horsepower
peakrpm	Numeric	car peak rpm
citympg	Numeric	Mileage in city
highwaympg	Numeric	Mileage on highway
price(Dependent variable)	Numeric	Price of car
Allcars (derived variable)	Categorical	Car company name

##Installing packages in R studio:

```
install.packages("tidyverse")
library(tidyverse)
install.packages("dplyr")
library(dplyr)
install.packages("ggplot2")
library(ggplot2)
install.packages("caret")
library(caret)
install.packages("car")
library(car)
install.packages("nortest")
library(nortest)
install.packages("ggplot2")
library(ggplot2)
```

input data to R studio

```
car_prediction=read.csv("C:/Users/admin/Desktop/MLR project/car prediction/CarPrice_project.csv")
```

##Checking head data:

```
head(car_prediction)
```

```
  car_ID symboling                  CarName fueltype
1      1         3       alfa-romero giulia      gas
2      2         3      alfa-romero stelvio      gas
3      3         1 alfa-romero Quadrifoglio      gas
4      4         2              audi 100 ls      gas
5      5         2               audi 100ls      gas
6      6         2                 audi fox      gas
  aspiration doornumber     carbody drivewheel
1        std        two convertible        rwd
2        std        two convertible        rwd
3        std        two   hatchback        rwd
4        std       four       sedan        fwd
5        std       four       sedan        4wd
6        std        two       sedan        fwd
  enginelocation wheelbase carlength carwidth carheight
1          front      88.6     168.8     64.1      48.8
2          front      88.6     168.8     64.1      48.8
3          front      94.5     171.2     65.5      52.4
4          front      99.8     176.6     66.2      54.3
5          front      99.4     176.6     66.4      54.3
6          front      99.8     177.3     66.3      53.1
  curbweight enginetype cylindernumber enginesize fuelsystem
1       2548       dohc           four        130       mpfi
2       2548       dohc           four        130       mpfi
3       2823       ohcv            six        152       mpfi
4       2337        ohc           four        109       mpfi
5       2824        ohc           five        136       mpfi
6       2507        ohc           five        136       mpfi
  boreratio stroke compressionratio horsepower peakrpm
1      3.47   2.68              9.0        111    5000
2      3.47   2.68              9.0        111    5000
3      2.68   3.47              9.0        154    5000
4      3.19   3.40             10.0        102    5500
5      3.19   3.40              8.0        115    5500
```

```
6      3.19   3.40             8.5        110     5500
  citympg highwaympg price
1      21         27 13495
2      21         27 16500
3      19         26 16500
4      24         30 13950
5      18         22 17450
6      19         25 15250
```

##Checking tail data

```
tail(car_prediction)
```

```
    car_ID symboling         CarName fueltype aspiration
200    200        -1    volvo diesel      gas      turbo
201    201        -1 volvo 145e (sw)      gas        std
202    202        -1     volvo 144ea      gas      turbo
203    203        -1     volvo 244dl      gas        std
204    204        -1       volvo 246   diesel      turbo
205    205        -1     volvo 264gl      gas      turbo
    doornumber carbody drivewheel enginelocation wheelbase
200       four   wagon        rwd          front     104.3
201       four   sedan        rwd          front     109.1
202       four   sedan        rwd          front     109.1
203       four   sedan        rwd          front     109.1
204       four   sedan        rwd          front     109.1
205       four   sedan        rwd          front     109.1
    carlength carwidth carheight curbweight enginetype
200     188.8     67.2      57.5       3157        ohc
201     188.8     68.9      55.5       2952        ohc
202     188.8     68.8      55.5       3049        ohc
203     188.8     68.9      55.5       3012       ohcv
204     188.8     68.9      55.5       3217        ohc
205     188.8     68.9      55.5       3062        ohc
    cylindernumber enginesize fuelsystem boreratio stroke
200           four        130       mpfi      3.62   3.15
201           four        141       mpfi      3.78   3.15
202           four        141       mpfi      3.78   3.15
203            six        173       mpfi      3.58   2.87
204            six        145        idi      3.01   3.40
205           four        141       mpfi      3.78   3.15
    compressionratio horsepower peakrpm citympg highwaympg
200              7.5        162    5100      17         22
201              9.5        114    5400      23         28
202              8.7        160    5300      19         25
203              8.8        134    5500      18         23
204             23.0        106    4800      26         27
205              9.5        114    5400      19         25
    price
200 18950
201 16845
202 19045
203 21485
204 22470
```

```
##Cleaning the Data
##Convert some variable types to Numeric from integer

car_prediction <- car_prediction %>%
 mutate(symboling = as.character(symboling),
     enginesize = as.numeric(enginesize),
     horsepower = as.numeric(horsepower),
     peakrpm = as.numeric(peakrpm),
     citympg = as.numeric(citympg),
     highwaympg = as.numeric(highwaympg),
     curbweight = as.numeric(curbweight)
 ) %>%
 select(-car_ID)


# #Convert some variable types to Numeric from character

unique(car_prediction$CarName)
car_prediction$CarName <- gsub("maxda", "mazda", car_prediction$CarName)
car_prediction$CarName <- gsub("nissan", "Nissan", car_prediction$CarName)
car_prediction$CarName <- gsub("porcshce", "porsche", car_prediction$CarName)
car_prediction$CarName <- gsub("toyouta", "toyota", car_prediction$CarName)
car_prediction$CarName <- gsub("vokswagen", "volkswagen", car_prediction$CarName)
car_prediction$CarName <- gsub("vw", "volkswagen", car_prediction$CarName)
unique(car_prediction$CarName)
```

```
[1] "alfa-romero giulia"
  [2] "alfa-romero stelvio"
  [3] "alfa-romero Quadrifoglio"
  [4] "audi 100 ls"
  [5] "audi 100ls"
  [6] "audi fox"
  [7] "audi 5000"
  [8] "audi 4000"
  [9] "audi 5000s (diesel)"
 [10] "bmw 320i"
 [11] "bmw x1"
 [12] "bmw x3"
 [13] "bmw z4"
 [14] "bmw x4"
 [15] "bmw x5"
 [16] "chevrolet impala"
 [17] "chevrolet monte carlo"
 [18] "chevrolet vega 2300"
 [19] "dodge rampage"
 [20] "dodge challenger se"
 [21] "dodge d200"
 [22] "dodge monaco (sw)"
 [23] "dodge colt hardtop"
 [24] "dodge colt (sw)"
 [25] "dodge coronet custom"
 [26] "dodge dart custom"
 [27] "dodge coronet custom (sw)"
 [28] "honda civic"
 [29] "honda civic cvcc"
```

```
[30] "honda accord cvcc"
[31] "honda accord lx"
[32] "honda civic 1500 gl"
[33] "honda accord"
[34] "honda civic 1300"
[35] "honda prelude"
[36] "honda civic (auto)"
[37] "isuzu MU-X"
[38] "isuzu D-Max "
[39] "isuzu D-Max V-Cross"
[40] "jaguar xj"
[41] "jaguar xf"
[42] "jaguar xk"
[43] "mazda rx3"
[44] "mazda glc deluxe"
[45] "mazda rx2 coupe"
[46] "mazda rx-4"
[47] "mazda 626"
[48] "mazda glc"
[49] "mazda rx-7 gs"
[50] "mazda glc 4"
[51] "mazda glc custom l"
[52] "mazda glc custom"
[53] "buick electra 225 custom"
[54] "buick century luxus (sw)"
[55] "buick century"
[56] "buick skyhawk"
[57] "buick opel isuzu deluxe"
[58] "buick skylark"
[59] "buick century special"
[60] "buick regal sport coupe (turbo)"
[61] "mercury cougar"
[62] "mitsubishi mirage"
[63] "mitsubishi lancer"
[64] "mitsubishi outlander"
[65] "mitsubishi g4"
[66] "mitsubishi mirage g4"
[67] "mitsubishi montero"
[68] "mitsubishi pajero"
[69] "Nissan versa"
[70] "Nissan gt-r"
[71] "Nissan rogue"
[72] "Nissan latio"
[73] "Nissan titan"
[74] "Nissan leaf"
[75] "Nissan juke"
[76] "Nissan note"
[77] "Nissan clipper"
[78] "Nissan nv200"
[79] "Nissan dayz"
[80] "Nissan fuga"
[81] "Nissan otti"
[82] "Nissan teana"
[83] "Nissan kicks"
[84] "peugeot 504"
[85] "peugeot 304"
[86] "peugeot 504 (sw)"
[87] "peugeot 604sl"
[88] "peugeot 505s turbo diesel"
[89] "plymouth fury iii"
[90] "plymouth cricket"
[91] "plymouth satellite custom (sw)"
```

```
 [92] "plymouth fury gran sedan"
 [93] "plymouth valiant"
 [94] "plymouth duster"
 [95] "porsche macan"
 [96] "porsche panamera"
 [97] "porsche cayenne"
 [98] "porsche boxter"
 [99] "renault 12tl"
[100] "renault 5 gtl"
[101] "saab 99e"
[102] "saab 99le"
[103] "saab 99gle"
[104] "subaru"
[105] "subaru dl"
[106] "subaru brz"
[107] "subaru baja"
[108] "subaru r1"
[109] "subaru r2"
[110] "subaru trezia"
[111] "subaru tribeca"
[112] "toyota corona mark ii"
[113] "toyota corona"
[114] "toyota corolla 1200"
[115] "toyota corona hardtop"
[116] "toyota corolla 1600 (sw)"
[117] "toyota carina"
[118] "toyota mark ii"
[119] "toyota corolla"
[120] "toyota corolla liftback"
[121] "toyota celica gt liftback"
[122] "toyota corolla tercel"
[123] "toyota corona liftback"
[124] "toyota starlet"
[125] "toyota tercel"
[126] "toyota cressida"
[127] "toyota celica gt"
[128] "volkswagen rabbit"
[129] "volkswagen 1131 deluxe sedan"
[130] "volkswagen model 111"
[131] "volkswagen type 3"
[132] "volkswagen 411 (sw)"
[133] "volkswagen super beetle"
[134] "volkswagen dasher"
[135] "volkswagen rabbit custom"
[136] "volvo 145e (sw)"
[137] "volvo 144ea"
[138] "volvo 244dl"
[139] "volvo 245"
[140] "volvo 264gl"
[141] "volvo diesel"
[142] "volvo 246"
```

##str(car_prediction)

```
str(car_prediction)
```

```
'data.frame':	205 obs. of  26 variables:
 $ car_ID          : int  1 2 3 4 5 6 7 8 9 10 ...
 $ symboling       : int  3 3 1 2 2 2 1 1 1 0 ...
```

```
 $ CarName        : chr  "alfa-romero giulia" "alfa-romero stelvio" "alfa-romero Quadrifoglio" "audi 100 ls" ...
 $ fueltype       : chr  "gas" "gas" "gas" "gas" ...
 $ aspiration     : chr  "std" "std" "std" "std" ...
 $ doornumber     : chr  "two" "two" "two" "four" ...
 $ carbody        : chr  "convertible" "convertible" "hatchback" "sedan" ...
 $ drivewheel     : chr  "rwd" "rwd" "rwd" "fwd" ...
 $ enginelocation  : chr  "front" "front" "front" "front" ...
 $ wheelbase      : num  88.6 88.6 94.5 99.8 99.4 ...
 $ carlength      : num  169 169 171 177 177 ...
 $ carwidth       : num  64.1 64.1 65.5 66.2 66.4 66.3 71.4 71.4 71.4 67.9 ...
 $ carheight      : num  48.8 48.8 52.4 54.3 54.3 53.1 55.7 55.7 55.9 52 ...
 $ curbweight     : int  2548 2548 2823 2337 2824 2507 2844 2954 3086 3053 ...
 $ enginetype     : chr  "dohc" "dohc" "ohcv" "ohc" ...
 $ cylindernumber  : chr  "four" "four" "six" "four" ...
 $ enginesize     : int  130 130 152 109 136 136 136 136 131 131 ...
 $ fuelsystem     : chr  "mpfi" "mpfi" "mpfi" "mpfi" ...
 $ boreratio      : num  3.47 3.47 2.68 3.19 3.19 3.19 3.19 3.19 3.13 3.13 ...
 $ stroke         : num  2.68 2.68 3.47 3.4 3.4 3.4 3.4 3.4 3.4 3.4 ...
 $ compressionratio: num  9 9 9 10 8 8.5 8.5 8.5 8.3 7 ...
 $ horsepower     : int  111 111 154 102 115 110 110 110 140 160 ...
 $ peakrpm        : int  5000 5000 5000 5500 5500 5500 5500 5500 5500 5500 ...
 $ citympg        : int  21 21 19 24 18 19 19 19 17 16 ...
 $ highwaympg     : int  27 27 26 30 22 25 25 25 20 22 ...
 $ price          : num  13495 16500 16500 13950 17450 ...
```

##plot Histogram, box plot, scatter plot (response vs. an important predictor)

```
qplot(car_prediction$price, geom="histogram", binwidth=15, main="Histogram for Price", xlab="Price", fill=I("gray"), col=I("red"))+theme_bw()
```

checking unique names of fuel system type.

```
 unique(car_prediction$fuelsystem)
[1] "mpfi" "2bbl" "mfi"  "1bbl" "spfi" "4bbl" "idi"  "spdi"
```

##Checking for any Missing values

```
 lapply(car_prediction,function(x) { sum(is.na(x))})

$car_ID
[1] 0

$symboling
[1] 0

$CarName
[1] 0

$fueltype
[1] 0
```

```
$aspiration
[1] 0

$doornumber
[1] 0

$carbody
[1] 0

$drivewheel
[1] 0

$enginelocation
[1] 0

$wheelbase
[1] 0

$carlength
[1] 0

$carwidth
[1] 0

$carheight
[1] 0

$curbweight
[1] 0

$enginetype
[1] 0

$cylindernumber
[1] 0

$enginesize
[1] 0

$fuelsystem
[1] 0

$boreratio
[1] 0

$stroke
[1] 0

$compressionratio
[1] 0

$horsepower
[1] 0

$peakrpm
[1] 0

$citympg
[1] 0

$highwaympg
```

```
[1] 0

$price
[1] 0
```

[Inference: No Missing values found.]

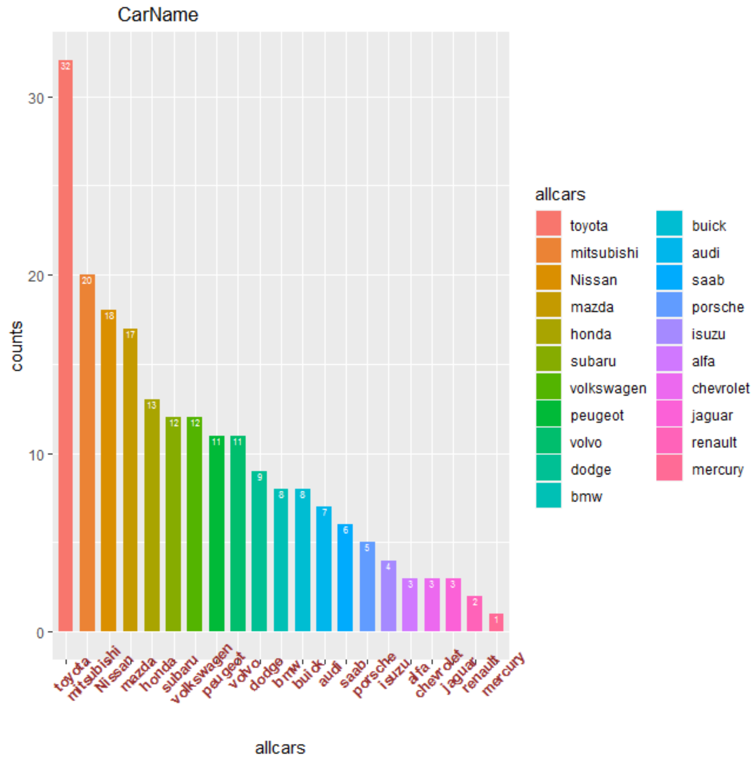
Checking summary of Data

```
summary(car_prediction)
```

```
    car_ID      symboling         CarName         
 Min.   :  1   Min.   :-2.0000   Length:205        
 1st Qu.: 52   1st Qu.: 0.0000   Class :character  
 Median :103   Median : 1.0000   Mode  :character  
 Mean   :103   Mean   : 0.8341                     
 3rd Qu.:154   3rd Qu.: 2.0000                     
 Max.   :205   Max.   : 3.0000                     
   fueltype          aspiration         doornumber       
 Length:205         Length:205         Length:205        
 Class :character   Class :character   Class :character  
 Mode  :character   Mode  :character   Mode  :character  
                                                         
   carbody           drivewheel         enginelocation   
 Length:205         Length:205         Length:205        
 Class :character   Class :character   Class :character  
 Mode  :character   Mode  :character   Mode  :character  
                                                         
   wheelbase        carlength        carwidth    
 Min.   : 86.60   Min.   :141.1   Min.   :60.30  
 1st Qu.: 94.50   1st Qu.:166.3   1st Qu.:64.10  
 Median : 97.00   Median :173.2   Median :65.50  
 Mean   : 98.76   Mean   :174.0   Mean   :65.91  
 3rd Qu.:102.40   3rd Qu.:183.1   3rd Qu.:66.90  
 Max.   :120.90   Max.   :208.1   Max.   :72.30  
   carheight       curbweight    enginetype       
 Min.   :47.80   Min.   :1488   Length:205        
 1st Qu.:52.00   1st Qu.:2145   Class :character  
 Median :54.10   Median :2414   Mode  :character  
 Mean   :53.72   Mean   :2556                     
 3rd Qu.:55.50   3rd Qu.:2935                     
 Max.   :59.80   Max.   :4066                     
 cylindernumber       enginesize     fuelsystem       
 Length:205         Min.   : 61.0   Length:205        
 Class :character   1st Qu.: 97.0   Class :character  
 Mode  :character   Median :120.0   Mode  :character  
                    Mean   :126.9                     
                    3rd Qu.:141.0                     
                    Max.   :326.0                     
```

```
  boreratio        stroke      compressionratio
 Min.   :2.54   Min.   :2.070   Min.   : 7.00  
 1st Qu.:3.15   1st Qu.:3.110   1st Qu.: 8.60  
 Median :3.31   Median :3.290   Median : 9.00  
 Mean   :3.33   Mean   :3.255   Mean   :10.14  
 3rd Qu.:3.58   3rd Qu.:3.410   3rd Qu.: 9.40  
 Max.   :3.94   Max.   :4.170   Max.   :23.00  
  horsepower       peakrpm        citympg     
 Min.   : 48.0   Min.   :4150   Min.   :13.00  
 1st Qu.: 70.0   1st Qu.:4800   1st Qu.:19.00  
 Median : 95.0   Median :5200   Median :24.00  
 Mean   :104.1   Mean   :5125   Mean   :25.22  
 3rd Qu.:116.0   3rd Qu.:5500   3rd Qu.:30.00  
 Max.   :288.0   Max.   :6600   Max.   :49.00  
  highwaympg        price      
 Min.   :16.00   Min.   : 5118  
 1st Qu.:25.00   1st Qu.: 7788  
 Median :30.00   Median :10295  
 Mean   :30.75   Mean   :13277  
 3rd Qu.:34.00   3rd Qu.:16503  
 Max.   :54.00   Max.   :45400  
```

```
##Visualising data
## saving data to new data frame "df"
df<- car_prediction
df<- df %>%
 mutate(allcars = factor(allcars)) %>%
 group_by(allcars) %>%
 summarise(counts= n()) %>%
 arrange(-counts) %>%   # sort by counts
 mutate(allcars = factor(allcars, allcars))

##Plot bar chart for allcars:
ggplot(df, aes(x = allcars, y= counts)) +
 geom_bar(
  aes(x = allcars, y= counts,fill = allcars),
  stat = "identity", position = position_dodge(0.8),
  width = 0.7
 ) +   theme(axis.text.x = element_text(face="bold", color="#993333", size=9, angle=45))+
 ggtitle("          CarName          ") +
 geom_text(aes(label= counts),vjust=1.0, color="white", size=2.0)
```



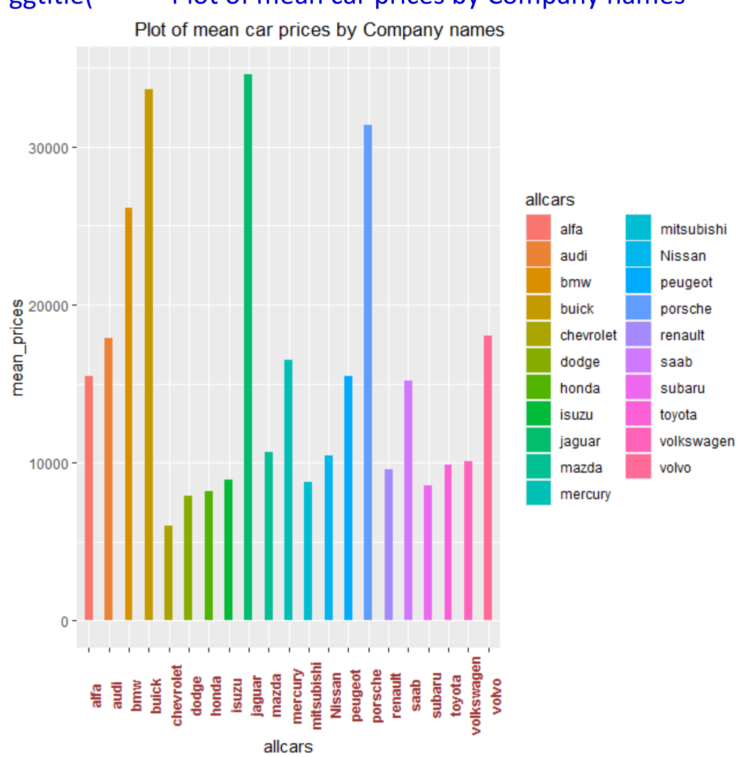
[Inference : Toyota seems to be the most favored car whereas Mercury is least preferred.]

```
##saving data to new data frame “df1”
##using mutate function to group all car company name only.
df1<- car_prediction
df1<- df1 %>%
  mutate(allcars = factor(allcars)) %>%
  group_by(allcars) %>%
  summarise(mean_prices= mean(price,na.rm=T)) %>%
  arrange(-mean_prices) %>%   # sort by counts
  mutate(allcars = factor(allcars, allcars))
glimpse(df1)
```

```
Rows: 21
Columns: 2
$ allcars     <fct> jaguar, buick, porsche, bmw, volvo, aud~
$ mean_prices <dbl> 34600.000, 33647.000, 31400.500, 26118.~
```

[Inference: Jaguar,Buick and porsche seems to have the highest average price.]

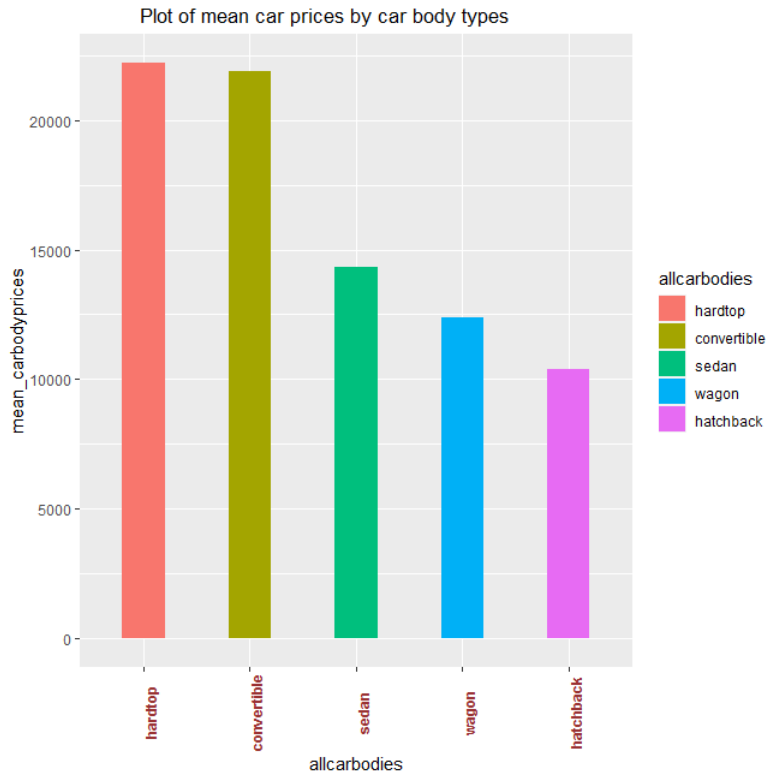
```
ggplot(df1, aes(x = allcars,y=mean_prices)) +
  geom_bar(
    aes(fill = allcars),
    stat = "identity", position = position_dodge(0.8),
    width = 0.4
  ) +theme(axis.text.x = element_text(face="bold", color="#993333", size=9, angle=90))+
  ggtitle("        Plot of mean car prices by Company names        ")
```



```
##saving data to new data frame "df2"
##using mutate function to group all car body.

df2<- car_prediction
 df2<- df2 %>%
 mutate( allcarbodies = factor(carbody)) %>%
 group_by(allcarbodies) %>%
 summarise(mean_carbodyprices= mean(price,na.rm=T)) %>%
 arrange(-mean_carbodyprices) %>%   # sort by counts
 mutate(allcarbodies = factor(allcarbodies, allcarbodies))
glimpse(df2)
```

```
Rows: 5
Columns: 2
$ allcarbodies        <fct> hardtop, convertible, sedan, wagon, hatchback
$ mean_carbodyprices <dbl> 22208.50, 21890.50, 14344.27, 12371.96, 1037~
```

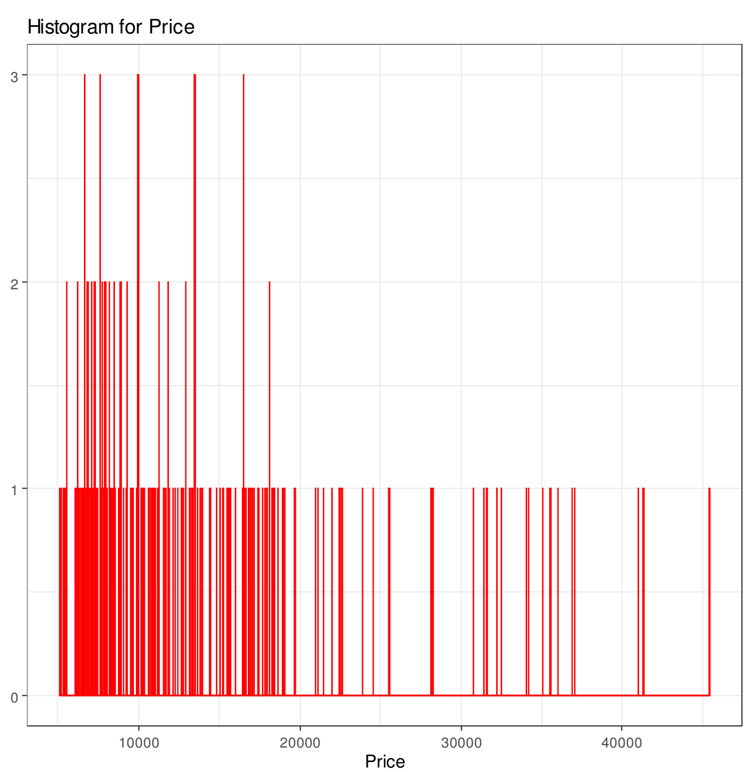
```
##Plotting  bar chart for mean price vs car bodies

ggplot(df2, aes(x = allcarbodies,y=mean_carbodyprices)) +
 geom_bar(
  aes(fill = allcarbodies),
  stat = "identity", position = position_dodge(0.8),
  width = 0.4
 ) +theme(axis.text.x = element_text(face="bold", color="#993333", size=9, angle=90))+
 ggtitle("        Plot of mean car prices by car body types       ")
```



[Inference: Hardtop and convertible seems to have the highest price]

##Plotting histogram for all cars prices

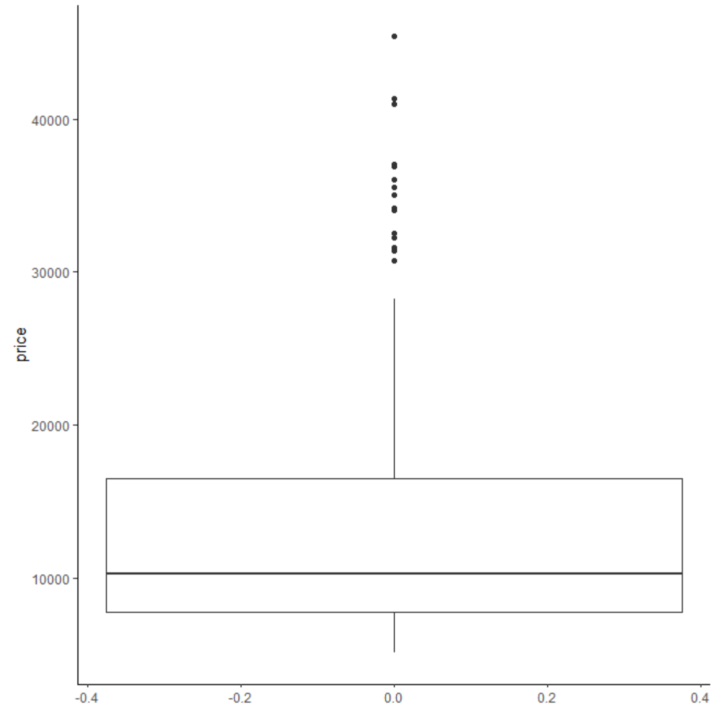
qplot(car_prediction$price, geom="histogram", binwidth=15, main="Histogram for Price", xlab="Price", fill=I("gray"), col=I("red"))+theme_bw()



[Inference: The price distribution of all the cars look to be right skewed.
It indicates that the prices mean prices range is 10000 to 15000 and most of the cars are less than 20000.]

##Plotting boxplot for all cars prices

```
ggplot(car_prediction, aes(y=price)) +
geom_boxplot()+ scale_fill_grey() +
theme_classic()
```



##Measureofcentre(mean,median,mode),measuresofdispersion(SD,CV),measuresof position (max, min, 25th and 75th percentiles)– in a single table:

```
stat.desc(car_prediction$price)
```

```
      nbr.val     nbr.null       nbr.na          min
2.050000e+02 0.000000e+00 0.000000e+00 5.118000e+03
          max        range          sum       median
4.540000e+04 4.028200e+04 2.721726e+06 1.029500e+04
         mean      SE.mean CI.mean.0.95          var
1.327671e+04 5.579656e+02 1.100119e+03 6.382176e+07
      std.dev     coef.var
7.988852e+03 6.017193e-01
```

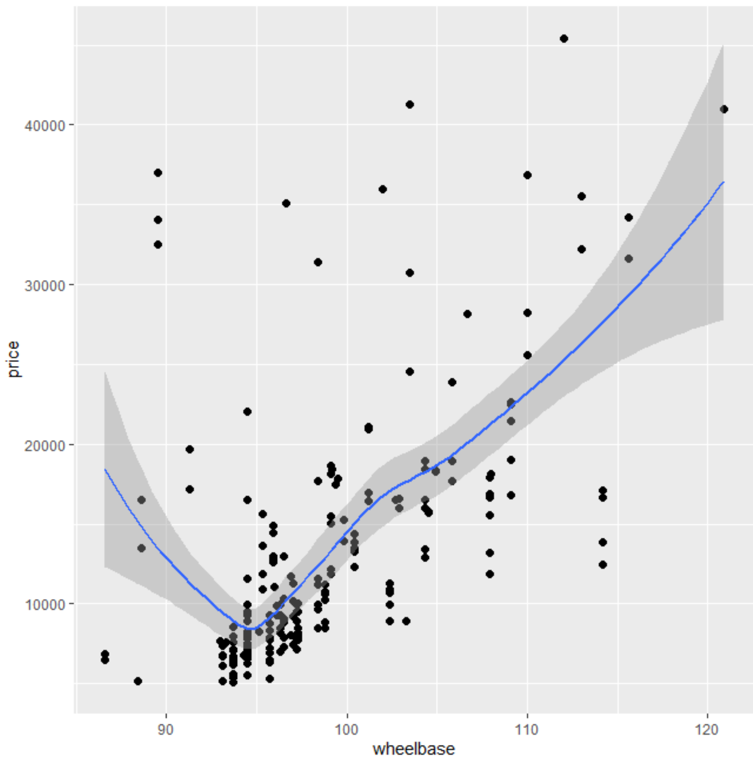
##Visualising numerical variables

```
for(i in 1:ncol(car_prediction)){
  if(class(car_prediction[,i])== "numeric")
    print(names(car_prediction)[i])
```

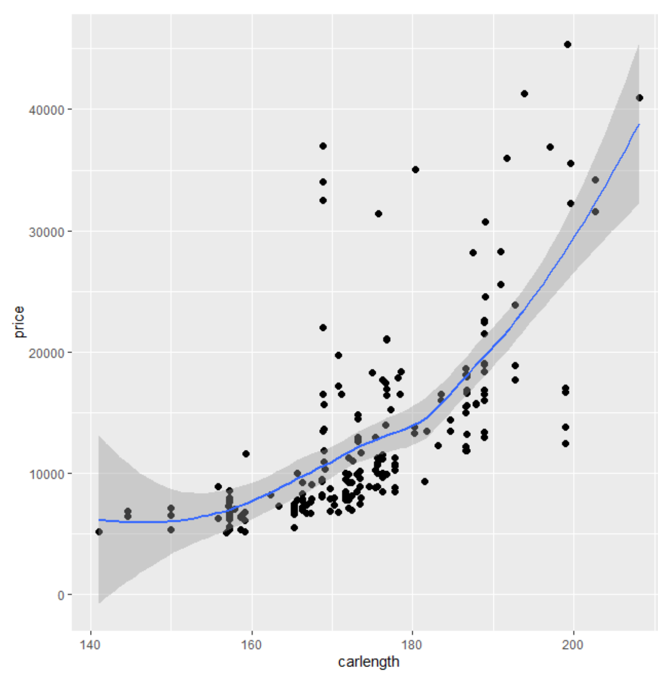
```
[1] "wheelbase"
[1] "carlength"
[1] "carwidth"
[1] "carheight"
[1] "boreratio"
[1] "stroke"
[1] "compressionratio"
[1] "price"
```

##Visualising numerical variables

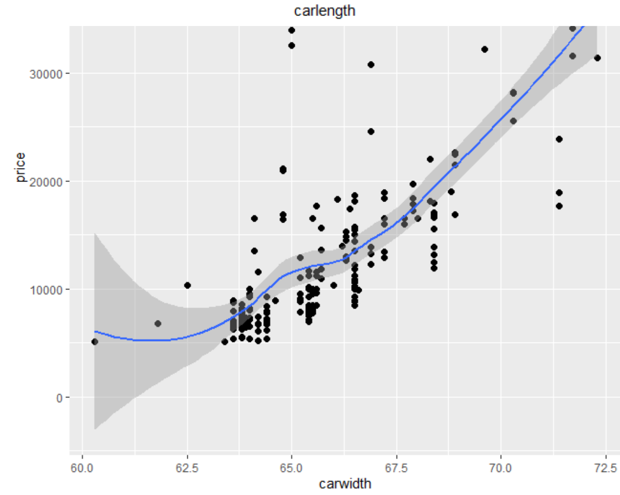
```
ggplot(car_prediction, aes(x=wheelbase, y=price))+geom_point(size=2)+
geom_smooth()
```



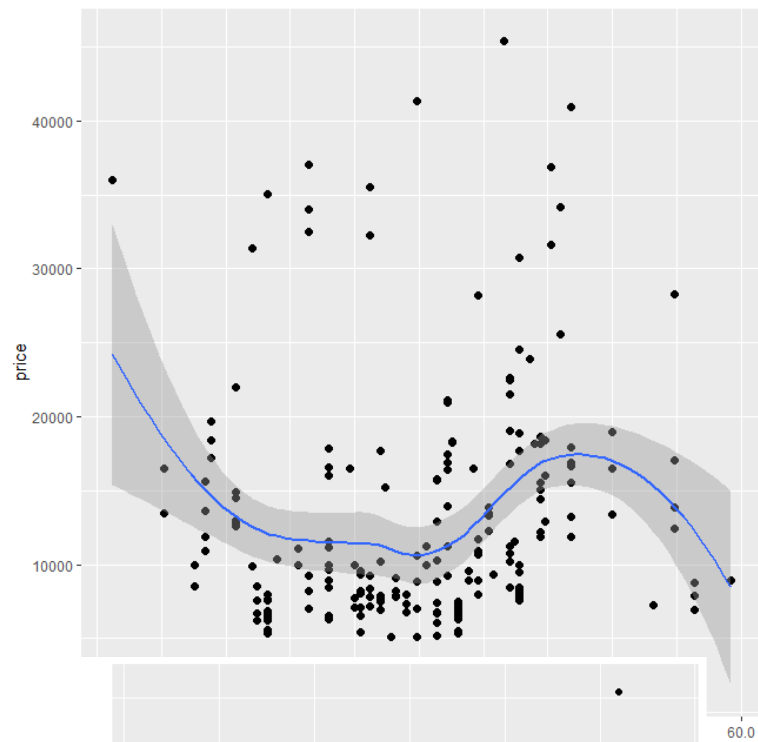

```
ggplot(car_prediction, aes(x=carlength, y=price)) +
  geom_point(size=2) +
  geom_smooth()
```



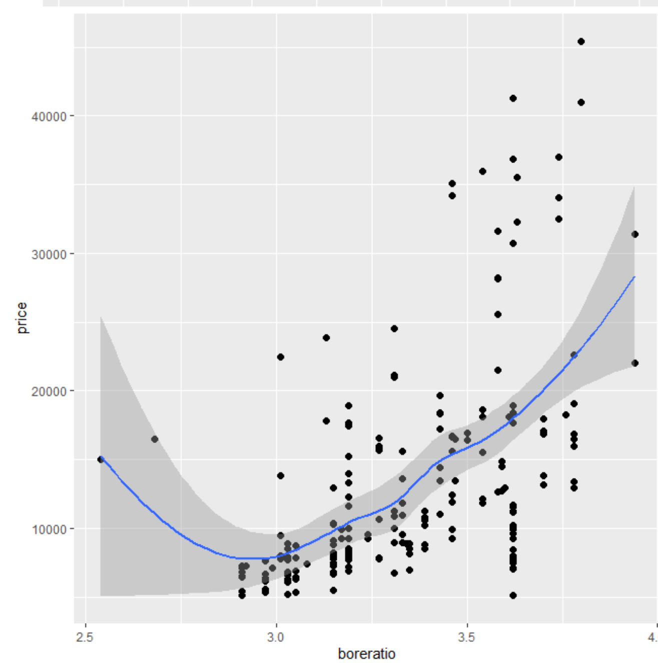

```
ggplot(car_prediction, aes(x=carwidth, y=price)) +
  geom_point(size=2) +
  geom_smooth()
```



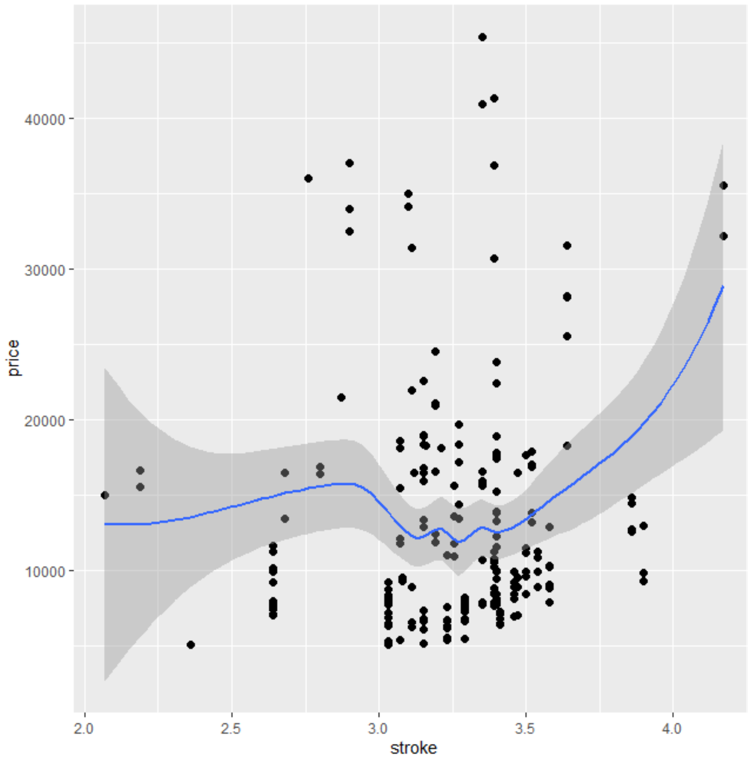
```
ggplot(car_prediction, aes(x=carheight, y=price)) +
  geom_point(size=2) +
  geom_smooth()
```



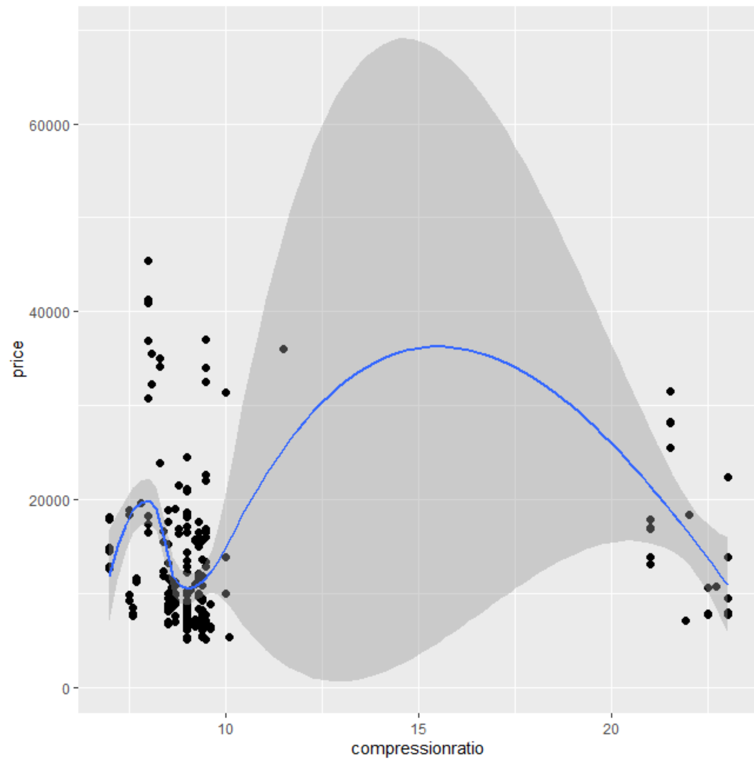

```
ggplot(car_prediction, aes(x=boreratio, y=price)) +
  geom_point(size=2) +
  geom_smooth()
```



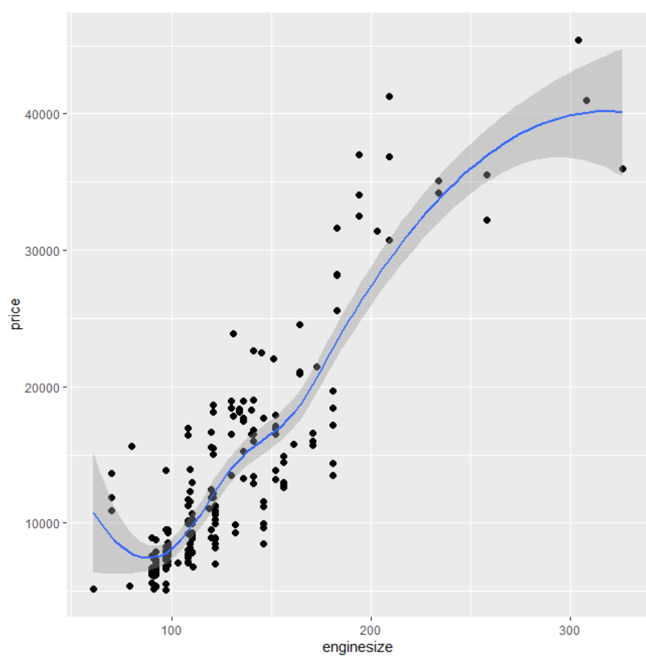
```
ggplot(car_prediction, aes(x=stroke, y=price)) +
  geom_point(size=2) +
  geom_smooth()
```



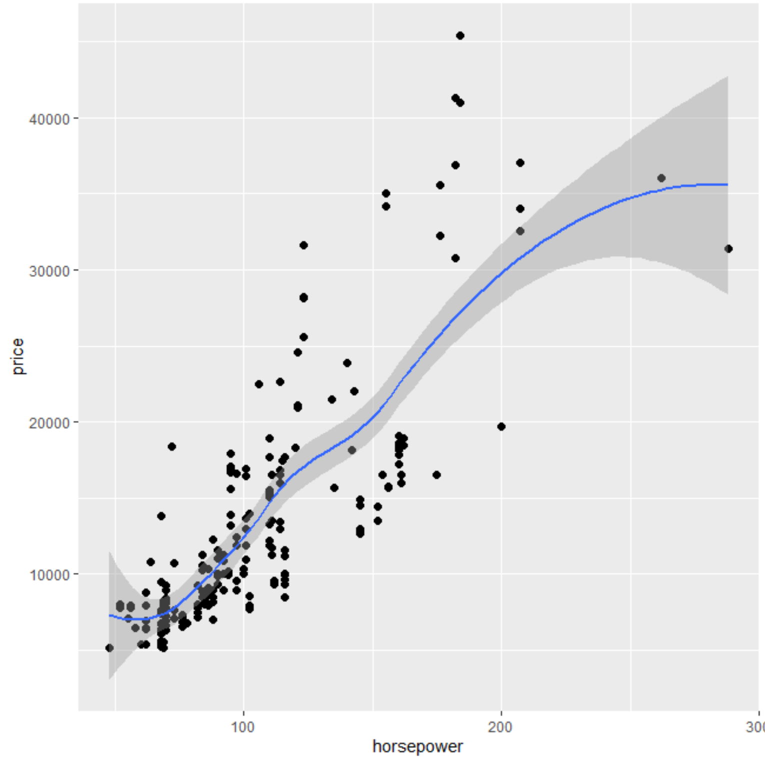

```
ggplot(car_prediction, aes(x=compressionratio, y=price)) +
  geom_point(size=2) +
  geom_smooth()
```



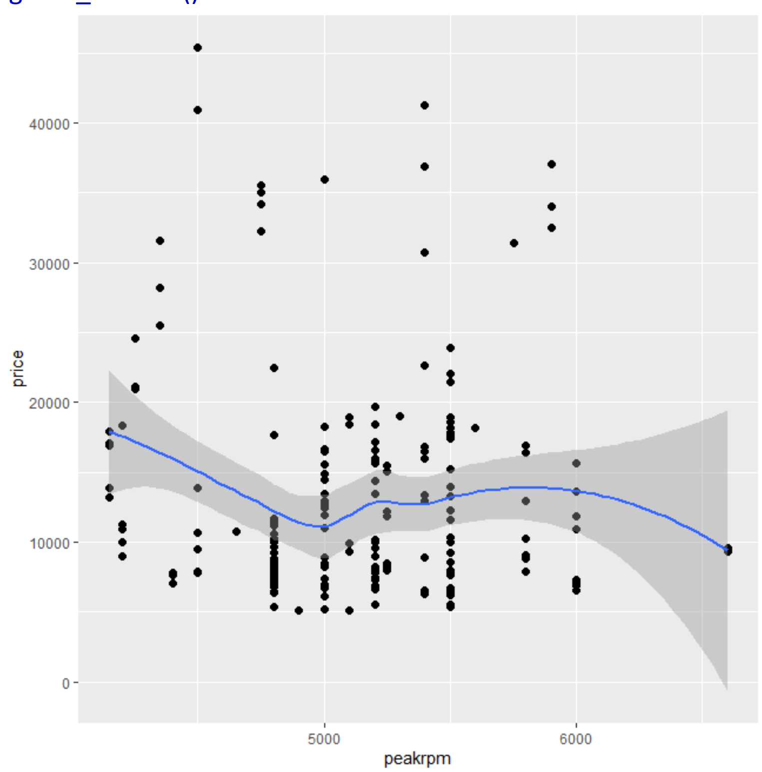

```
ggplot(car_prediction, aes(x=enginesize, y=price)) +  geom_point(size=2) +  geom_smooth()
```



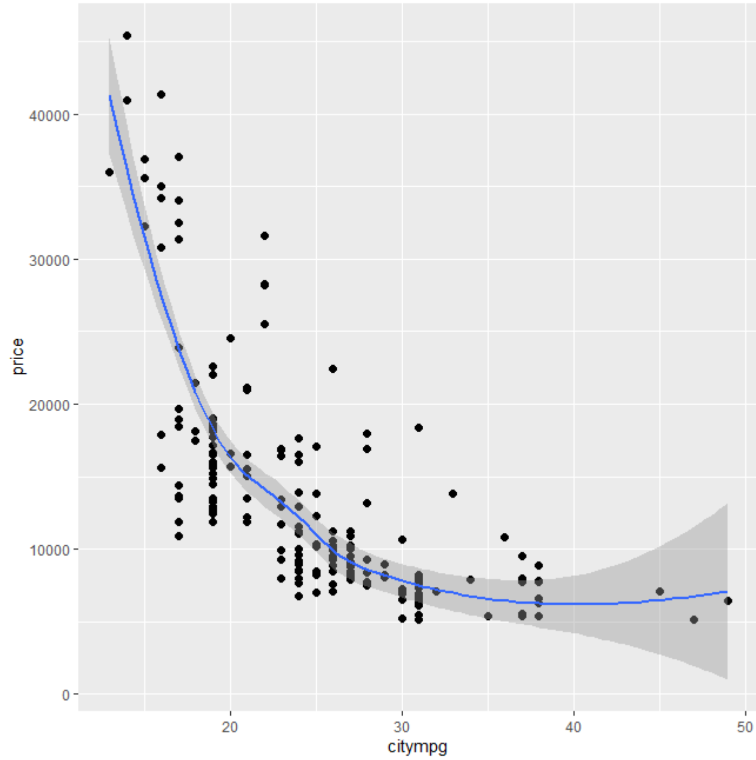
```
ggplot(car_prediction, aes(x=horsepower, y=price)) +
  geom_point(size=2) +
  geom_smooth()
```



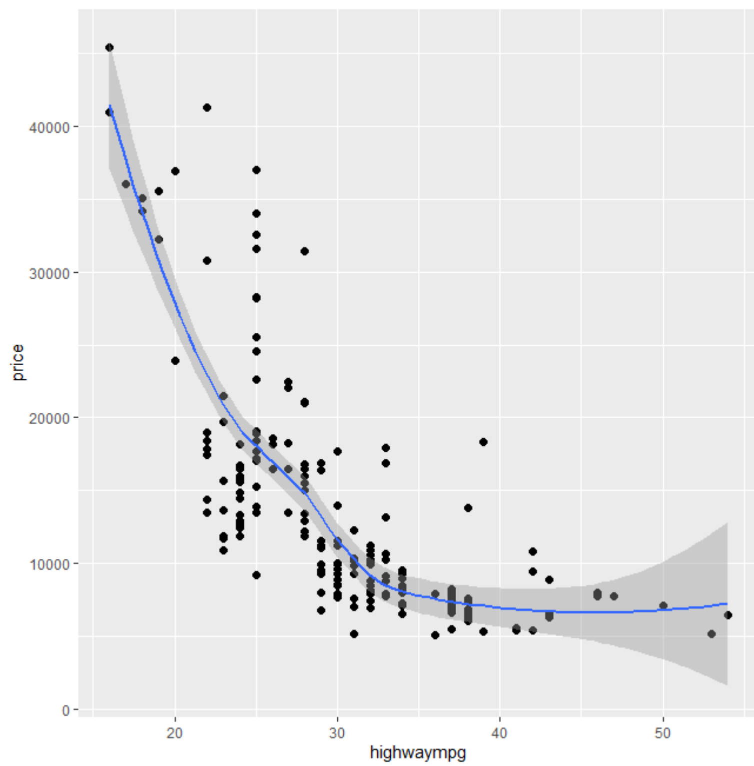

```
ggplot(car_prediction, aes(x=peakrpm, y=price)) +
  geom_point(size=2) +
  geom_smooth()
```



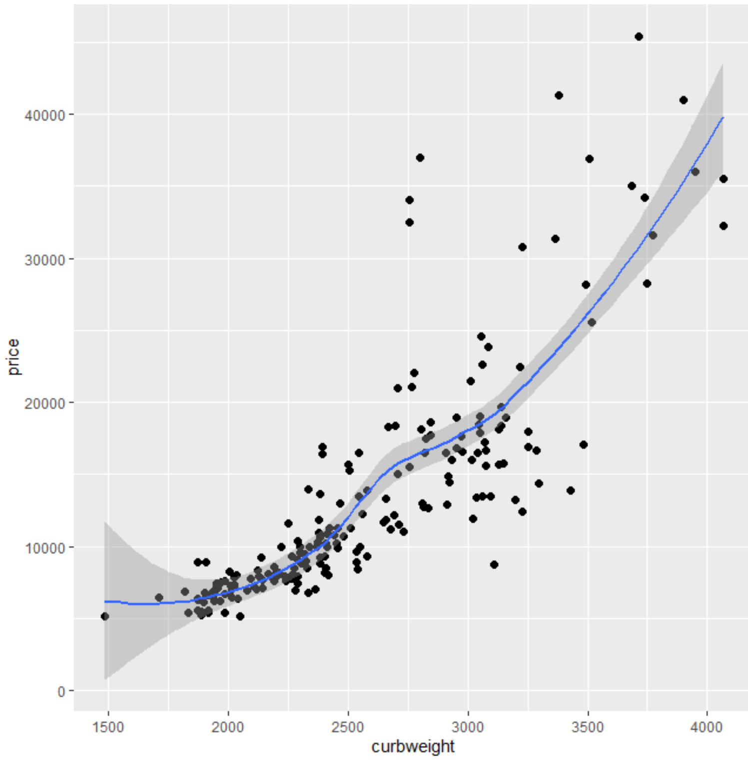
```
ggplot(car_prediction, aes(x=citympg, y=price)) +
  geom_point(size=2) +
  geom_smooth()
```




```
ggplot(car_prediction, aes(x=highwaympg, y=price)) +
  geom_point(size=2) +
  geom_smooth()
```



```
ggplot(car_prediction, aes(x=curbweight, y=price)) +
  geom_point(size=2) +
  geom_smooth()
```



##Creating Dummy Variables

```
library(dplyr)
car_prediction <- car_prediction
library(caret)
select(CarName, allcars)
dmy <- dummyVars( ~. ,data=car_prediction,fullRank =T)
mod1 <- data.frame(predict(dmy,newdata =car_prediction))
names(mod1)
```

##Splitting the Data into Training and Testing Sets

```
library(simEd)
set.seed(50)
k=nrow(mod1)
train_index = sample(1:k,round(0.9*k))
train_data = mod1[train_index,]
test_data = mod1[-train_index,]
nrow(train_data)
#184
nrow(test_data)
#21
```

```
names(train_data)
```

```
  [1] "car_ID"
  [2] "symboling"
  [3] "CarNamealfa.romero.Quadrifoglio"
  [4] "CarNamealfa.romero.stelvio"
  [5] "CarNameaudi.100.ls"
  [6] "CarNameaudi.100ls"
  [7] "CarNameaudi.4000"
  [8] "CarNameaudi.5000"
  [9] "CarNameaudi.5000s..diesel."
 [10] "CarNameaudi.fox"
 [11] "CarNamebmw.320i"
 [12] "CarNamebmw.x1"
 [13] "CarNamebmw.x3"
 [14] "CarNamebmw.x4"
 [15] "CarNamebmw.x5"
 [16] "CarNamebmw.z4"
 [17] "CarNamebuick.century"
 [18] "CarNamebuick.century.luxus..sw."
 [19] "CarNamebuick.century.special"
 [20] "CarNamebuick.electra.225.custom"
 [21] "CarNamebuick.opel.isuzu.deluxe"
 [22] "CarNamebuick.regal.sport.coupe..turbo."
 [23] "CarNamebuick.skyhawk"
 [24] "CarNamebuick.skylark"
 [25] "CarNamechevrolet.impala"
 [26] "CarNamechevrolet.monte.carlo"
 [27] "CarNamechevrolet.vega.2300"
 [28] "CarNamedodge.challenger.se"
 [29] "CarNamedodge.colt..sw."
 [30] "CarNamedodge.colt.hardtop"
 [31] "CarNamedodge.coronet.custom"
 [32] "CarNamedodge.coronet.custom..sw."
 [33] "CarNamedodge.d200"
```

```
[34] "CarNamedodge.dart.custom"
[35] "CarNamedodge.monaco..sw."
[36] "CarNamedodge.rampage"
[37] "CarNamehonda.accord"
[38] "CarNamehonda.accord.cvcc"
[39] "CarNamehonda.accord.lx"
[40] "CarNamehonda.civic"
[41] "CarNamehonda.civic..auto."
[42] "CarNamehonda.civic.1300"
[43] "CarNamehonda.civic.1500.gl"
[44] "CarNamehonda.civic.cvcc"
[45] "CarNamehonda.prelude"
[46] "CarNameisuzu.D.Max."
[47] "CarNameisuzu.D.Max.V.Cross"
[48] "CarNameisuzu.MU.X"
[49] "CarNamejaguar.xf"
[50] "CarNamejaguar.xj"
[51] "CarNamejaguar.xk"
[52] "CarNamemaxda.glc.deluxe"
[53] "CarNamemaxda.rx3"
[54] "CarNamemazda.626"
[55] "CarNamemazda.glc"
[56] "CarNamemazda.glc.4"
[57] "CarNamemazda.glc.custom"
[58] "CarNamemazda.glc.custom.l"
[59] "CarNamemazda.glc.deluxe"
[60] "CarNamemazda.rx.4"
[61] "CarNamemazda.rx.7.gs"
[62] "CarNamemazda.rx2.coupe"
[63] "CarNamemercury.cougar"
[64] "CarNamemitsubishi.g4"
[65] "CarNamemitsubishi.lancer"
[66] "CarNamemitsubishi.mirage"
[67] "CarNamemitsubishi.mirage.g4"
[68] "CarNamemitsubishi.montero"
[69] "CarNamemitsubishi.outlander"
[70] "CarNamemitsubishi.pajero"
[71] "CarNamenissan.clipper"
[72] "CarNamenissan.dayz"
[73] "CarNamenissan.fuga"
[74] "CarNamenissan.gt.r"
[75] "CarNamenissan.juke"
[76] "CarNamenissan.kicks"
[77] "CarNamenissan.latio"
[78] "CarNamenissan.leaf"
[79] "CarNamenissan.note"
[80] "CarNamenissan.nv200"
[81] "CarNamenissan.otti"
[82] "CarNamenissan.rogue"
[83] "CarNamenissan.teana"
[84] "CarNamenissan.titan"
[85] "CarNameNissan.versa"
[86] "CarNamepeugeot.304"
[87] "CarNamepeugeot.504"
[88] "CarNamepeugeot.504..sw."
[89] "CarNamepeugeot.505s.turbo.diesel"
[90] "CarNamepeugeot.604sl"
[91] "CarNameplymouth.cricket"
[92] "CarNameplymouth.duster"
[93] "CarNameplymouth.fury.gran.sedan"
[94] "CarNameplymouth.fury.iii"
[95] "CarNameplymouth.satellite.custom..sw."
```

```
 [96] "CarNameplymouth.valiant"
 [97] "CarNameporcshce.panamera"
 [98] "CarNameporsche.boxter"
 [99] "CarNameporsche.cayenne"
[100] "CarNameporsche.macan"
[101] "CarNamerenault.12tl"
[102] "CarNamerenault.5.gtl"
[103] "CarNamesaab.99e"
[104] "CarNamesaab.99gle"
[105] "CarNamesaab.99le"
[106] "CarNamesubaru"
[107] "CarNamesubaru.baja"
[108] "CarNamesubaru.brz"
[109] "CarNamesubaru.dl"
[110] "CarNamesubaru.r1"
[111] "CarNamesubaru.r2"
[112] "CarNamesubaru.trezia"
[113] "CarNamesubaru.tribeca"
[114] "CarNametoyota.carina"
[115] "CarNametoyota.celica.gt"
[116] "CarNametoyota.celica.gt.liftback"
[117] "CarNametoyota.corolla"
[118] "CarNametoyota.corolla.1200"
[119] "CarNametoyota.corolla.1600..sw."
[120] "CarNametoyota.corolla.liftback"
[121] "CarNametoyota.corolla.tercel"
[122] "CarNametoyota.corona"
[123] "CarNametoyota.corona.hardtop"
[124] "CarNametoyota.corona.liftback"
[125] "CarNametoyota.corona.mark.ii"
[126] "CarNametoyota.cressida"
[127] "CarNametoyota.mark.ii"
[128] "CarNametoyota.starlet"
[129] "CarNametoyota.tercel"
[130] "CarNametoyouta.tercel"
[131] "CarNamevokswagen.rabbit"
[132] "CarNamevolkswagen.1131.deluxe.sedan"
[133] "CarNamevolkswagen.411..sw."
[134] "CarNamevolkswagen.dasher"
[135] "CarNamevolkswagen.model.111"
[136] "CarNamevolkswagen.rabbit"
[137] "CarNamevolkswagen.rabbit.custom"
[138] "CarNamevolkswagen.super.beetle"
[139] "CarNamevolkswagen.type.3"
[140] "CarNamevolvo.144ea"
[141] "CarNamevolvo.145e..sw."
[142] "CarNamevolvo.244dl"
[143] "CarNamevolvo.245"
[144] "CarNamevolvo.246"
[145] "CarNamevolvo.264gl"
[146] "CarNamevolvo.diesel"
[147] "CarNamevw.dasher"
[148] "CarNamevw.rabbit"
[149] "fueltypegas"
[150] "aspirationturbo"
[151] "doornumbertwo"
[152] "carbodyhardtop"
[153] "carbodyhatchback"
[154] "carbodysedan"
[155] "carbodywagon"
[156] "drivewheelfwd"
[157] "drivewheelrwd"
```

```
[158] "enginelocationrear"
[159] "wheelbase"
[160] "carlength"
[161] "carwidth"
[162] "carheight"
[163] "curbweight"
[164] "enginetypedohcv"
[165] "enginetypel"
[166] "enginetypeohc"
[167] "enginetypeohcf"
[168] "enginetypeohcv"
[169] "enginetyperotor"
[170] "cylindernumberfive"
[171] "cylindernumberfour"
[172] "cylindernumbersix"
[173] "cylindernumberthree"
[174] "cylindernumbertwelve"
[175] "cylindernumbertwo"
[176] "enginesize"
[177] "fuelsystem2bbl"
[178] "fuelsystem4bbl"
[179] "fuelsystemidi"
[180] "fuelsystemmfi"
[181] "fuelsystemmpfi"
[182] "fuelsystemspdi"
[183] "fuelsystemspfi"
[184] "boreratio"
[185] "stroke"
[186] "compressionratio"
[187] "horsepower"
[188] "peakrpm"
[189] "citympg"
[190] "highwaympg"
[191] "price"
[192] "allcarsaudi"
[193] "allcarsbmw"
[194] "allcarsbuick"
[195] "allcarschevrolet"
[196] "allcarsdodge"
[197] "allcarshonda"
[198] "allcarsisuzu"
[199] "allcarsjaguar"
[200] "allcarsmazda"
[201] "allcarsmercury"
[202] "allcarsmitsubishi"
[203] "allcarsNissan"
[204] "allcarspeugeot"
[205] "allcarsporsche"
[206] "allcarsrenault"
[207] "allcarssaab"
[208] "allcarssubaru"
[209] "allcarstoyota"
[210] "allcarsvolkswagen"
[211] "allcarsvolvo"
```

```
names(test_data)
```

```
[1] "car_ID"
  [2] "symboling"
```

```
 [3] "CarNamealfa.romero.Quadrifoglio"
 [4] "CarNamealfa.romero.stelvio"
 [5] "CarNameaudi.100.ls"
 [6] "CarNameaudi.100ls"
 [7] "CarNameaudi.4000"
 [8] "CarNameaudi.5000"
 [9] "CarNameaudi.5000s..diesel."
[10] "CarNameaudi.fox"
[11] "CarNamebmw.320i"
[12] "CarNamebmw.x1"
[13] "CarNamebmw.x3"
[14] "CarNamebmw.x4"
[15] "CarNamebmw.x5"
[16] "CarNamebmw.z4"
[17] "CarNamebuick.century"
[18] "CarNamebuick.century.luxus..sw."
[19] "CarNamebuick.century.special"
[20] "CarNamebuick.electra.225.custom"
[21] "CarNamebuick.opel.isuzu.deluxe"
[22] "CarNamebuick.regal.sport.coupe..turbo."
[23] "CarNamebuick.skyhawk"
[24] "CarNamebuick.skylark"
[25] "CarNamechevrolet.impala"
[26] "CarNamechevrolet.monte.carlo"
[27] "CarNamechevrolet.vega.2300"
[28] "CarNamedodge.challenger.se"
[29] "CarNamedodge.colt..sw."
[30] "CarNamedodge.colt.hardtop"
[31] "CarNamedodge.coronet.custom"
[32] "CarNamedodge.coronet.custom..sw."
[33] "CarNamedodge.d200"
[34] "CarNamedodge.dart.custom"
[35] "CarNamedodge.monaco..sw."
[36] "CarNamedodge.rampage"
[37] "CarNamehonda.accord"
[38] "CarNamehonda.accord.cvcc"
[39] "CarNamehonda.accord.lx"
[40] "CarNamehonda.civic"
[41] "CarNamehonda.civic..auto."
[42] "CarNamehonda.civic.1300"
[43] "CarNamehonda.civic.1500.gl"
[44] "CarNamehonda.civic.cvcc"
[45] "CarNamehonda.prelude"
[46] "CarNameisuzu.D.Max."
[47] "CarNameisuzu.D.Max.V.Cross"
[48] "CarNameisuzu.MU.X"
[49] "CarNamejaguar.xf"
[50] "CarNamejaguar.xj"
[51] "CarNamejaguar.xk"
[52] "CarNamemaxda.glc.deluxe"
[53] "CarNamemaxda.rx3"
[54] "CarNamemazda.626"
[55] "CarNamemazda.glc"
[56] "CarNamemazda.glc.4"
[57] "CarNamemazda.glc.custom"
[58] "CarNamemazda.glc.custom.l"
[59] "CarNamemazda.glc.deluxe"
[60] "CarNamemazda.rx.4"
[61] "CarNamemazda.rx.7.gs"
[62] "CarNamemazda.rx2.coupe"
[63] "CarNamemercury.cougar"
[64] "CarNamemitsubishi.g4"
```

```
 [65] "CarNamemitsubishi.lancer"
 [66] "CarNamemitsubishi.mirage"
 [67] "CarNamemitsubishi.mirage.g4"
 [68] "CarNamemitsubishi.montero"
 [69] "CarNamemitsubishi.outlander"
 [70] "CarNamemitsubishi.pajero"
 [71] "CarNamenissan.clipper"
 [72] "CarNamenissan.dayz"
 [73] "CarNamenissan.fuga"
 [74] "CarNamenissan.gt.r"
 [75] "CarNamenissan.juke"
 [76] "CarNamenissan.kicks"
 [77] "CarNamenissan.latio"
 [78] "CarNamenissan.leaf"
 [79] "CarNamenissan.note"
 [80] "CarNamenissan.nv200"
 [81] "CarNamenissan.otti"
 [82] "CarNamenissan.rogue"
 [83] "CarNamenissan.teana"
 [84] "CarNamenissan.titan"
 [85] "CarNameNissan.versa"
 [86] "CarNamepeugeot.304"
 [87] "CarNamepeugeot.504"
 [88] "CarNamepeugeot.504..sw."
 [89] "CarNamepeugeot.505s.turbo.diesel"
 [90] "CarNamepeugeot.604sl"
 [91] "CarNameplymouth.cricket"
 [92] "CarNameplymouth.duster"
 [93] "CarNameplymouth.fury.gran.sedan"
 [94] "CarNameplymouth.fury.iii"
 [95] "CarNameplymouth.satellite.custom..sw."
 [96] "CarNameplymouth.valiant"
 [97] "CarNameporcshce.panamera"
 [98] "CarNameporsche.boxter"
 [99] "CarNameporsche.cayenne"
[100] "CarNameporsche.macan"
[101] "CarNamerenault.12tl"
[102] "CarNamerenault.5.gtl"
[103] "CarNamesaab.99e"
[104] "CarNamesaab.99gle"
[105] "CarNamesaab.99le"
[106] "CarNamesubaru"
[107] "CarNamesubaru.baja"
[108] "CarNamesubaru.brz"
[109] "CarNamesubaru.dl"
[110] "CarNamesubaru.r1"
[111] "CarNamesubaru.r2"
[112] "CarNamesubaru.trezia"
[113] "CarNamesubaru.tribeca"
[114] "CarNametoyota.carina"
[115] "CarNametoyota.celica.gt"
[116] "CarNametoyota.celica.gt.liftback"
[117] "CarNametoyota.corolla"
[118] "CarNametoyota.corolla.1200"
[119] "CarNametoyota.corolla.1600..sw."
[120] "CarNametoyota.corolla.liftback"
[121] "CarNametoyota.corolla.tercel"
[122] "CarNametoyota.corona"
[123] "CarNametoyota.corona.hardtop"
[124] "CarNametoyota.corona.liftback"
[125] "CarNametoyota.corona.mark.ii"
[126] "CarNametoyota.cressida"
```

```
[127] "CarNametoyota.mark.ii"
[128] "CarNametoyota.starlet"
[129] "CarNametoyota.tercel"
[130] "CarNametoyouta.tercel"
[131] "CarNamevokswagen.rabbit"
[132] "CarNamevolkswagen.1131.deluxe.sedan"
[133] "CarNamevolkswagen.411..sw."
[134] "CarNamevolkswagen.dasher"
[135] "CarNamevolkswagen.model.111"
[136] "CarNamevolkswagen.rabbit"
[137] "CarNamevolkswagen.rabbit.custom"
[138] "CarNamevolkswagen.super.beetle"
[139] "CarNamevolkswagen.type.3"
[140] "CarNamevolvo.144ea"
[141] "CarNamevolvo.145e..sw."
[142] "CarNamevolvo.244dl"
[143] "CarNamevolvo.245"
[144] "CarNamevolvo.246"
[145] "CarNamevolvo.264gl"
[146] "CarNamevolvo.diesel"
[147] "CarNamevw.dasher"
[148] "CarNamevw.rabbit"
[149] "fueltypegas"
[150] "aspirationturbo"
[151] "doornumbertwo"
[152] "carbodyhardtop"
[153] "carbodyhatchback"
[154] "carbodysedan"
[155] "carbodywagon"
[156] "drivewheelfwd"
[157] "drivewheelrwd"
[158] "enginelocationrear"
[159] "wheelbase"
[160] "carlength"
[161] "carwidth"
[162] "carheight"
[163] "curbweight"
[164] "enginetypedohcv"
[165] "enginetypel"
[166] "enginetypeohc"
[167] "enginetypeohcf"
[168] "enginetypeohcv"
[169] "enginetyperotor"
[170] "cylindernumberfive"
[171] "cylindernumberfour"
[172] "cylindernumbersix"
[173] "cylindernumberthree"
[174] "cylindernumbertwelve"
[175] "cylindernumbertwo"
[176] "enginesize"
[177] "fuelsystem2bbl"
[178] "fuelsystem4bbl"
[179] "fuelsystemidi"
[180] "fuelsystemmfi"
[181] "fuelsystemmpfi"
[182] "fuelsystemspdi"
[183] "fuelsystemspfi"
[184] "boreratio"
[185] "stroke"
[186] "compressionratio"
[187] "horsepower"
[188] "peakrpm"
```

```
[189] "citympg"
[190] "highwaympg"
[191] "price"
[192] "allcarsaudi"
[193] "allcarsbmw"
[194] "allcarsbuick"
[195] "allcarschevrolet"
[196] "allcarsdodge"
[197] "allcarshonda"
[198] "allcarsisuzu"
[199] "allcarsjaguar"
[200] "allcarsmazda"
[201] "allcarsmercury"
[202] "allcarsmitsubishi"
[203] "allcarsNissan"
[204] "allcarspeugeot"
[205] "allcarsporsche"
[206] "allcarsrenault"
[207] "allcarssaab"
[208] "allcarssubaru"
[209] "allcarstoyota"
[210] "allcarsvolkswagen"
[211] "allcarsvolvo"
```

##We have considered 90% as train data and 10% as test data.
##Applying the linear Regression model

```
fit1=lm(price~as.factor(carheight)+as.factor(stroke)+as.factor(compressionratio)+as.factor(symboling)+as.factor(peakrpm),data=train_data)

summary(fit1)
```

```
Call:
lm(formula = price ~ as.factor(carheight) + as.factor(stroke) + 
    as.factor(compressionratio) + as.factor(symboling) + as.factor(peakrpm), 
    data = train_data)

Residuals:
    Min      1Q  Median      3Q     Max 
-3326.2  -343.3     0.0   306.3  3326.2 

Coefficients: (27 not defined because of singularities)
                                Estimate Std. Error t value Pr(>|t|)    
(Intercept)                      18528.8    17698.2   1.047 0.298495    
as.factor(carheight)48.8          -451.0     4192.8  -0.108 0.914620    
as.factor(carheight)49.4        -19049.3     4249.9  -4.482 2.61e-05 ***
as.factor(carheight)49.6         29206.1     7550.8   3.868 0.000232 ***
as.factor(carheight)49.7         24522.7    20261.0   1.210 0.229949    
as.factor(carheight)50.2         12828.5    16930.8   0.758 0.451003    
as.factor(carheight)50.5         36230.4     8482.2   4.271 5.63e-05 ***
as.factor(carheight)50.6         24451.1    19361.1   1.263 0.210538    
as.factor(carheight)50.8         24556.1    19276.6   1.274 0.206639    
as.factor(carheight)51            3287.9    15495.4   0.212 0.832540    
as.factor(carheight)51.4         17586.8    20302.7   0.866 0.389127    
as.factor(carheight)51.6        -21409.7     4006.6  -5.344 9.44e-07 ***
as.factor(carheight)52           17075.8    19843.3   0.861 0.392237    
as.factor(carheight)52.4         11594.3    23785.9   0.487 0.627366    
```

```
as.factor(carheight)52.5        24341.8    19387.7   1.256 0.213184    
as.factor(carheight)52.6        23617.5    19405.8   1.217 0.227408    
as.factor(carheight)52.8        25324.9    19702.4   1.285 0.202619    
as.factor(carheight)53          24579.8    19768.1   1.243 0.217589    
as.factor(carheight)53.1        22468.6    20371.5   1.103 0.273580    
as.factor(carheight)53.2       -16919.5    29070.9  -0.582 0.562307    
as.factor(carheight)53.3        23296.5    18928.4   1.231 0.222254    
as.factor(carheight)53.5        22461.6    19232.1   1.168 0.246538    
as.factor(carheight)53.7        30666.0    20686.2   1.482 0.142413    
as.factor(carheight)53.9        25739.0    19868.5   1.295 0.199132    
as.factor(carheight)54.1        26096.5    18486.1   1.412 0.162178    
as.factor(carheight)54.3        25528.1    20646.4   1.236 0.220153    
as.factor(carheight)54.4        16636.1    19933.1   0.835 0.406597    
as.factor(carheight)54.5        21546.6    19264.3   1.118 0.266936    
as.factor(carheight)54.7        31669.8    20210.5   1.567 0.121326    
as.factor(carheight)54.8       -11605.5     4179.5  -2.777 0.006930 ** 
as.factor(carheight)54.9        26019.0    19774.6   1.316 0.192255    
as.factor(carheight)55.1        21176.3    20725.2   1.022 0.310174    
as.factor(carheight)55.2        36293.2     8836.1   4.107 0.000101 ***
as.factor(carheight)55.4        27890.4    19132.8   1.458 0.149091    
as.factor(carheight)55.5        34629.0    21159.6   1.637 0.105911    
as.factor(carheight)55.6        19201.8    20302.7   0.946 0.347300    
as.factor(carheight)55.7        24563.7    20583.6   1.193 0.236489    
as.factor(carheight)55.9        21459.0    19431.3   1.104 0.272971    
as.factor(carheight)56.1        22076.3    20605.0   1.071 0.287423    
as.factor(carheight)56.2        31539.3    20989.0   1.503 0.137127    
as.factor(carheight)56.3        27841.1    20135.5   1.383 0.170865    
as.factor(carheight)56.5        21420.4    19799.3   1.082 0.282775    
as.factor(carheight)56.7        25083.0    19498.1   1.286 0.202249    
as.factor(carheight)57.5        29863.6    21417.1   1.394 0.167319    
as.factor(carheight)58.7        24078.2    19600.2   1.228 0.223112    
as.factor(carheight)59.1        25899.2    19721.3   1.313 0.193098    
as.factor(carheight)59.8       -20876.2     4605.7  -4.533 2.16e-05 ***
as.factor(stroke)2.19            1180.1     4938.2   0.239 0.811772    
as.factor(stroke)2.36          -23639.1     5005.4  -4.723 1.06e-05 ***
as.factor(stroke)2.64            8299.3    19677.4   0.422 0.674399    
as.factor(stroke)2.68                NA         NA      NA       NA    
as.factor(stroke)2.76                NA         NA      NA       NA    
as.factor(stroke)2.8           -27436.2     5543.0  -4.950 4.45e-06 ***
as.factor(stroke)2.87          -33860.1     5565.2  -6.084 4.56e-08 ***
as.factor(stroke)2.9            36550.9     5340.0   6.845 1.81e-09 ***
as.factor(stroke)3.03            7884.1    19001.6   0.415 0.679388    
as.factor(stroke)3.07           -1588.2     2459.1  -0.646 0.520340    
as.factor(stroke)3.08             836.9    19390.1   0.043 0.965689    
as.factor(stroke)3.1            -7990.7     4987.7  -1.602 0.113345    
as.factor(stroke)3.11          -10197.3    21861.6  -0.466 0.642246    
as.factor(stroke)3.12                NA         NA      NA       NA    
as.factor(stroke)3.15          -36468.2    15563.0  -2.343 0.021768 *  
as.factor(stroke)3.16          -24831.9    18335.8  -1.354 0.179712    
as.factor(stroke)3.19            -965.7     4708.5  -0.205 0.838043    
as.factor(stroke)3.23           -2832.6    19254.6  -0.147 0.883438    
as.factor(stroke)3.255               NA         NA      NA       NA    
as.factor(stroke)3.27           15003.8    21229.8   0.707 0.481922    
as.factor(stroke)3.29           20358.2    40255.6   0.506 0.614535    
as.factor(stroke)3.35           29385.8    20287.7   1.448 0.151659    
as.factor(stroke)3.39            -102.0     4786.3  -0.021 0.983062    
as.factor(stroke)3.4           -17470.9     4704.2  -3.714 0.000390 ***
as.factor(stroke)3.41            3712.7     5990.2   0.620 0.537266    
as.factor(stroke)3.46                NA         NA      NA       NA    
as.factor(stroke)3.47          -11900.9    21185.1  -0.562 0.575956    
as.factor(stroke)3.5            25208.7    19501.0   1.293 0.200087    
as.factor(stroke)3.52          -27138.3     4969.7  -5.461 5.90e-07 ***
```

```
as.factor(stroke)3.54           -32312.4    12744.4  -2.535 0.013314 *
as.factor(stroke)3.58           -12829.7     6061.7  -2.117 0.037617 *
as.factor(stroke)3.64           -16546.3     4817.9  -3.434 0.000971 ***
as.factor(stroke)3.86           -35415.3    16186.5  -2.188 0.031788 *
as.factor(stroke)3.9            -36139.0    16090.4  -2.246 0.027648 *
as.factor(compressionratio)7.5   10418.0    17627.8   0.591 0.556299
as.factor(compressionratio)7.6  -36274.5     4307.0  -8.422 1.87e-12 ***
as.factor(compressionratio)7.7  -18389.1     4979.7  -3.693 0.000419 ***
as.factor(compressionratio)7.8  -18246.9     4895.1  -3.728 0.000373 ***
as.factor(compressionratio)8     -9524.2     3651.2  -2.608 0.010971 *
as.factor(compressionratio)8.3        NA         NA      NA       NA
as.factor(compressionratio)8.4  -46125.7    20357.1  -2.266 0.026348 *
as.factor(compressionratio)8.5   -8664.7     3460.9  -2.504 0.014469 *
as.factor(compressionratio)8.6  -18950.1    23489.2  -0.807 0.422357
as.factor(compressionratio)8.7   -3524.7    10802.7  -0.326 0.745119
as.factor(compressionratio)8.8        NA         NA      NA       NA
as.factor(compressionratio)9    -20826.0     4201.6  -4.957 4.33e-06 ***
as.factor(compressionratio)9.1        NA         NA      NA       NA
as.factor(compressionratio)9.2  -40369.0     6677.4  -6.046 5.36e-08 ***
as.factor(compressionratio)9.3  -28603.0     6356.7  -4.500 2.44e-05 ***
as.factor(compressionratio)9.31 -30204.7     6995.6  -4.318 4.76e-05 ***
as.factor(compressionratio)9.4  -34923.2    19047.8  -1.833 0.070703 .
as.factor(compressionratio)9.5     858.1    12485.8   0.069 0.945392
as.factor(compressionratio)9.6  -18273.3    22076.8  -0.828 0.410458
as.factor(compressionratio)10   -14519.5     3407.0  -4.262 5.83e-05 ***
as.factor(compressionratio)10.1 -36517.1     4856.9  -7.519 9.79e-11 ***
as.factor(compressionratio)11.5       NA         NA      NA       NA
as.factor(compressionratio)21         NA         NA      NA       NA
as.factor(compressionratio)21.5       NA         NA      NA       NA
as.factor(compressionratio)21.9       NA         NA      NA       NA
as.factor(compressionratio)22         NA         NA      NA       NA
as.factor(compressionratio)22.5 -42735.1     5091.8  -8.393 2.13e-12 ***
as.factor(compressionratio)22.7 -41986.3     4771.9  -8.799 3.60e-13 ***
as.factor(compressionratio)23     5724.0    22882.6   0.250 0.803158
as.factor(symboling)-1            2187.3     2140.0   1.022 0.310005
as.factor(symboling)0             -274.5     1843.5  -0.149 0.882015
as.factor(symboling)1             1358.1     1556.9   0.872 0.385823
as.factor(symboling)2              388.2     1524.4   0.255 0.799663
as.factor(symboling)3                 NA         NA      NA       NA
as.factor(peakrpm)4200                NA         NA      NA       NA
as.factor(peakrpm)4250                NA         NA      NA       NA
as.factor(peakrpm)4350                NA         NA      NA       NA
as.factor(peakrpm)4400          -43978.4    11269.0  -3.903 0.000206 ***
as.factor(peakrpm)4500          -22238.8    21114.6  -1.053 0.295612
as.factor(peakrpm)4650                NA         NA      NA       NA
as.factor(peakrpm)4750                NA         NA      NA       NA
as.factor(peakrpm)4800          -22433.5    21013.9  -1.068 0.289144
as.factor(peakrpm)4900                NA         NA      NA       NA
as.factor(peakrpm)5000           17745.8    18292.8   0.970 0.335118
as.factor(peakrpm)5100           -5588.7     5733.9  -0.975 0.332853
as.factor(peakrpm)5200          -20109.4    21783.3  -0.923 0.358886
as.factor(peakrpm)5250            3037.9     3264.8   0.930 0.355104
as.factor(peakrpm)5300            3692.8     4317.1   0.855 0.395060
as.factor(peakrpm)5400                NA         NA      NA       NA
as.factor(peakrpm)5500                NA         NA      NA       NA
as.factor(peakrpm)5750                NA         NA      NA       NA
as.factor(peakrpm)5800                NA         NA      NA       NA
as.factor(peakrpm)5900                NA         NA      NA       NA
as.factor(peakrpm)6000                NA         NA      NA       NA
as.factor(peakrpm)6600                NA         NA      NA       NA
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
Residual standard error: 1576 on 75 degrees of freedom
Multiple R-squared:  0.9841,  Adjusted R-squared:  0.9613
F-statistic: 43.04 on 108 and 75 DF,  p-value: < 2.2e-16
```

[Inference:

- **The R2 is 0.9841, that is, 98.41% of the total variability in the response variable, 'Price' is being explained by the full MLR model.**
- **The adjusted R2 is 0.9613; 96.13% of the variability in Price is explained by the full MLR model after being adjusted for redundant predictors.**
- **A low difference between R2 and adjusted R2 suggests that not many redundant predictors are present in the model. Based on the overall model parameters, we can also comment that the model fit is highly adequate.**
- **The p-value generated on testing for the overall model significance comes out to be less than 2.2×10^{-16}. Hence, we can confidently say the full MLR model is significant at a 5% level of significance]**

##Overall model significance:

F stat is distributed with an F-distribution with 43.04on 108 variables and have 75 Degree of freedom.

##Hypothesis condition

Null hypothesis: Ho- we reject the null hypothesis that all the Betas are 0s, except the one related to the intercept.

H1- we can not reject the null hypothesis any of the Beta is non zero.

[Inference: The full model is significant at 5% level of significance.]

##Prediction on test data

testfit=lm(price~as.factor(carheight)+as.factor(stroke)+as.factor(compressionratio)+as.factor(symboling)+as.factor(peakrpm),as.factor(wheelbase),data=test_data)

summary(testfit)

```
Call:
lm(formula = price ~ as.factor(carheight) + as.factor(stroke) + 
    as.factor(compressionratio) + as.factor(symboling) + as.factor(peakrpm), 
    data = test_data, subset = as.factor(wheelbase))

Residuals:
   Min     1Q Median     3Q    Max 
 -1650      0      0      0   1650 

Coefficients: (27 not defined because of singularities)
                                Estimate Std. Error t value Pr(>|t|)    
(Intercept)                      13645.0      777.8  17.543 2.88e-08 ***
as.factor(carheight)50.6         -6036.0     1100.0  -5.487 0.000387 ***
as.factor(carheight)50.8         -8073.0      952.6  -8.474 1.39e-05 ***
as.factor(carheight)52.5         -6519.0      952.6  -6.843 7.53e-05 ***
as.factor(carheight)52.8         20255.0      952.6  21.262 5.29e-09 ***
as.factor(carheight)53.3         -4550.0      952.6  -4.776 0.001007 ** 
as.factor(carheight)54.5         -6350.0      952.6  -6.666 9.21e-05 ***
as.factor(carheight)55.1          -146.0     1100.0  -0.133 0.897329    
as.factor(carheight)55.5         -5150.0      898.1  -5.734 0.000282 ***
as.factor(carheight)56            4505.0     1100.0   4.095 0.002695 ** 
as.factor(carheight)58.3         -6350.0      898.1  -7.070 5.86e-05 ***
as.factor(carheight)59.8         -4724.0     1100.0  -4.295 0.002007 ** 
as.factor(stroke)3.21                 NA         NA      NA       NA    
as.factor(stroke)3.23                 NA         NA      NA       NA    
as.factor(stroke)3.255                NA         NA      NA       NA    
as.factor(stroke)3.27                 NA         NA      NA       NA    
as.factor(stroke)3.39                 NA         NA      NA       NA    
as.factor(stroke)3.41                 NA         NA      NA       NA    
as.factor(stroke)3.46                 NA         NA      NA       NA    
as.factor(stroke)3.58                 NA         NA      NA       NA    
as.factor(stroke)4.17                 NA         NA      NA       NA    
as.factor(compressionratio)8.1        NA         NA      NA       NA    
as.factor(compressionratio)8.5        NA         NA      NA       NA    
as.factor(compressionratio)8.6        NA         NA      NA       NA    
```

```
as.factor(compressionratio)9          NA          NA       NA         NA
as.factor(compressionratio)9.2        NA          NA       NA         NA
as.factor(compressionratio)9.4        NA          NA       NA         NA
as.factor(compressionratio)9.41       NA          NA       NA         NA
as.factor(compressionratio)9.5        NA          NA       NA         NA
as.factor(symboling)0                 NA          NA       NA         NA
as.factor(symboling)1                 NA          NA       NA         NA
as.factor(symboling)3                 NA          NA       NA         NA
as.factor(peakrpm)4800                NA          NA       NA         NA
as.factor(peakrpm)5000                NA          NA       NA         NA
as.factor(peakrpm)5200                NA          NA       NA         NA
as.factor(peakrpm)5500                NA          NA       NA         NA
as.factor(peakrpm)5600                NA          NA       NA         NA
as.factor(peakrpm)5800                NA          NA       NA         NA
as.factor(peakrpm)6000                NA          NA       NA         NA
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 777.8 on 9 degrees of freedom
Multiple R-squared:  0.9959,  Adjusted R-squared:  0.9908
F-statistic: 197.1 on 11 and 9 DF,  p-value: 2.448e-09
```

[Inference:

- **The R2 is 0.99.59, that is, 99.59% of the total variability in the response variable, 'Price' is being explained by the full MLR model.**
- **The adjusted R2 is 0.9908; 99.08% of the variability in Price is explained by the full MLR model after being adjusted for redundant predictors.**
- **A low difference between R2 and adjusted R2 suggests that not many redundant predictors are present in the model. Based on the overall model parameters, we can also comment that the model fit is highly adequate.**
- **The p-value generated on testing for the overall model significance comes out to be less than 2.44×10−09. Hence, we can confidently say the full MLR model is significant at a 5% level of significance]**

##Prediction on test data

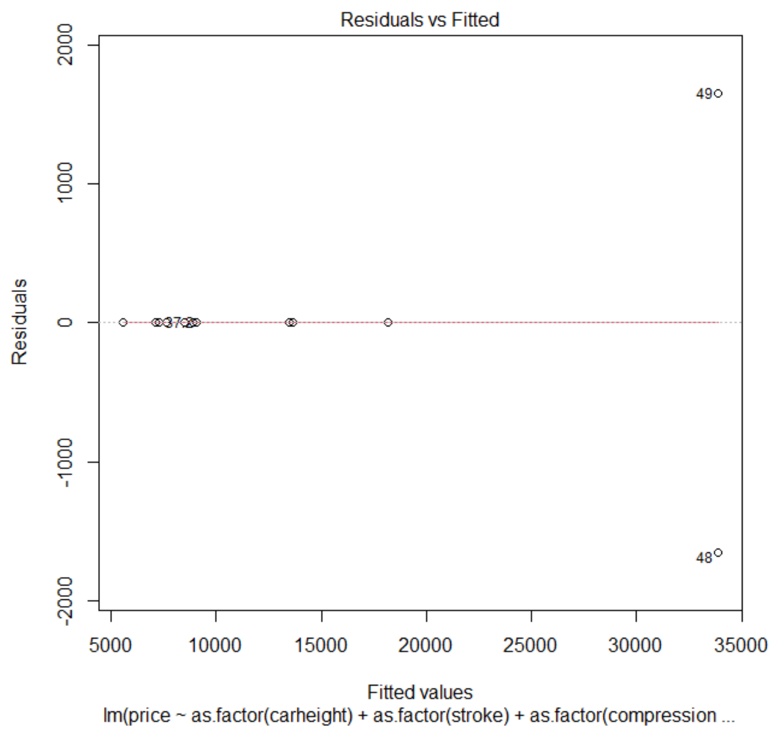
```
mean(test_data$price)
## 11994.43
rss <- sum((test_data$predicted_price - test_data$price) ^ 2)
##0
tss <- sum((test_data$price - mean(test_data$price)) ^ 2)
##1255570543.14286
r2 <- 1-(rss/tss) *100
##1
```

[Inference: Which means the R2 for train data is 0.9951 and 1 which is very close to each other.]

#Performance of Model :Evaluation

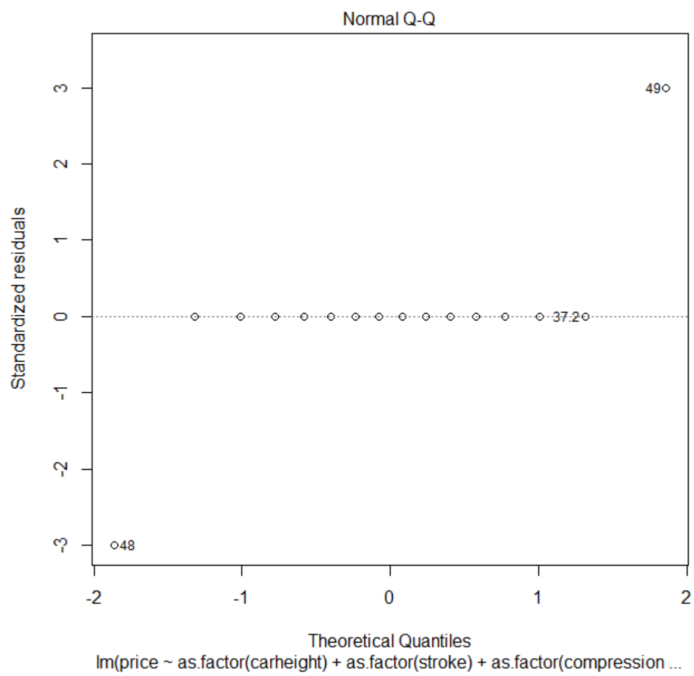
#1.Linearity between X and Y

```
plot(testfit,1)
```

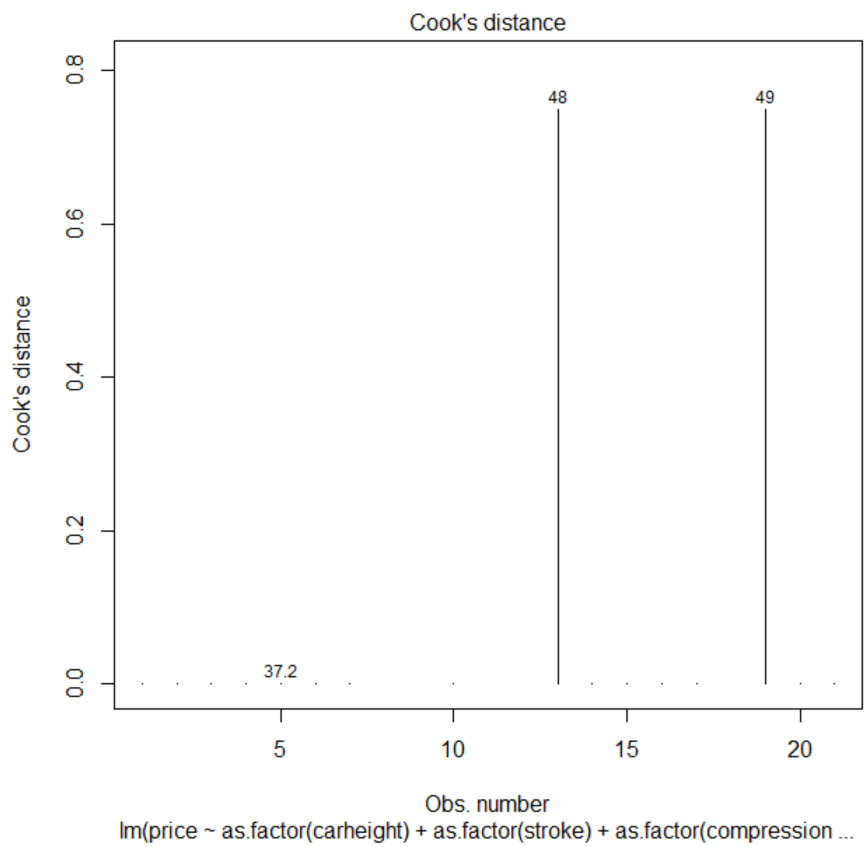


#Normality check

plot(testfit,2)



#Cooks Distance v/s Obs.number



#Final Model Conclusion :

Hence, We can say that our model is good enough to predict
#the Car prices using below predictor variables:
#Equation of Line to predict the Car prices values:
 # carheight
 # compression ratio
 # stroke
 # peakrpm
##
#Residual standard error: 1576 on 75 degrees of freedom
#Multiple R-squared: 0.9841,   Adjusted R-squared: 0.9613
#F-statistic: 43.04 on 108 and 75 DF, p-value: < 2.2e-16