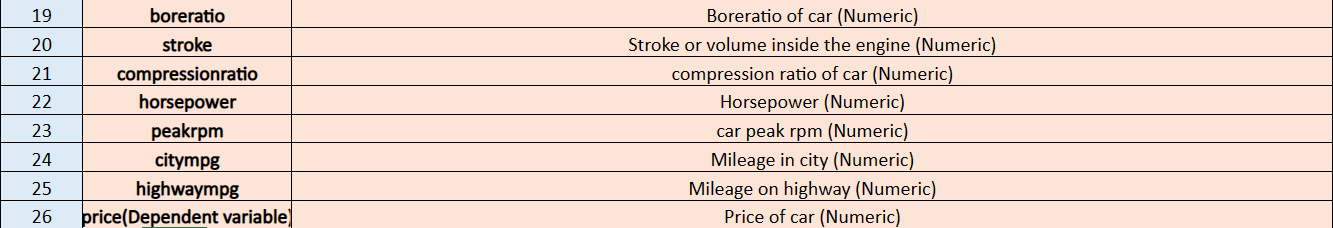
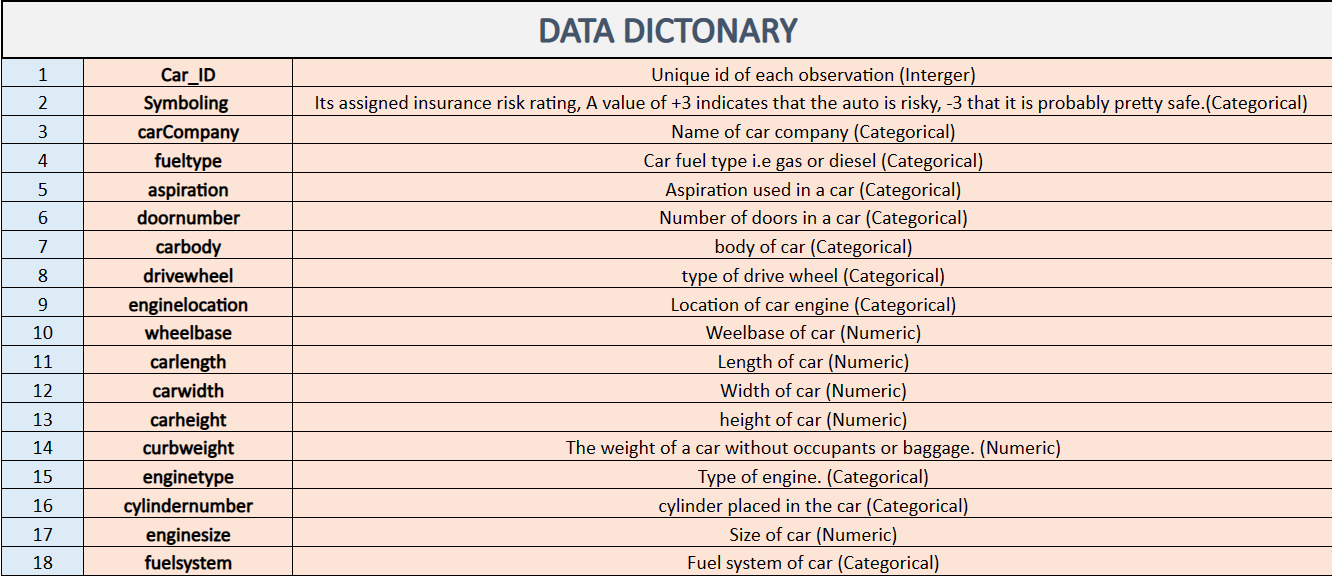
CNO- C22019881934

NAME- AKSHADA MALPURE

ROLL NO- 3934

CASE STUDY OF CAR PRICE PREDICTION USING R



#the tidyverse library is used to import various packages like the ggplot

library(tidyverse)

library(GGally)

install.packages("caTools")    # For Logistic regression

install.packages("ROCR")       # For ROC curve to evaluate model

#to read any file into our Rstudio we use read.csv and store it into a variable

predict\_carprice <- read.csv("D:/CaseStudy/carprice.csv")

#using the range function we get to know the min and max value of all the entries in that particular dataset column

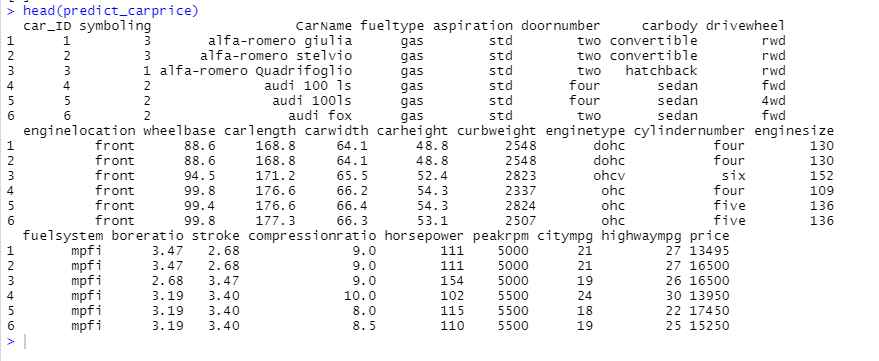
#the dim function gives number of rows and columns of the dataset

dim(predict\_carprice)

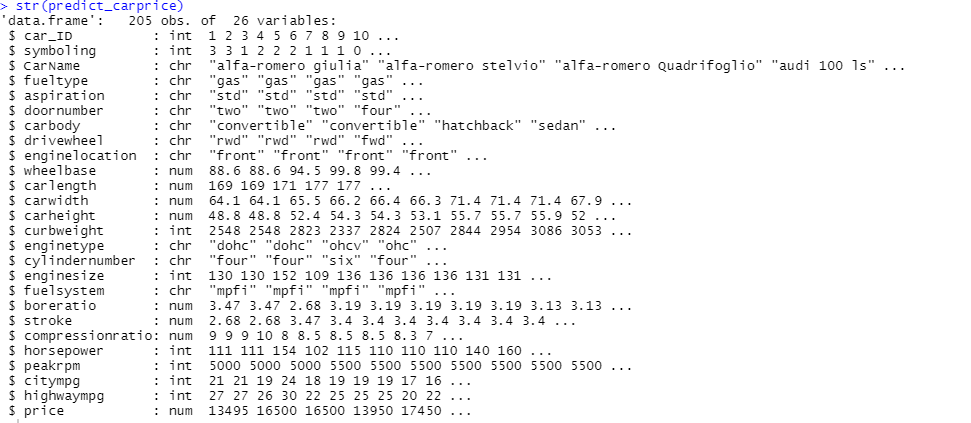


#this function gives the top 6 rows of each column

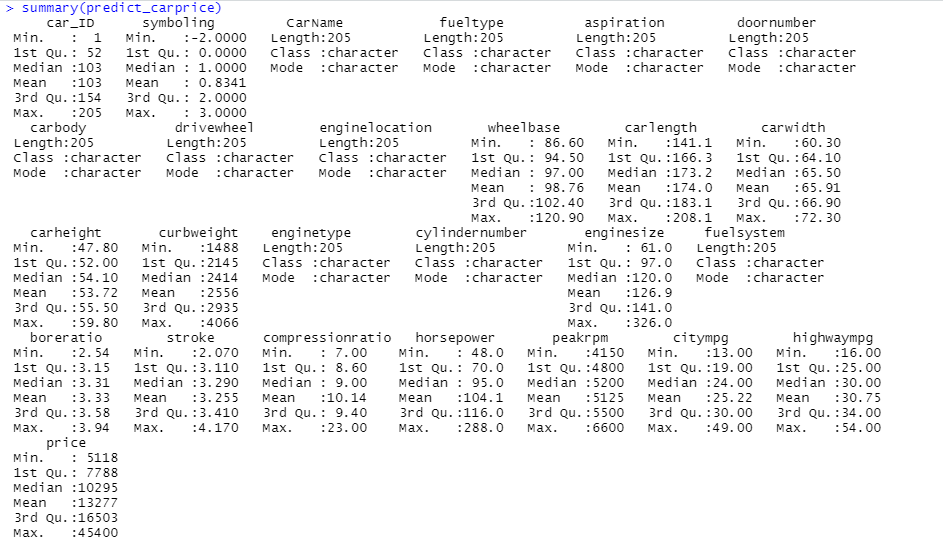
head(predict\_carprice)



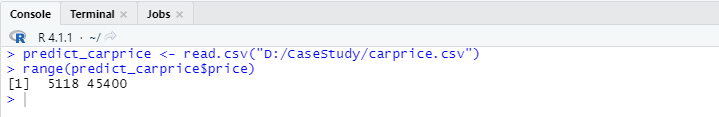
#[str()](https://stat.ethz.ch/R-manual/R-patched/library/utils/html/str.html): shows the structure of the data frame



#[summary()](https://stat.ethz.ch/R-manual/R-devel/library/base/html/summary.html): provides summary statistics on the columns of the data frame



range(predict\_carprice$price)



#symboling column- Its assigned insurance risk rating,

#A value of +3 indicates that the auto is risky, -3 that it is probably pretty safe.

#x axis-independent variable

#y axis -dependent variable

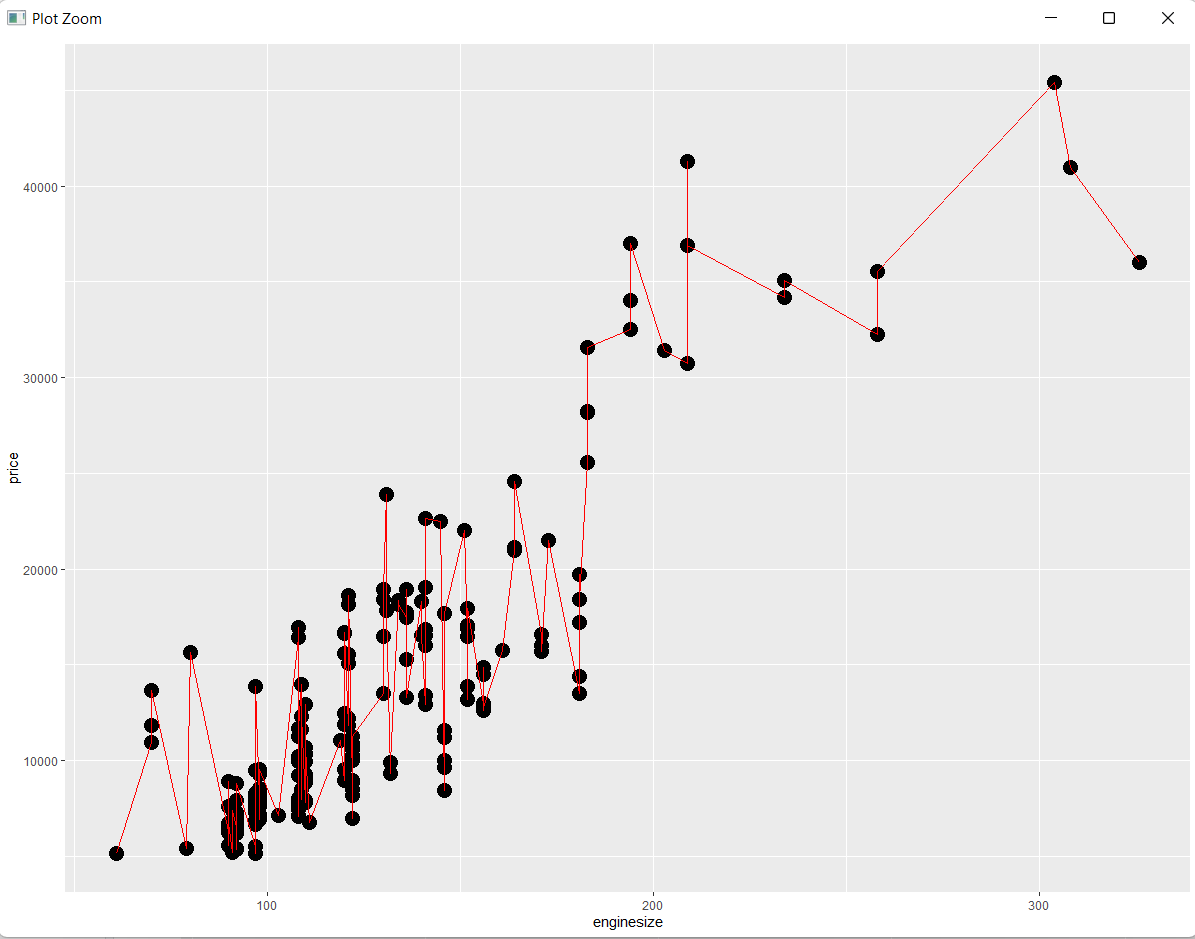
#aes - aesthetics

#geom point- using this all points are dotted

#geom size- helps to set the size of the points/dots

#geom line- it joins all the dots

ggplot(data=predict\_carprice,mapping = aes(x=enginesize,y=price))+geom\_point(size=5)+geom\_line(color="red")



#color=fueltype - this helps to mark fuels of the same type in same colour

#alpha -is used to makes the dots transparent

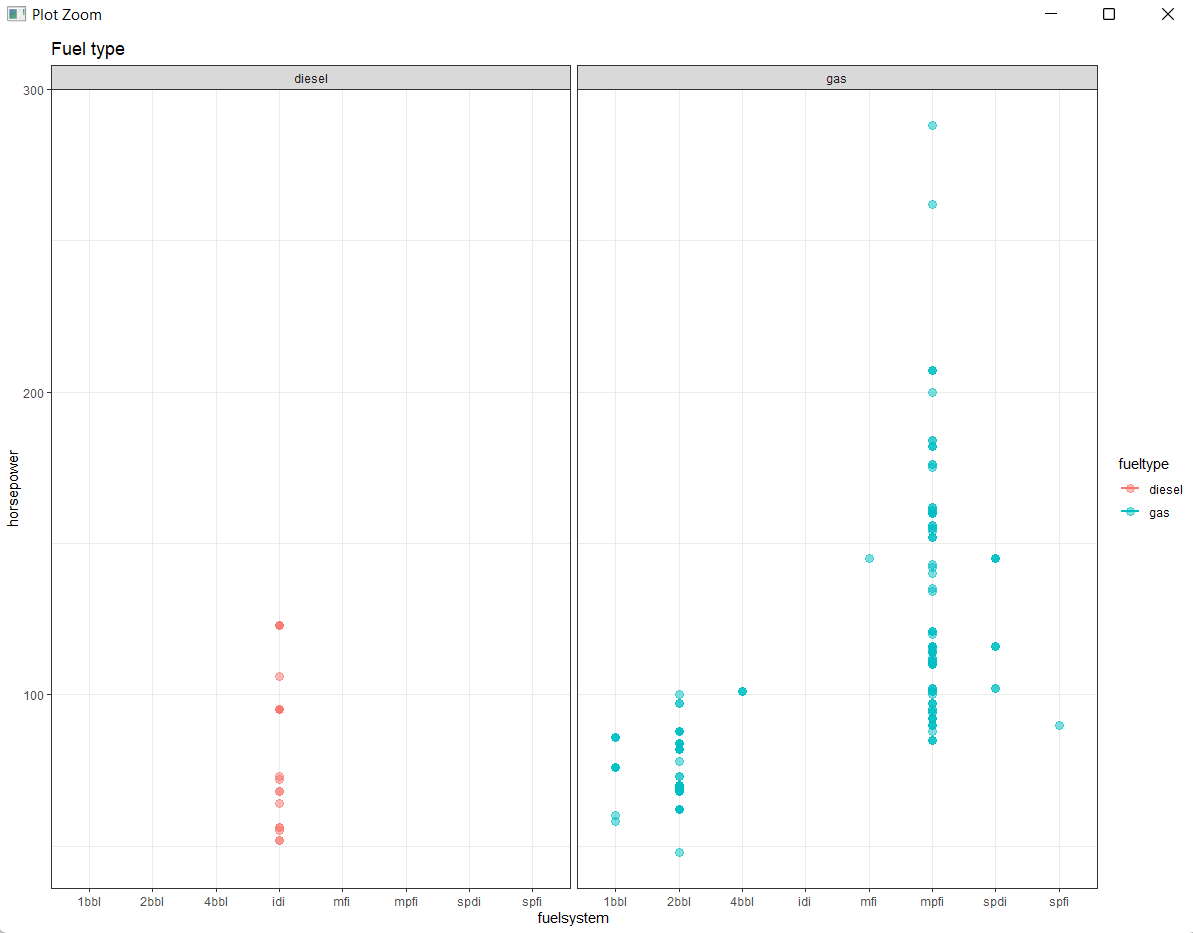
#geom\_smooth- makes the plot smoother

#facet\_wrap(~fueltype)- draws separate graphs based on the fueltype

#labs- add title to our graph

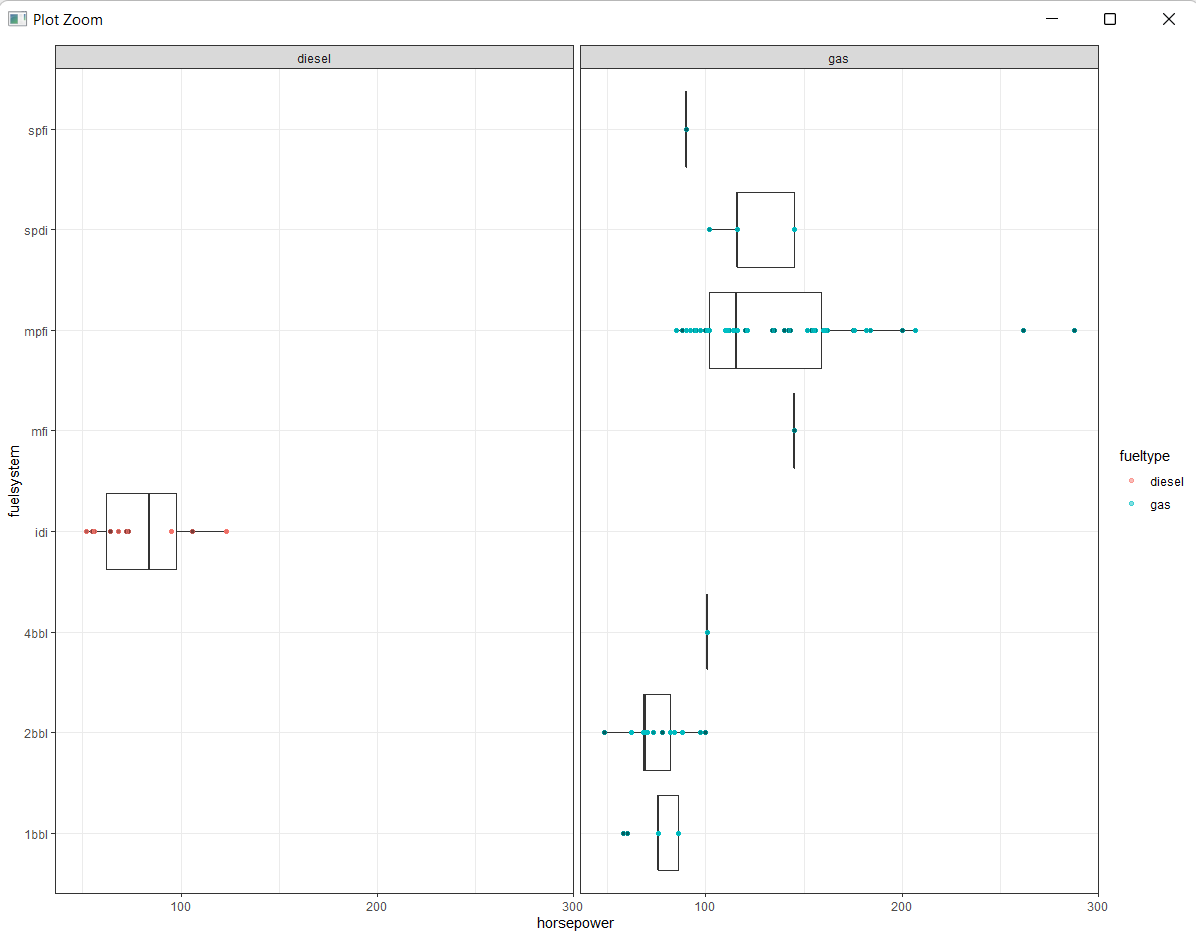
#method=lm means liner model

ggplot(predict\_carprice,aes(fuelsystem,horsepower,color=fueltype))+geom\_point(size=3,alpha=0.5)+geom\_smooth(method = lm,se=F)+facet\_wrap(~fueltype)+labs(title = "Fuel type") +theme\_bw()#add black and white theme

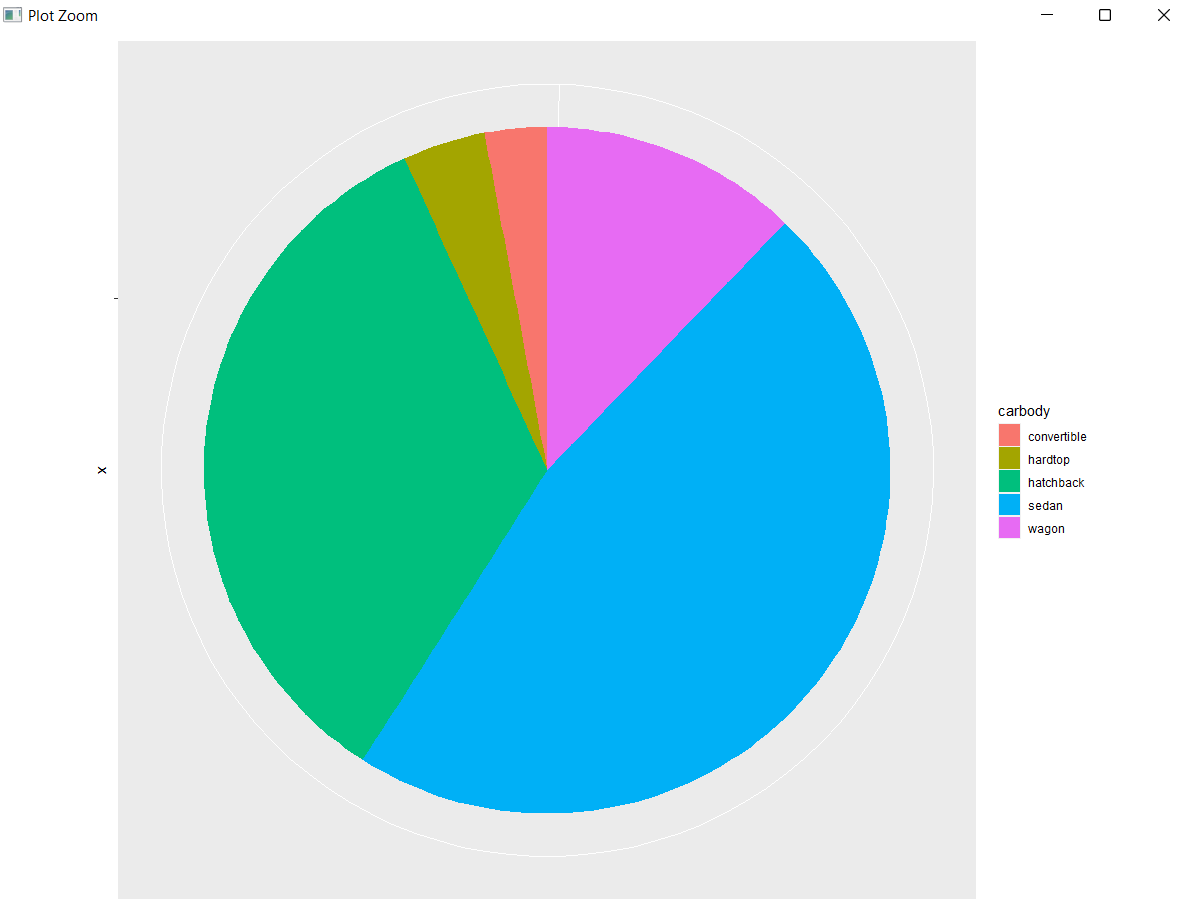


#coord\_flip() - used to flip the boxplot

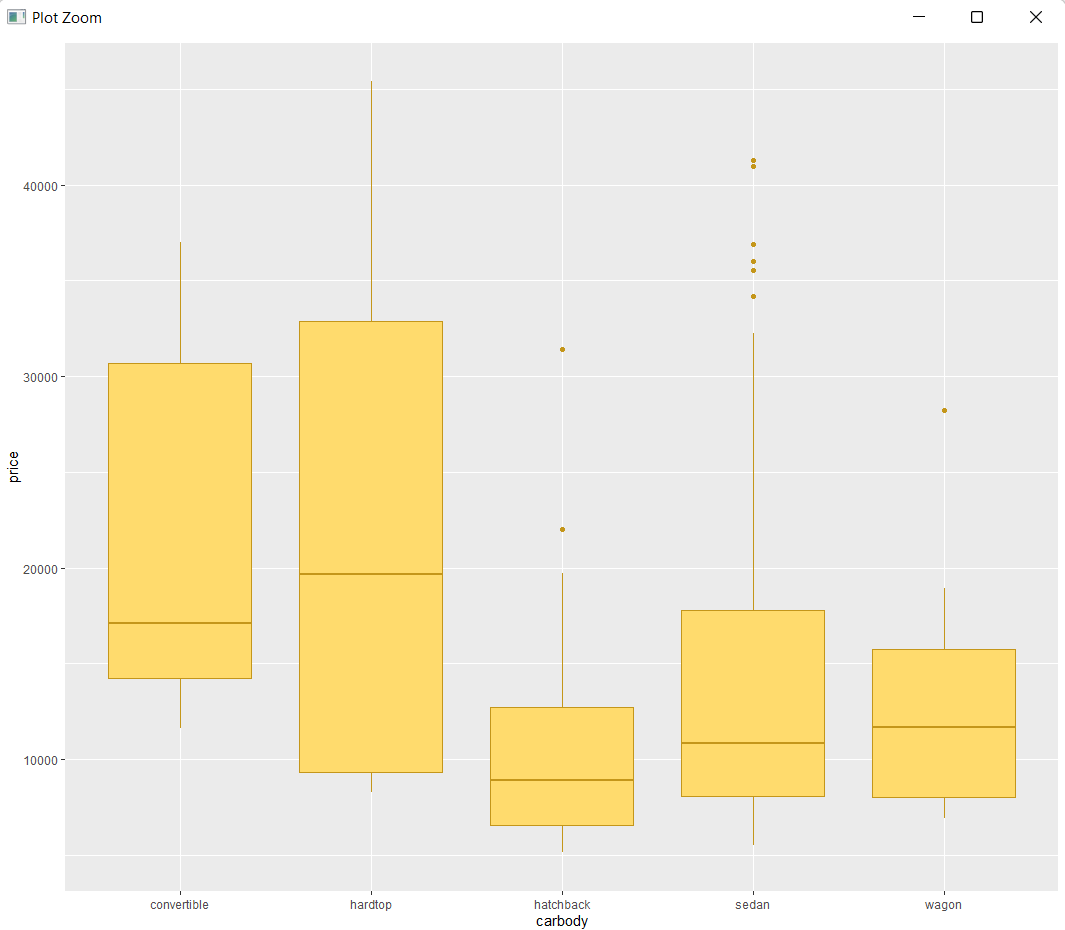
ggplot(predict\_carprice,aes(fuelsystem,horsepower))+geom\_boxplot()+geom\_point()+geom\_point(alpha=0.5,aes(color=fueltype))+coord\_flip()+theme\_bw()+facet\_wrap(~fueltype)



ggplot(predict\_carprice, aes(x="", y="", fill=carbody)) +geom\_bar(stat="identity", width=2) +coord\_polar("y", start=0)



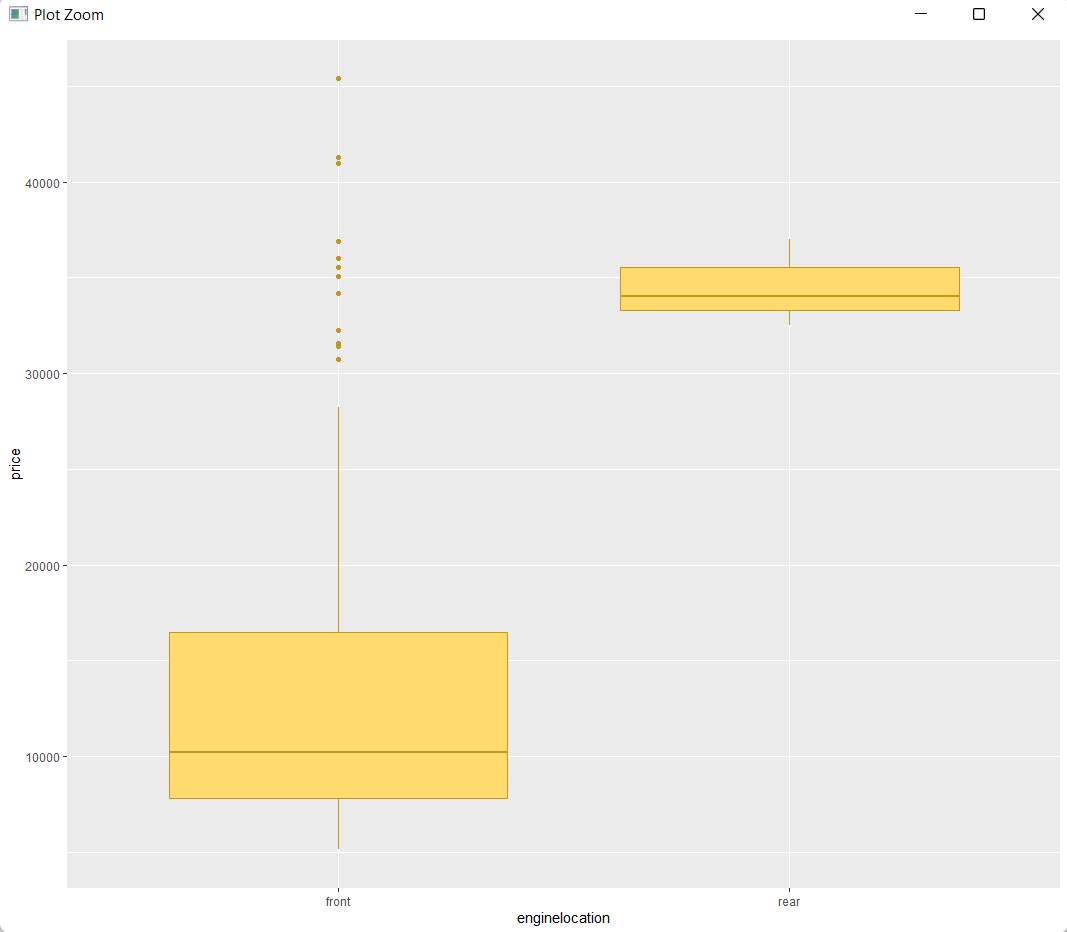
ggplot(predict\_carprice,aes(carbody,price))+geom\_boxplot(fill="#FFDB6D",color="#C4961A")



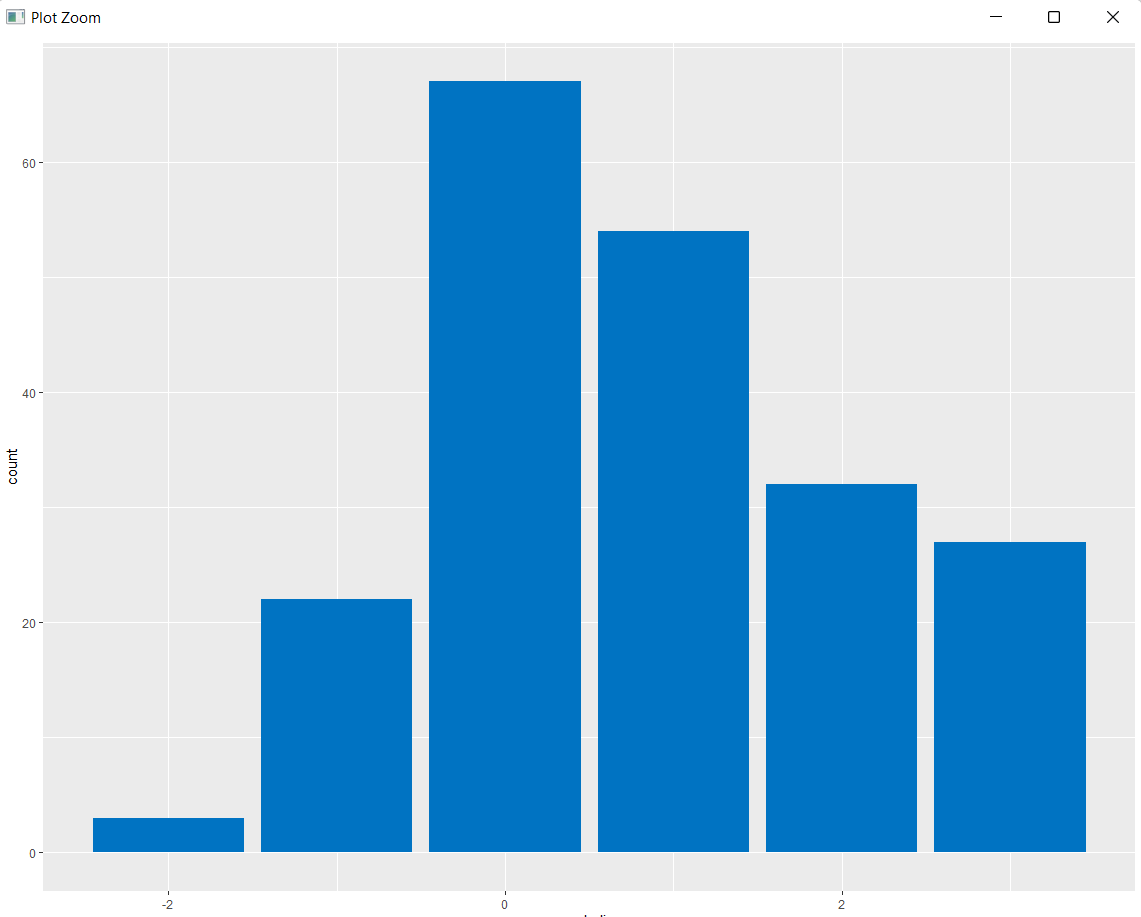
#the price of hardtop type of cars is the most dispersed

#the price of convertible and sedan is most skewed indicating that their prices are more than that of median

ggplot(predict\_carprice,aes(enginelocation,price))+geom\_boxplot(fill="#FFDB6D",color="#C4961A")

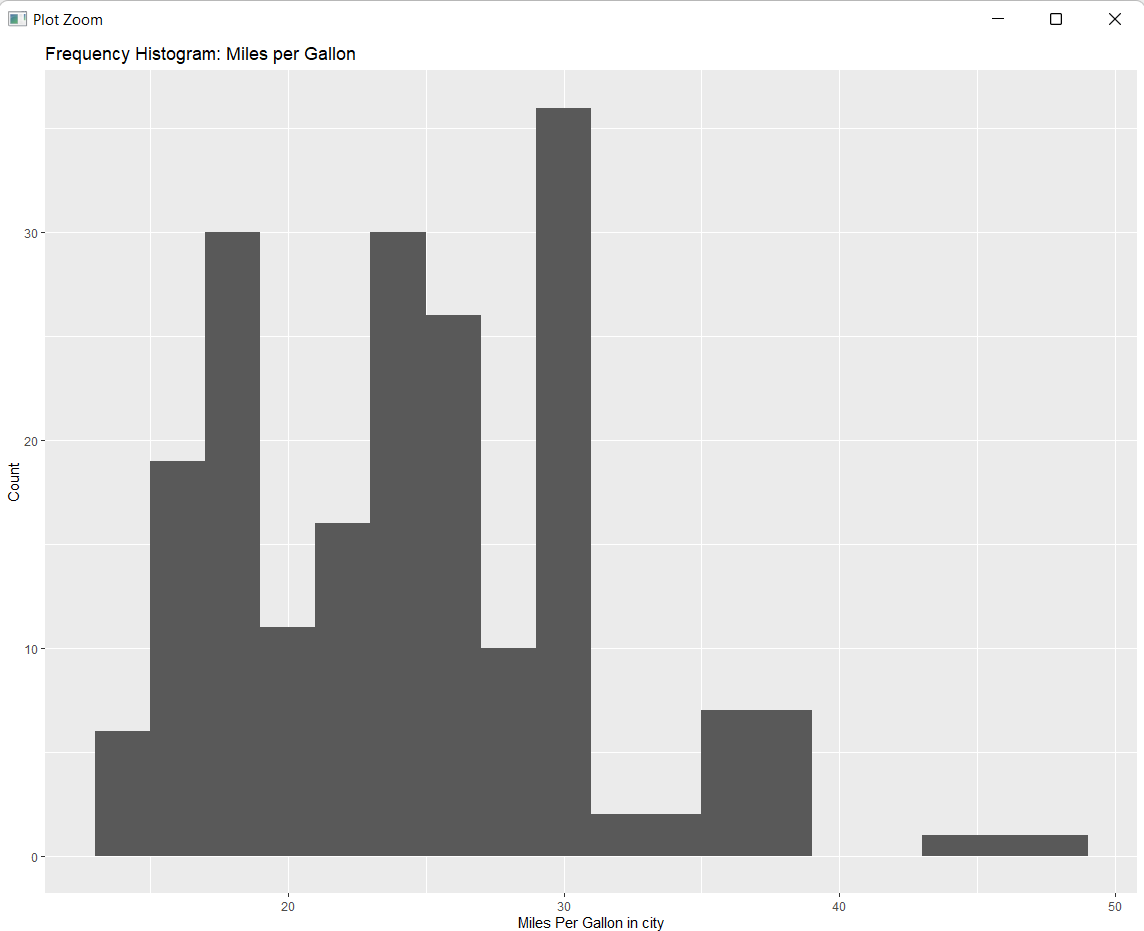


ggplot(predict\_carprice,aes(symboling))+geom\_bar(fill="#0073C2FF")



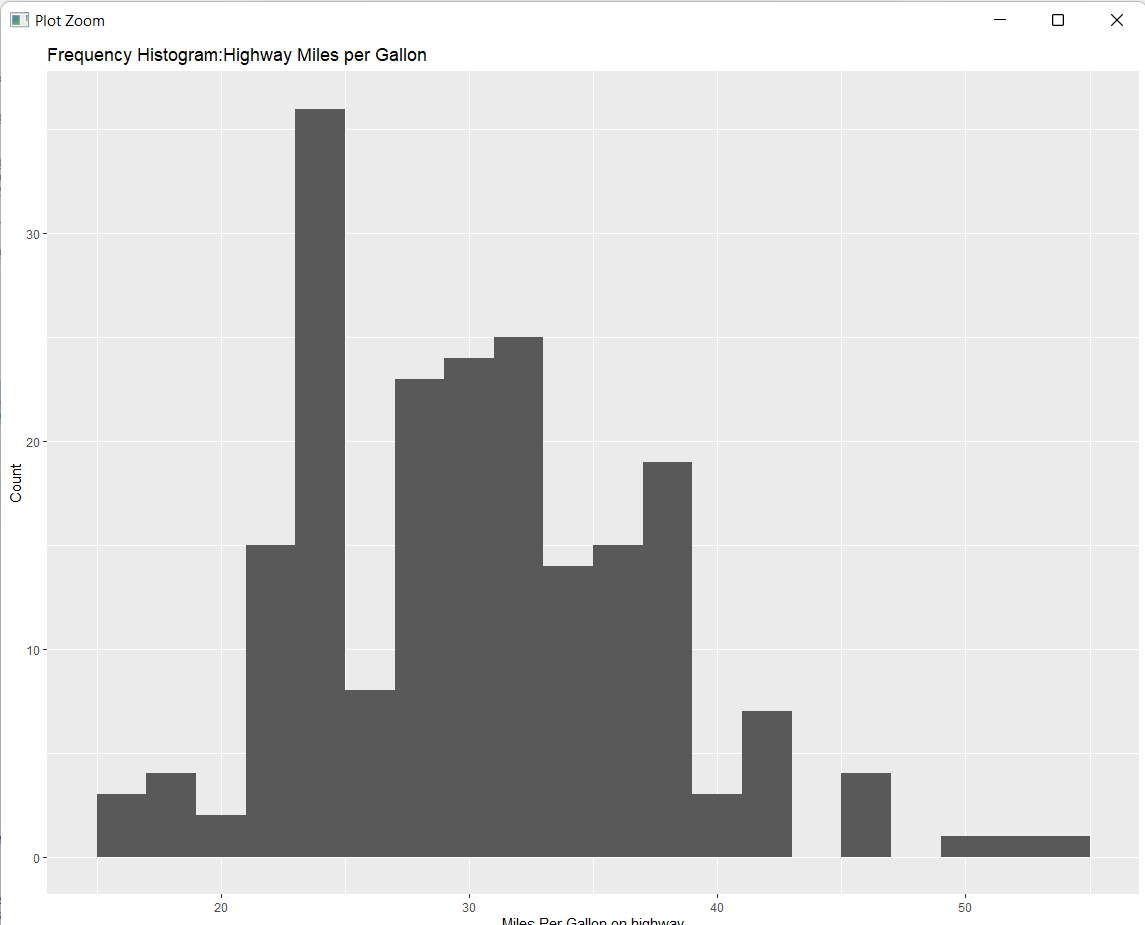
#City Miles Per Gallon

qplot(predict\_carprice$citympg, xlab = 'Miles Per Gallon in city', ylab = 'Count', binwidth = 2, main='Frequency Histogram: Miles per Gallon')



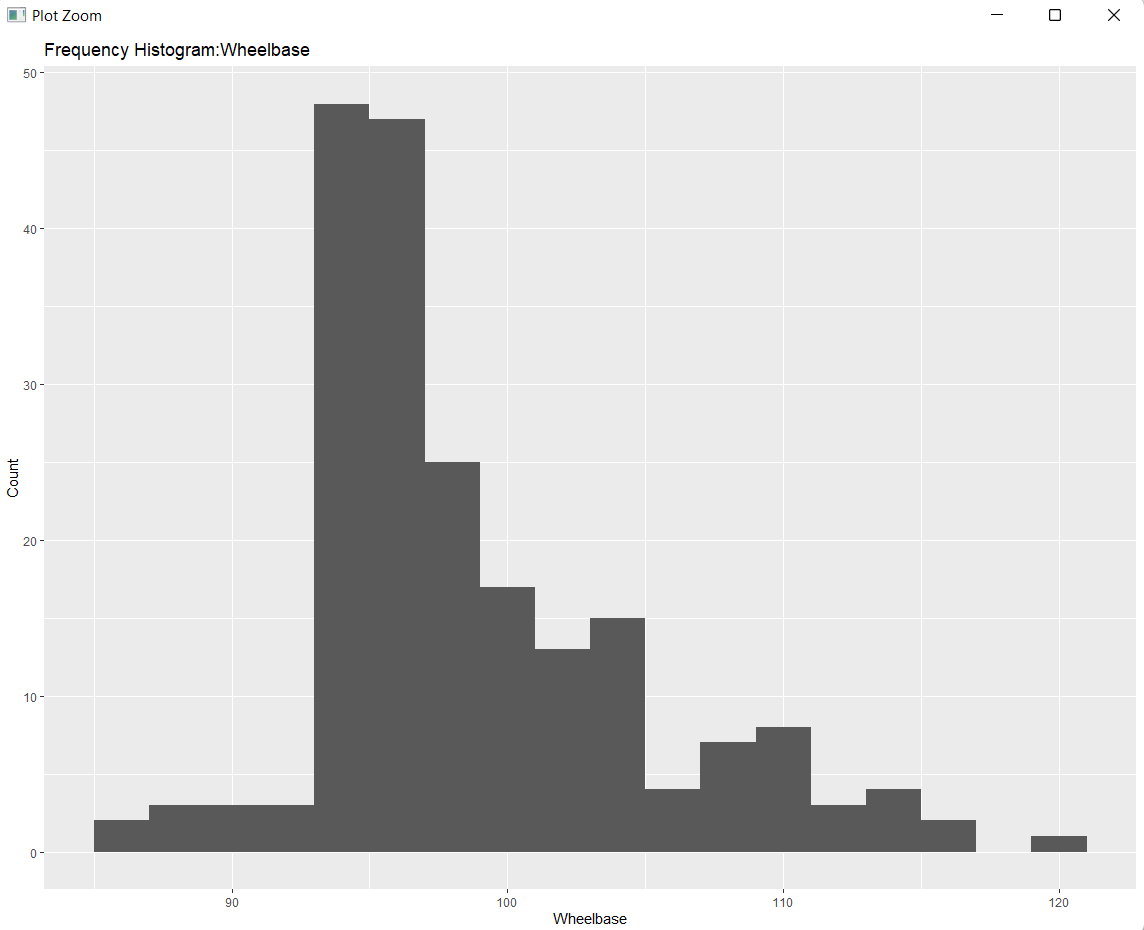
#Highway Miles Per Gallon

qplot(predict\_carprice$highwaympg, xlab = 'Miles Per Gallon on highway', ylab = 'Count', binwidth = 2,main='Frequency Histogram:Highway Miles per Gallon')



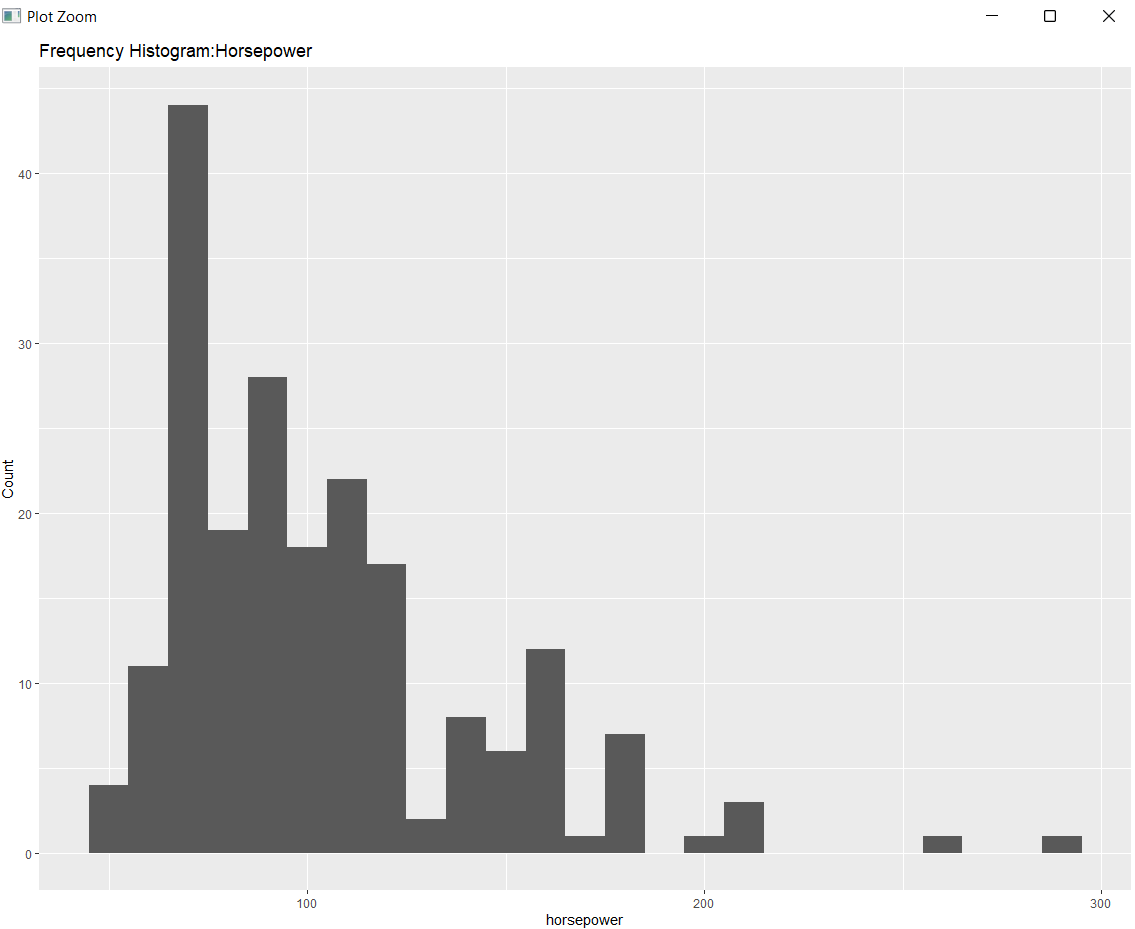
# Wheelbase

qplot(predict\_carprice$wheelbase, xlab = 'Wheelbase', ylab = 'Count', binwidth = 2,main='Frequency Histogram:Wheelbase')



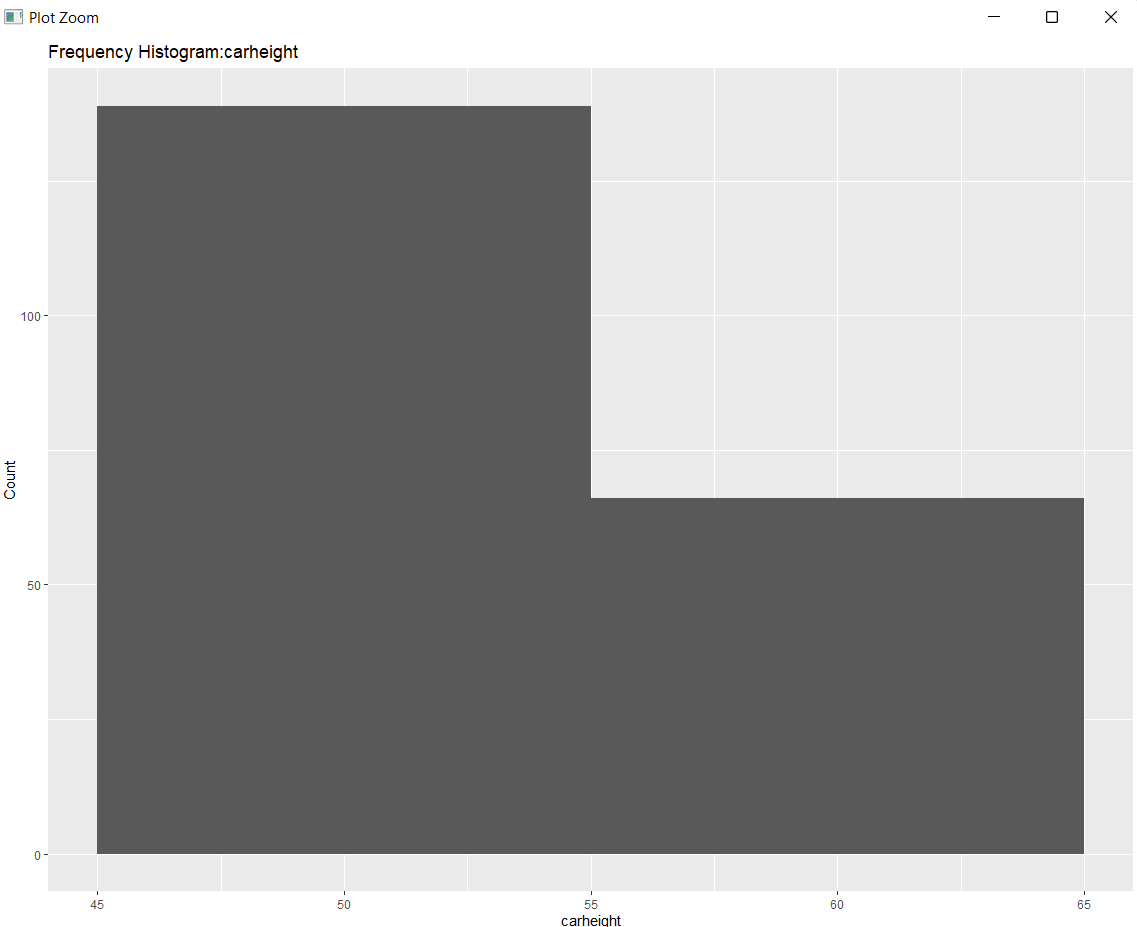
#Horsepower

qplot(predict\_carprice$horsepower, xlab = 'horsepower', ylab = 'Count', binwidth = 10,main='Frequency Histogram:Horsepower')



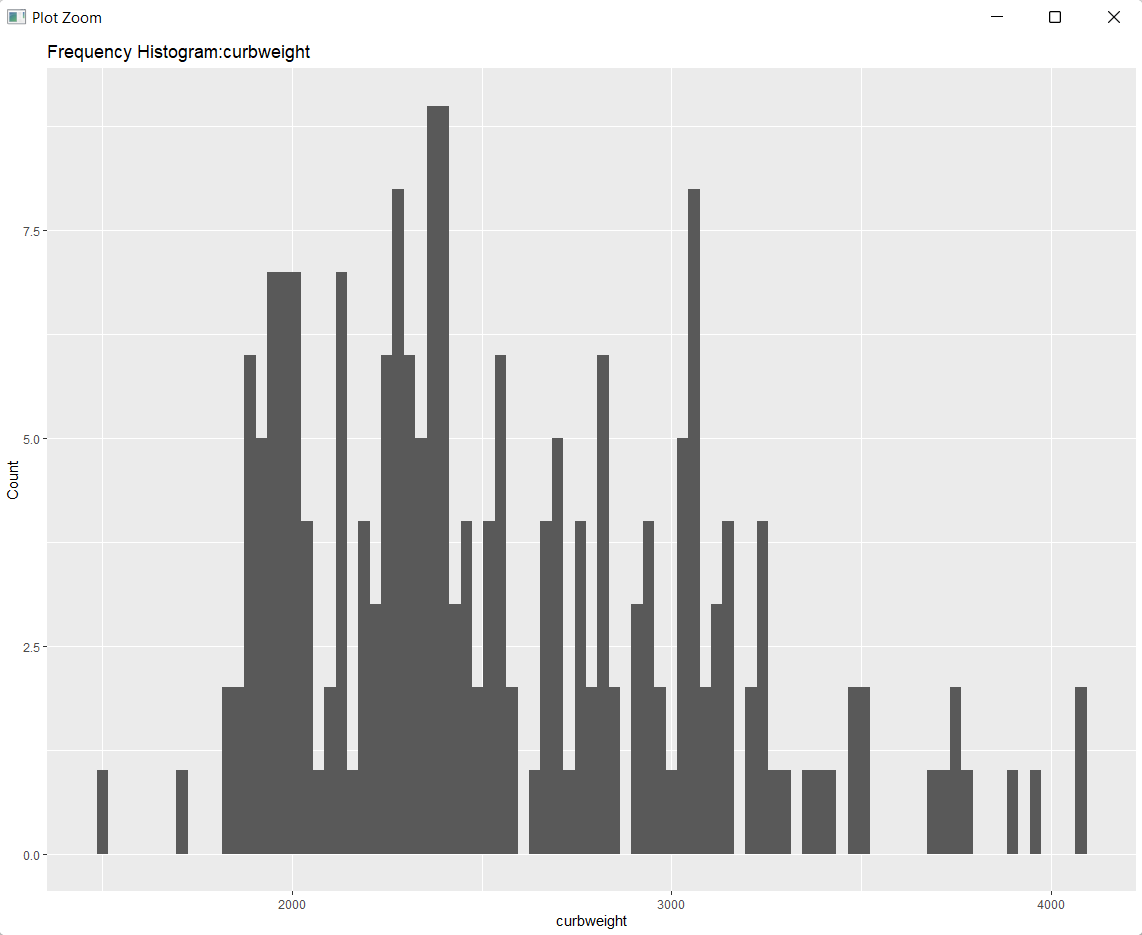
#carheight

qplot(predict\_carprice$carheight, xlab = 'carheight', ylab = 'Count', binwidth = 10,main='Frequency Histogram:carheight')



#curbweight

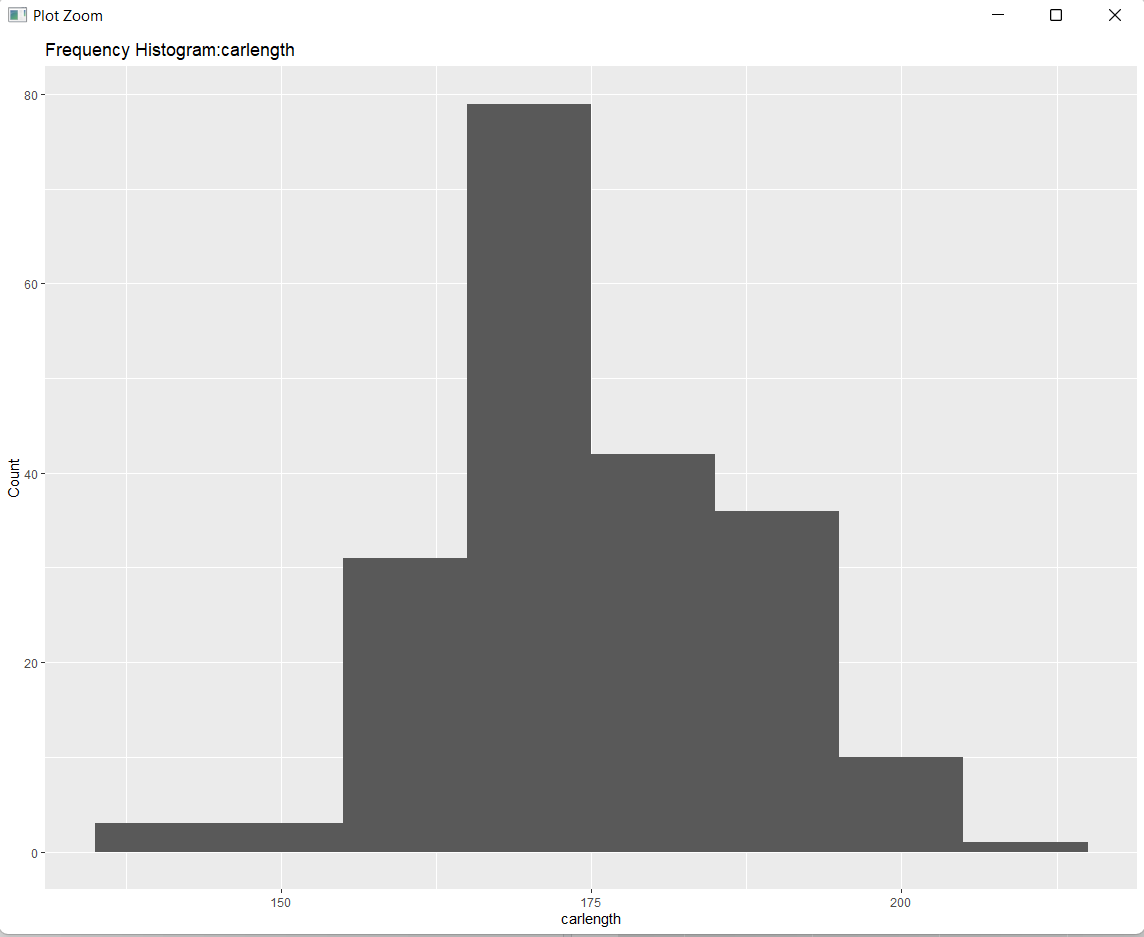
qplot(predict\_carprice$curbweight, xlab = 'curbweight', ylab = 'Count', binwidth = 30,main='Frequency Histogram:curbweight')



The distributions for citympg, highwaymsg, horsepower, carheight, carweight and curbweight are all skewed right–a longer tail toward the higher end of the scale. This supports my intuition that there is a strong correlation between all of those variables.

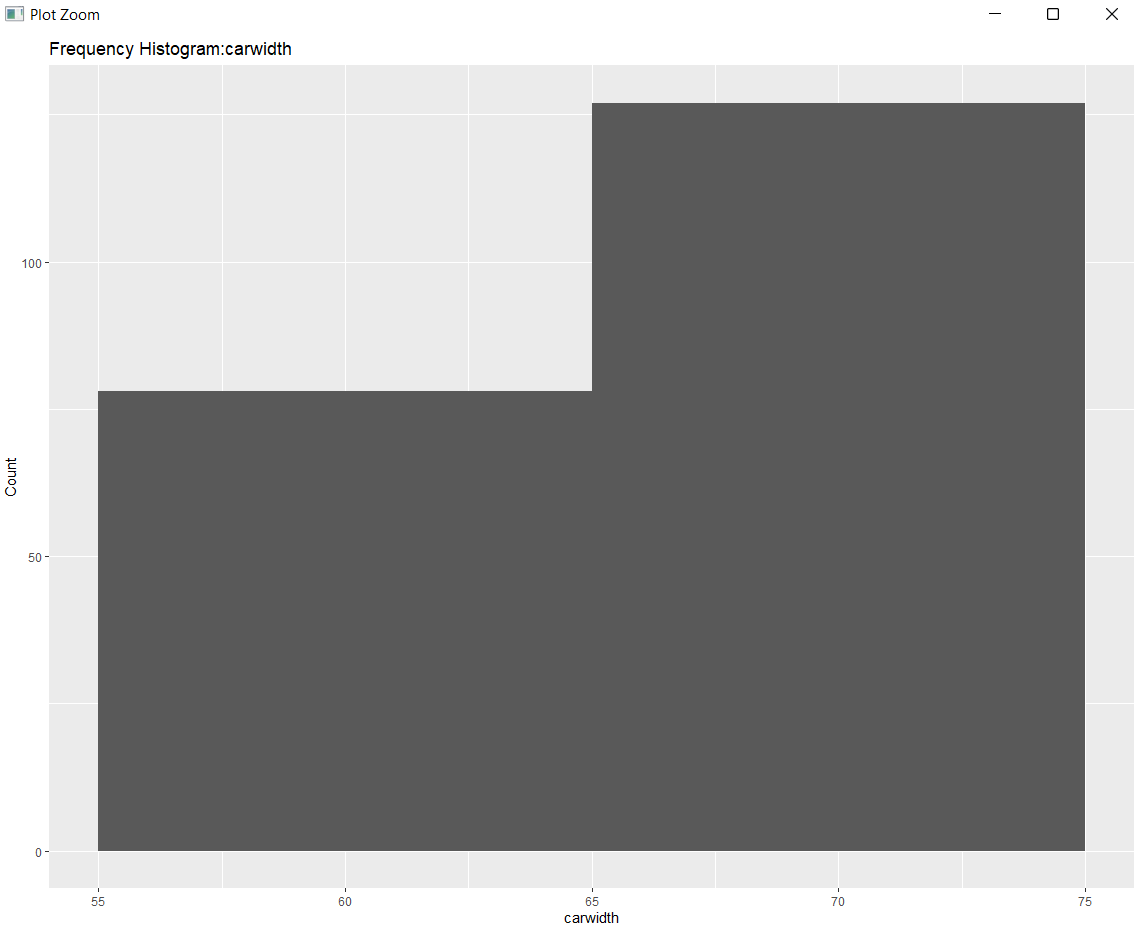
#carlength

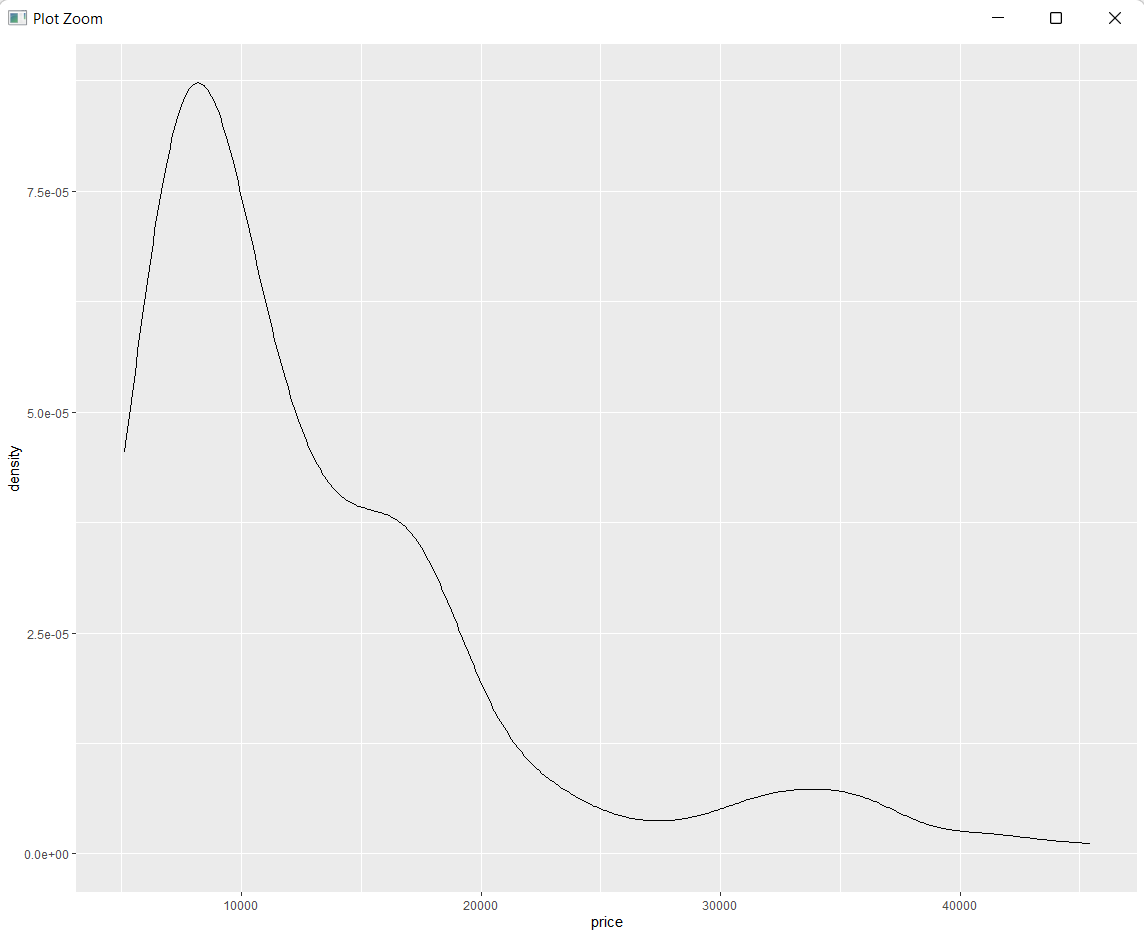
qplot(predict\_carprice$carlength, xlab = 'carlength', ylab = 'Count', binwidth = 10,main='Frequency Histogram:carlength')



#carwidth

qplot(predict\_carprice$carwidth, xlab = 'carwidth', ylab = 'Count', binwidth = 10,main='Frequency Histogram:carwidth')





# Inference

# Mean and median of price are significantly different.

# Large standard deviation indicates that there is considerable variance in the prices of the automobiles.

# Price values are right-skewed, most cars are priced at the lower end (9000) of the price range.

Price Analysis

ggplot(predict\_carprice, aes(x = predict\_carprice$curbweight, y = citympg)) +

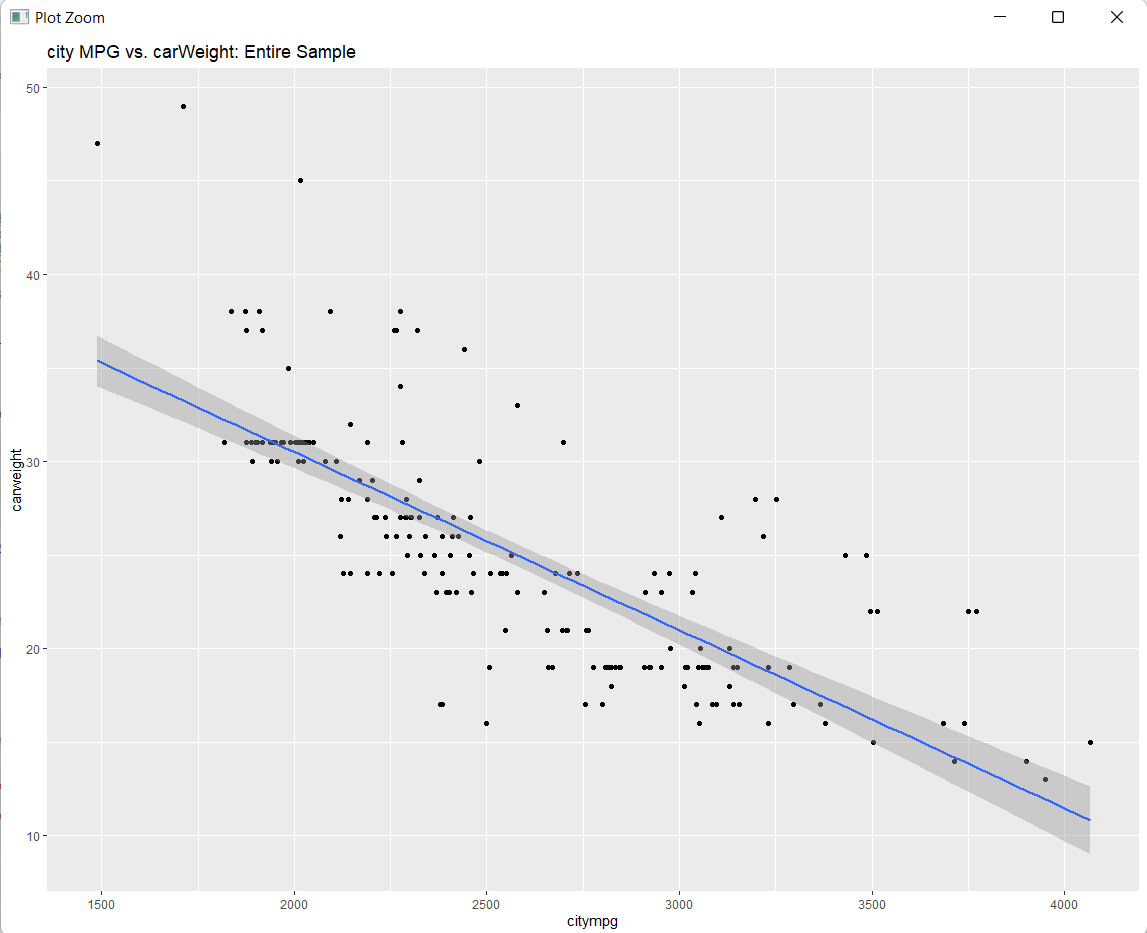
geom\_point() +

geom\_smooth(method='lm') +

xlab('citympg') +

ylab('carweight') +

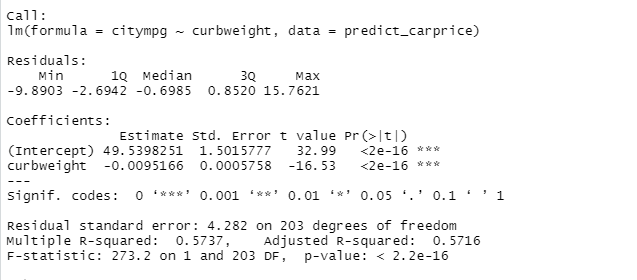
ggtitle('city MPG vs. carWeight: Entire Sample')



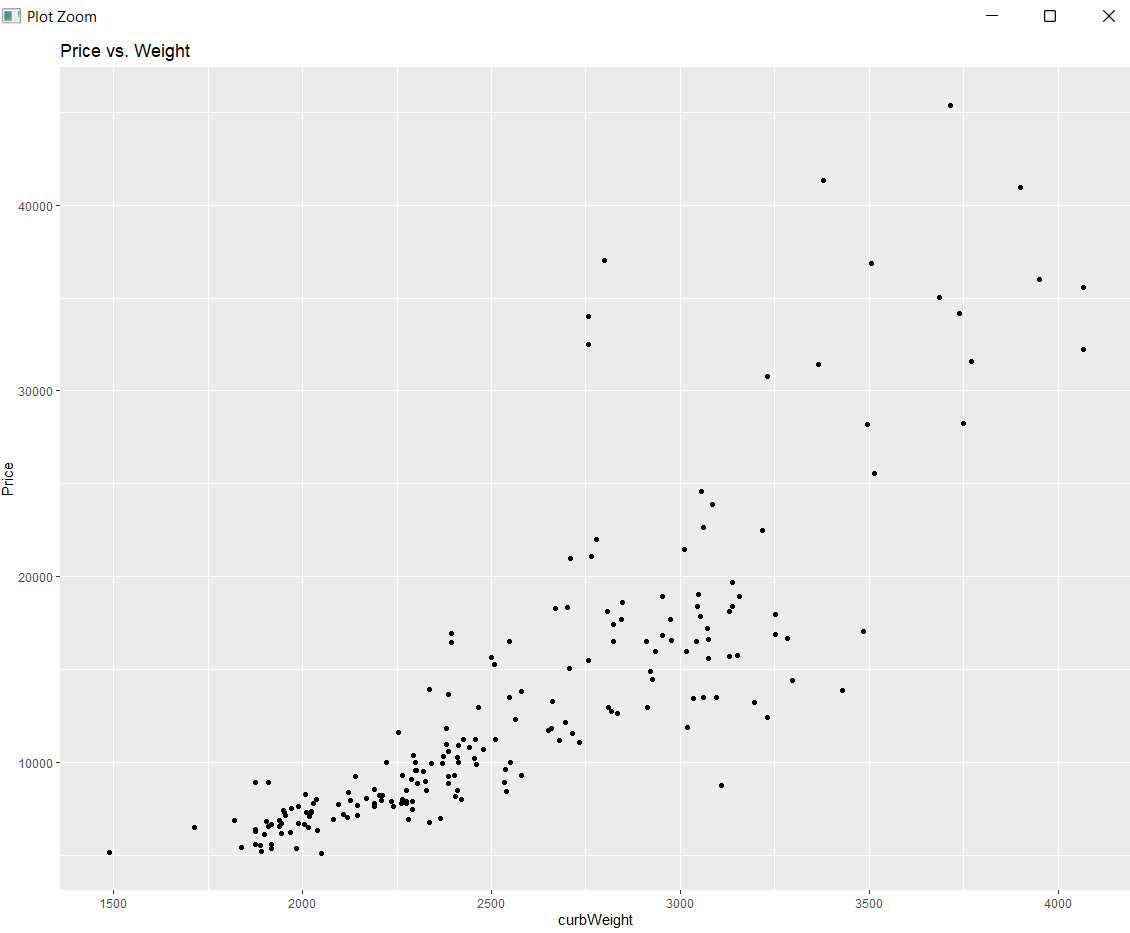
The data clearly shows that weight and MPG are inversely related: as weight increases, MPG decreases. The R-squared of the linear best fit line, as shown below, is over 57%. This means that variations in a car’s weight explain over 57% of the changes to its MPG.

fit = lm(citympg ~ curbweight, predict\_carprice)

summary(fit)



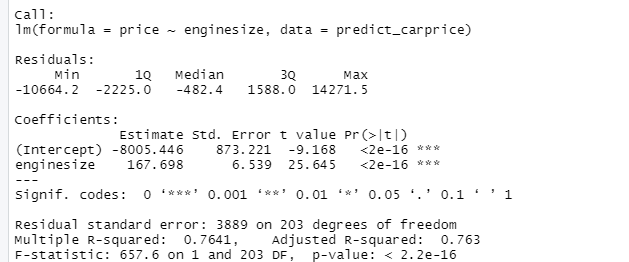
ggplot(predict\_carprice, aes(x = curbweight, y = price)) +geom\_point() +xlab('curbWeight') + ylab('Price') +ggtitle('Price vs. Weight')



There is positive correlation between weight and price.

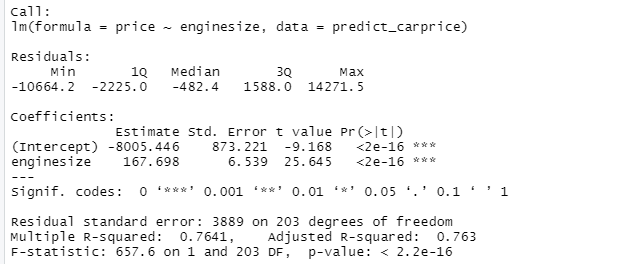
fit = lm(price ~enginesize, predict\_carprice)

summary(fit)



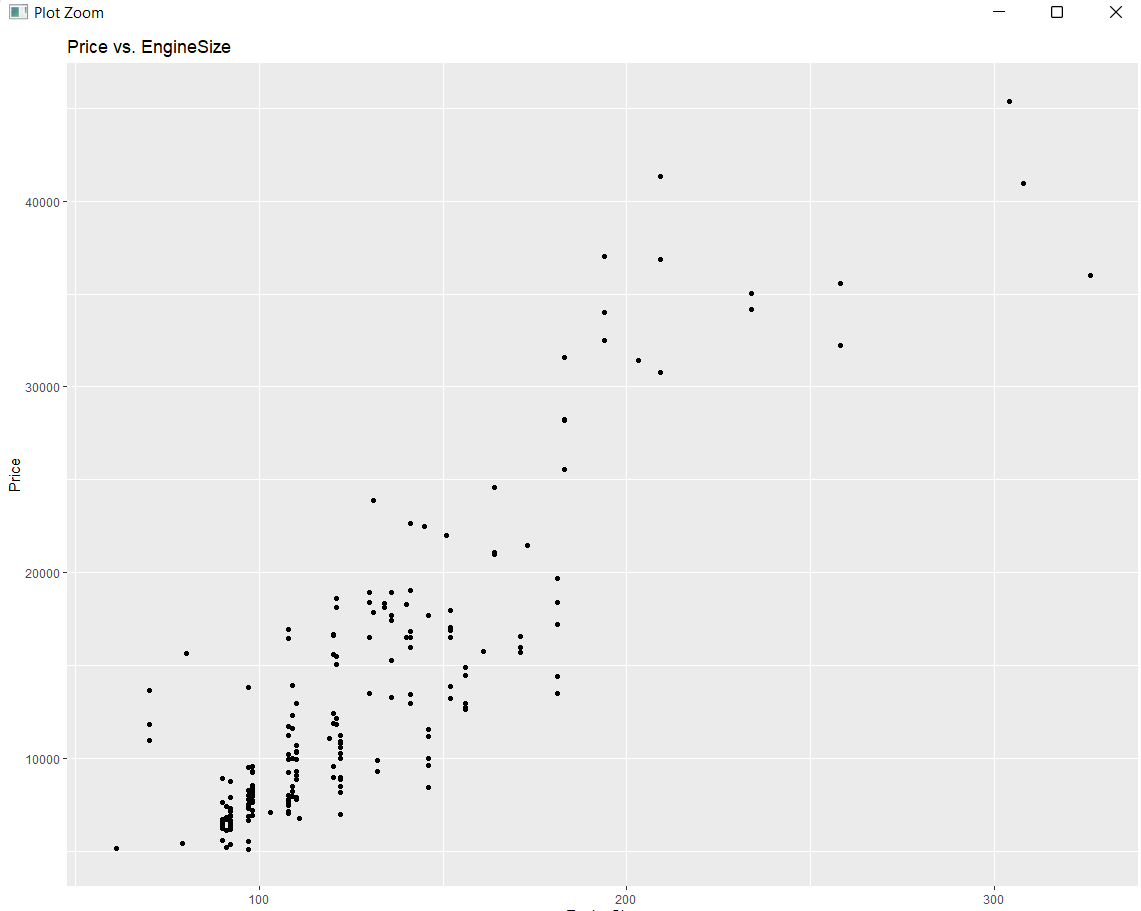
fit = lm(price ~enginesize, predict\_carprice)

summary(fit)



ggplot(predict\_carprice, aes(x =enginesize, y = price)) + geom\_point() +

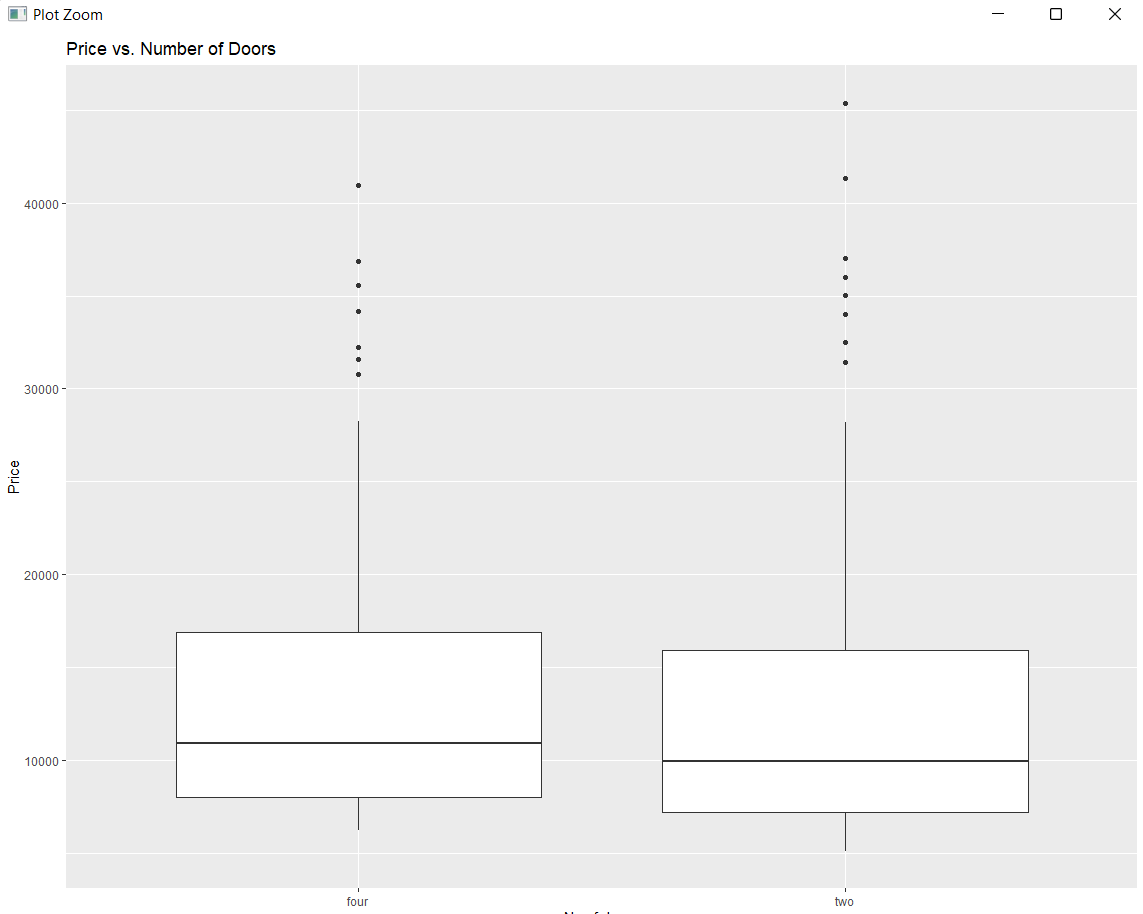
xlab('EngineSize') + ylab('Price') +ggtitle('Price vs. EngineSize')



There is positive correlation between enginesize and price.

ggplot(predict\_carprice, aes(x = doornumber, y = price)) +geom\_boxplot() +xlab('No of doors') +ylab('Price') +

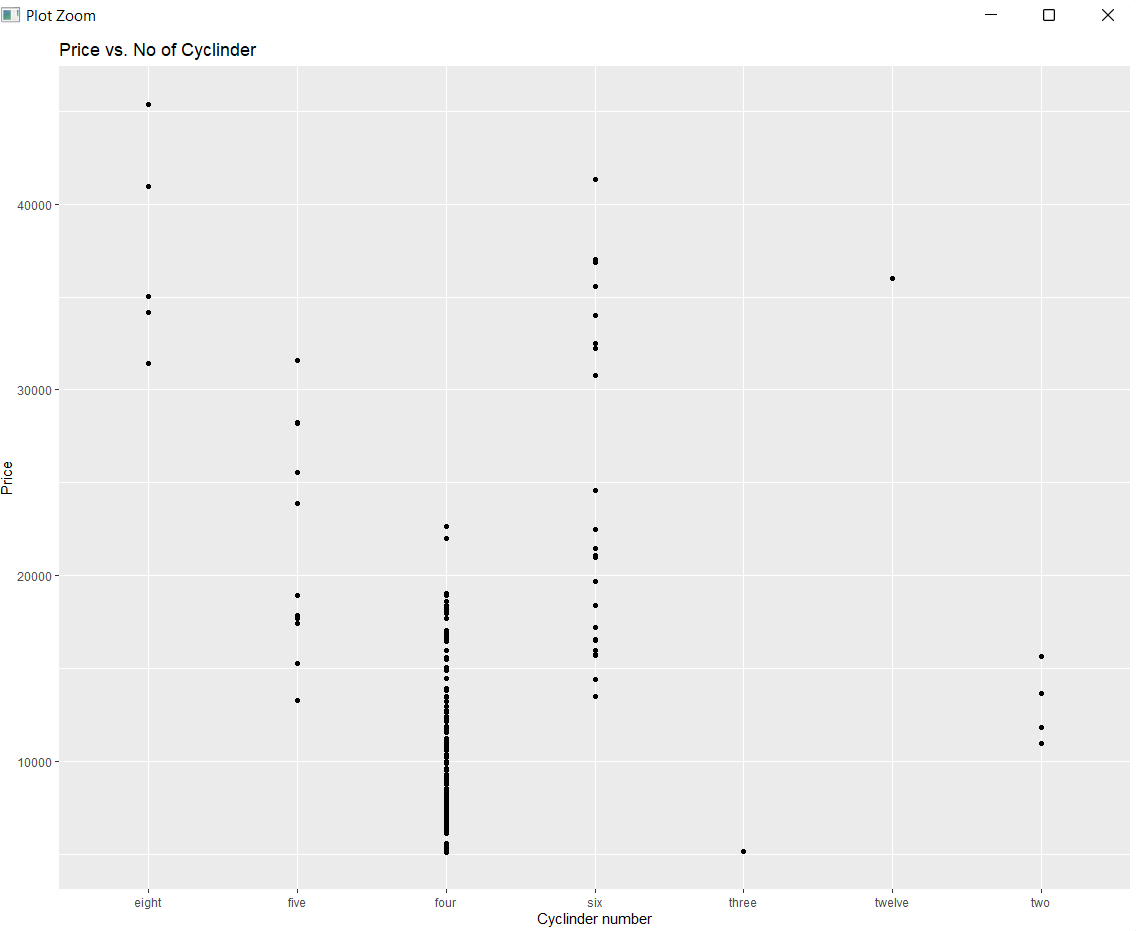
ggtitle('Price vs. Number of Doors')



#number of doors does not significantly contribute towards price

ggplot(predict\_carprice, aes(x =cylindernumber, y = price)) +geom\_point() +

xlab('Cyclinder number') + ylab('Price') + ggtitle('Price vs. No of Cyclinder')



**From the above Univariate and bivariate analysis we can filter out variables which does not affect price much.**  
**The most important driver variable for prediction of price are:-**

1. Brand Category
2. Fuel Type
3. Aspiration
4. Car Body
5. Drive Wheel
6. Wheelbase
7. Car Length
8. Car Width
9. Curb weight
10. Engine Type
11. Cylinder Number
12. Engine Size
13. Bore Ratio
14. Horsepower
15. Mileage

**3-D VISUALIZATION**

**HEATMAP**

data<- data.frame(predict\_carprice$price,predict\_carprice$symboling,predict\_carprice$wheelbase,predict\_carprice$carlength,predict\_carprice$carwidth,predict\_carprice$carheight,predict\_carprice$curbweight,predict\_carprice$enginesize,predict\_carprice$peakrpm,predict\_carprice$citympg,predict\_carprice$highwaympg)

x<- as.matrix(data)

rc <- rainbow(nrow(x),start=0,end=0.3)

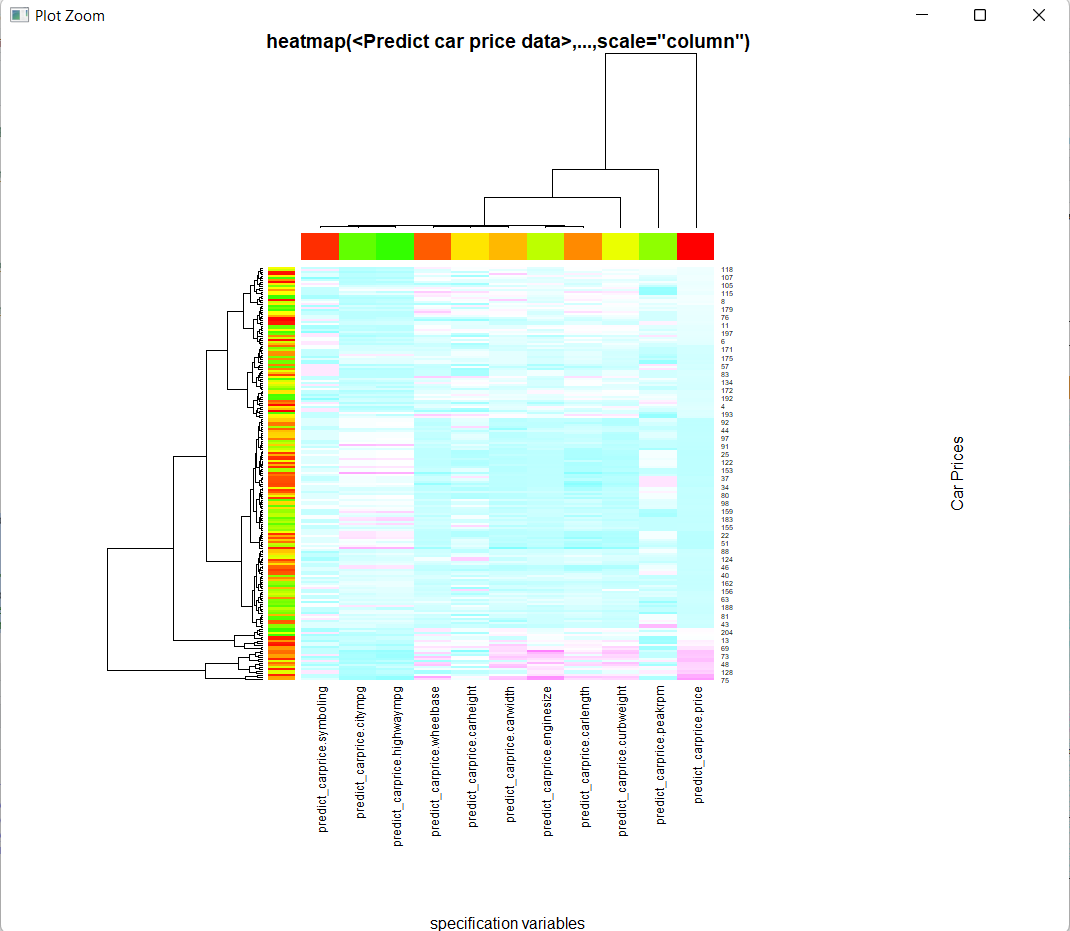
cc <- rainbow(ncol(x),start = 0,end = 0.3)

heatmap(x,col= cm.colors(254),scale= "column",

RowSideColors = rc, ColSideColors = cc,margins = c(20,20),

xlab = "specification variables", ylab = "Car Prices",

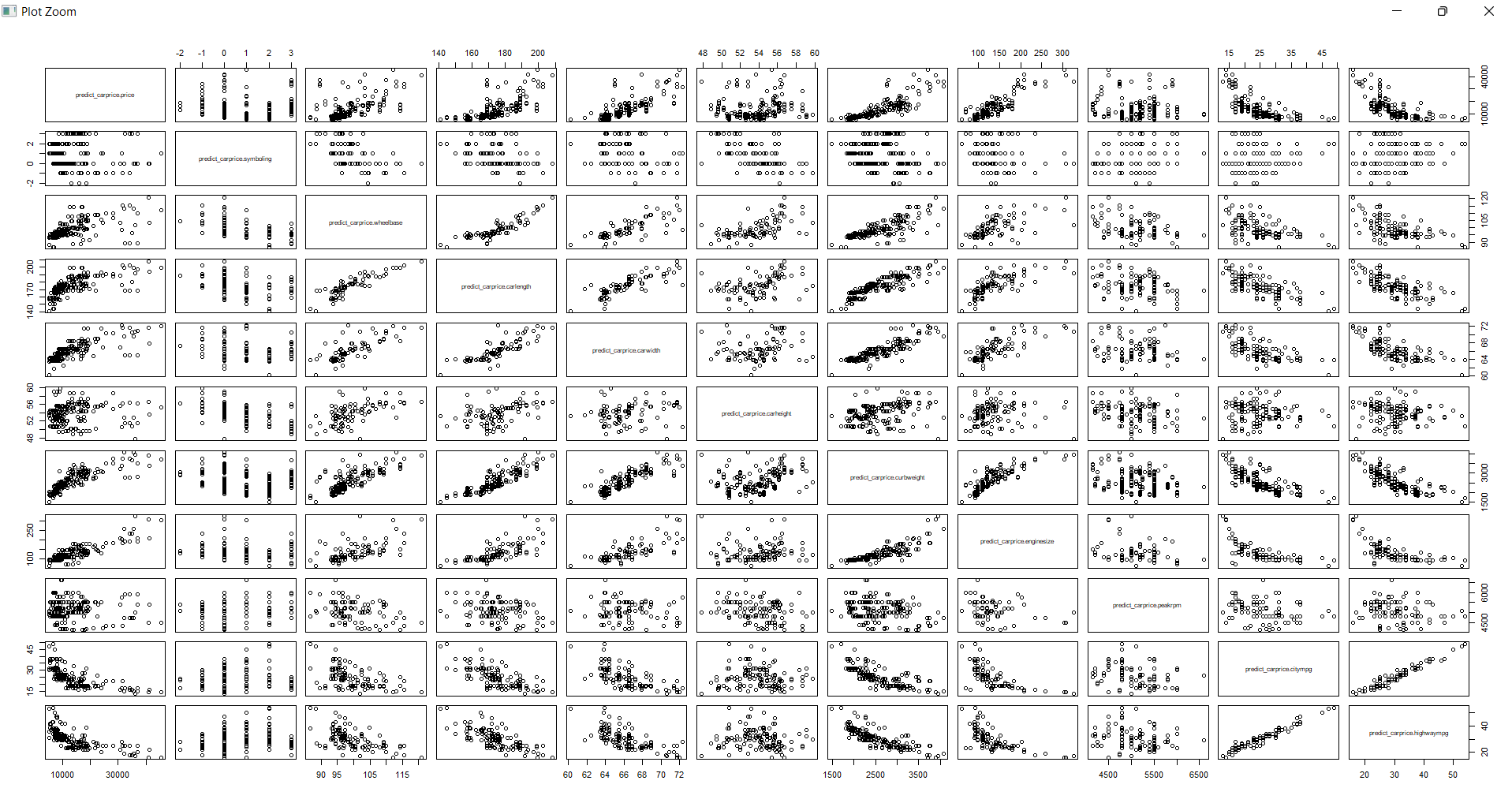
main = "heatmap(<Predict car price data>,...,scale=\"column\")")



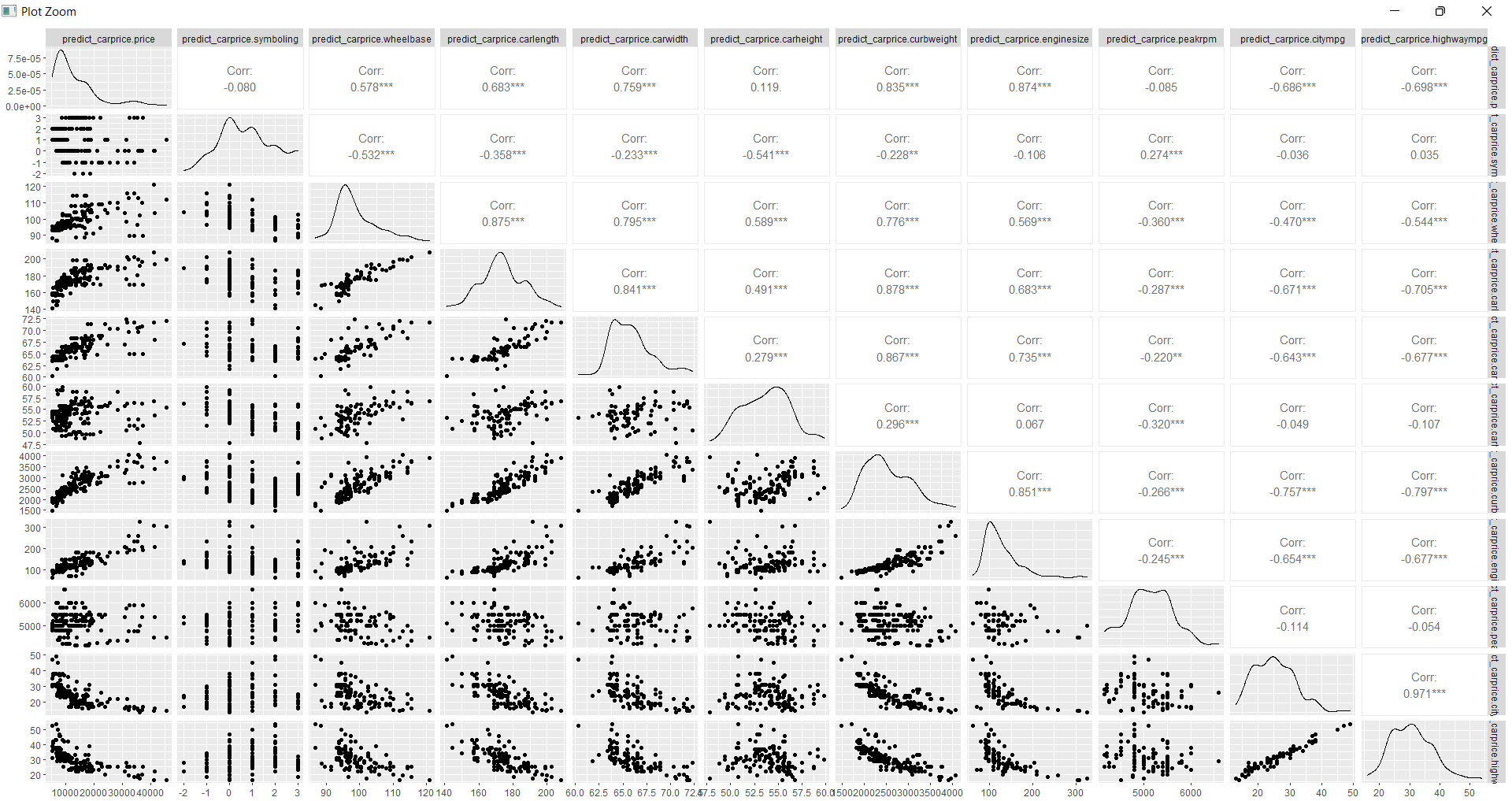
#pairplots give us relation of every variable with every other variable

data<- data.frame(predict\_carprice$price,predict\_carprice$wheelbase,predict\_carprice$carlength,predict\_carprice$carwidth,predict\_carprice$carheight,predict\_carprice$curbweight,predict\_carprice$enginesize,predict\_carprice$peakrpm,predict\_carprice$citympg,predict\_carprice$highwaympg)

pairs(data)



ggpairs(data)



# From the above plot we can infer that-

#1. symbolling doesn’t significantly matter for car price prediction

#2. Wheelbase, car length, car width, curb weight, engine size has a positive relationship with car price

#3. City mpg, highway mpg has a negative relationship with car price

#4. This suggest that cars having high mileage may fall in the 'economy' cars category, and are priced lower.

#Corelation plot between the most significant attributes

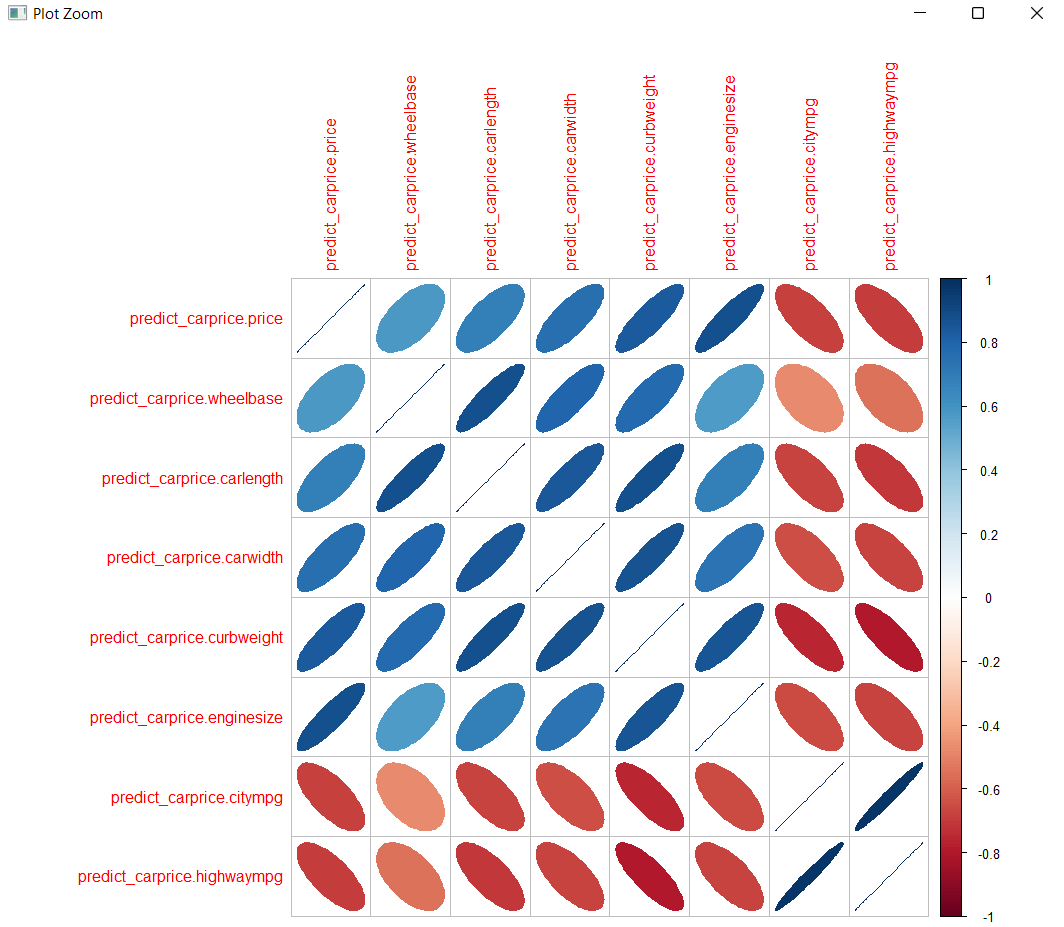
library(corrplot)

most\_significant <- data.frame(predict\_carprice$price,predict\_carprice$wheelbase,predict\_carprice$carlength,predict\_carprice$carwidth,predict\_carprice$curbweight,predict\_carprice$enginesize,predict\_carprice$citympg,predict\_carprice$highwaympg)

pairs(most\_significant)

cor\_plot <-cor(most\_significant,use = "complete.obs")

corrplot(cor\_plot,method="ellipse")



#highest positive corelation is found between: curbweight and price

#also there is strong corelation between enginesize and price

# there is negative corelation between price and citympg, price and highwaympg

#from the above observations we can infer that

1.price increases with curbweight and enginesize

2.price increases with decrease in citympg and highwaympg

REGRESSION MODEL

# remove unused variables

car\_data <- predict\_carprice %>% select(-c(car\_ID, CarName))

# remove cylinder number with only 1 instance

cylinder <- predict\_carprice %>% count(cylindernumber) %>% filter(n > 3)

car\_data <- car\_data[predict\_carprice$cylindernumber %in% cylinder$cylindernumber, ]

car\_data$cylindernumber <- factor(car\_data$cylindernumber, unique(car\_data$cylindernumber))

# remove fueltype with only 1 instance

fuel <- predict\_carprice %>% count(fuelsystem) %>% filter(n > 3)

car\_data <- car\_data[car\_data$fuelsystem %in% fuel$fuelsystem, ]

car\_data$fuelsystem <- factor(car\_data$fuelsystem, unique(car\_data$fuelsystem))

# remove engine type with 1 instance

engine <- car\_data %>% count(enginetype) %>% filter(n > 3)

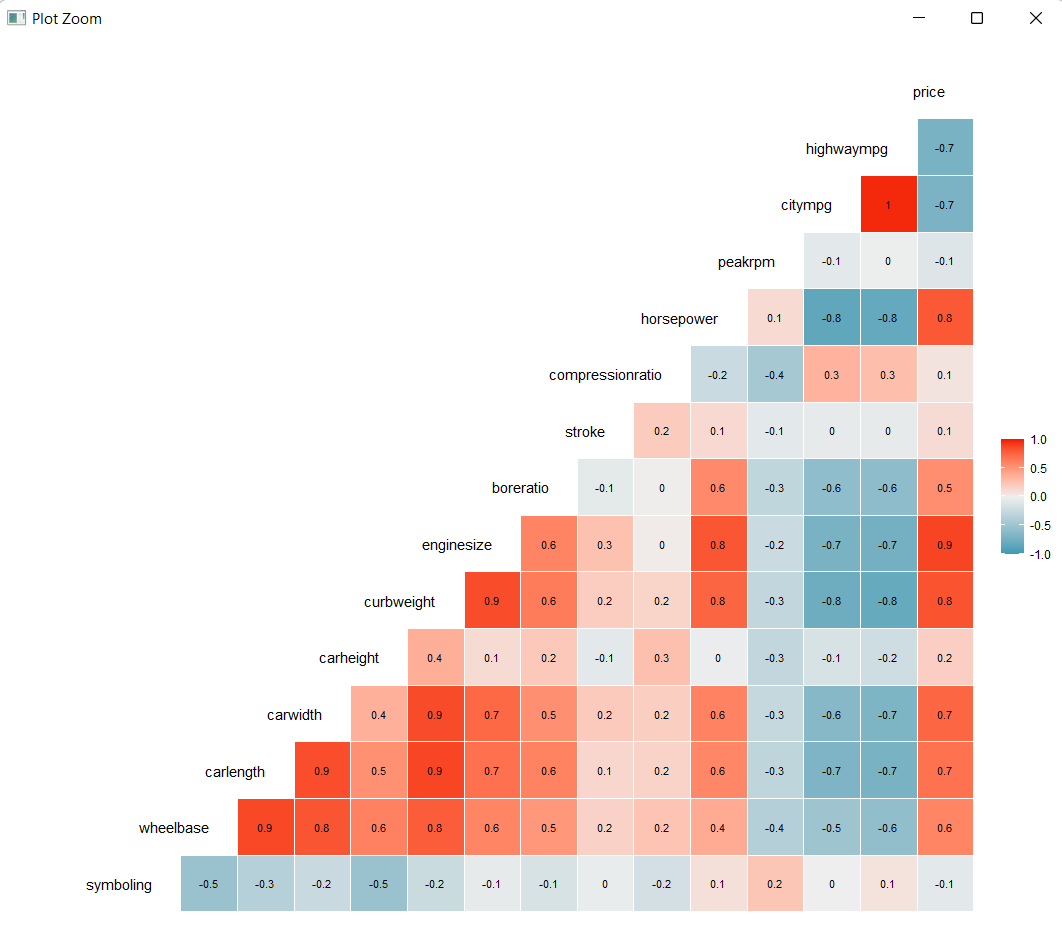
car\_data <- car\_data[car\_data$enginetype %in% engine$enginetype, ]

car\_data$enginetype <- factor(car\_data$enginetype, unique(car\_data$enginetype))

# transform character into factor

car\_data <- car\_data %>% mutate\_if(~is.character(.), ~as.factor(.))

ggcorr(car\_data, label = TRUE, label\_size = 2.9, hjust = 1, layout.exp = 2)



#The graphic shows that a lot of variable has strong correlation with the price variables.

The test dataset will be used as a comparasion and see if the model get overfit and can not predict new data that hasn’t been seen during training phase. We will 70% of the data as the training data and the rest of it as the testing data.

set.seed(123)

samplesize <- round(0.7 \* nrow(car\_data), 0)

index <- sample(seq\_len(nrow(car\_data)), size = samplesize)

data\_train <- car\_data[index, ]

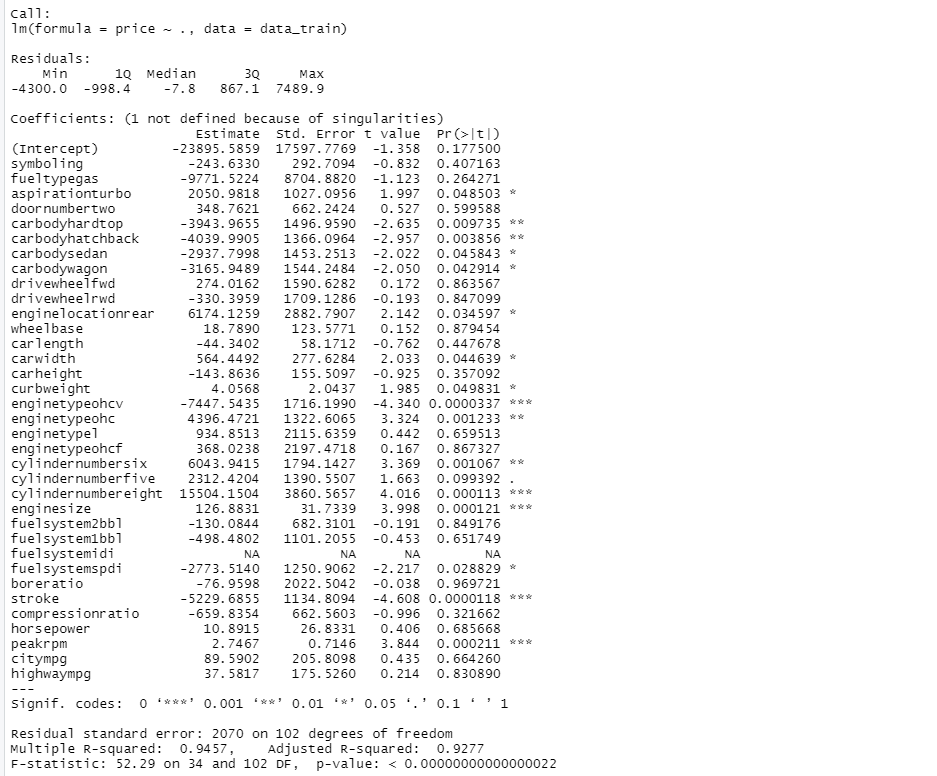
data\_test <- car\_data[-index, ]

Now we will try to model the linear regression using price as the target variable

set.seed(123)

car\_lm <- lm(price ~ ., data = data\_train)

summary(car\_lm)



The summary of car\_lm model shows a lot of information. But for now, we may be better focus on the Pr(>|t|). This column shows the signifance level of the variable toward the model. If the value is below 0.05, than we can safely asume that the variable has significant effect toward the model (meaning that the estimated coefficient are no different than 0), and vice versa. Thus, we can made a simpler model by removing variables that has p-value > 0.05, since they don’t have significant effect toward our model. The estimate value shows the coefficient of each variable. To interpret the value of each coefficient, for example with every increased value of 1 unit-point in highwaympg will contribute to 37.5817 increase in the car price.

car2 <- car\_data %>% select(aspiration, carbody, enginelocation, carwidth, curbweight,

enginetype, cylindernumber, enginesize, stroke, peakrpm, price)

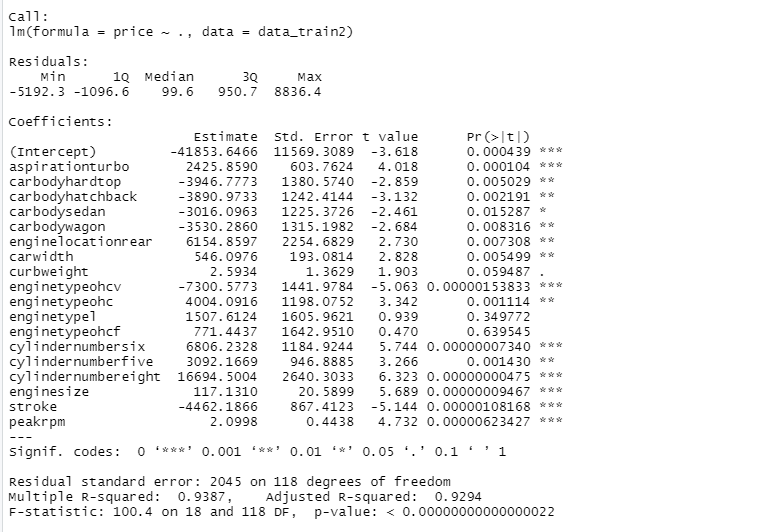
data\_train2 <- car2[index, ]

data\_test2 <- car2[-index, ]

set.seed(123)

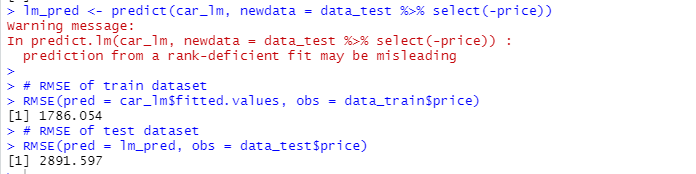
car\_lm2 <- lm(price ~ ., data = data\_train2)

summary(car\_lm2)

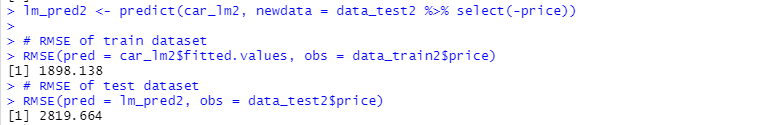


We have removed the non-significant variables. Too see if this action affect our model, we can check the Adjusted R-Squared value from our two previous models. The first model with complete variables has adjusted R-squared of 0.9277, meaning that the model can explain 92.77% of variance in the target variable (car price). Meanwhile, our simpler model has adjusted R-squared of 92.94%, no big difference with our first model. This shows that it is safe to remove variables that has no significant coefficient values.

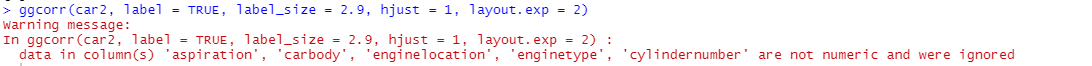
RMSE is better than MAE or mean absolute error, because RMSE squared the difference between the actual values and the predicted values, meaning that prediction with higher error will be penalized greatly. This metric is often used to compare two or more alternative models, even though it is harder to interpret than MAE. We can use the RMSE () functions from caret package. Below is the first model (with complete variables) performance.

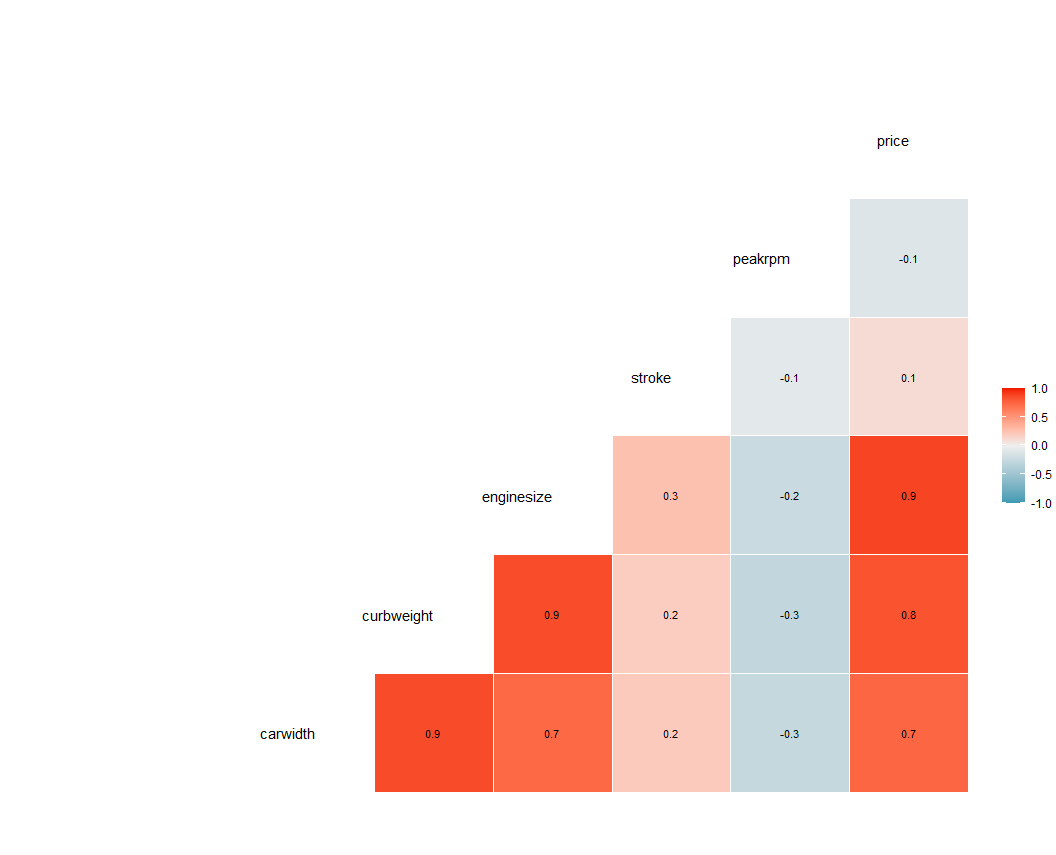


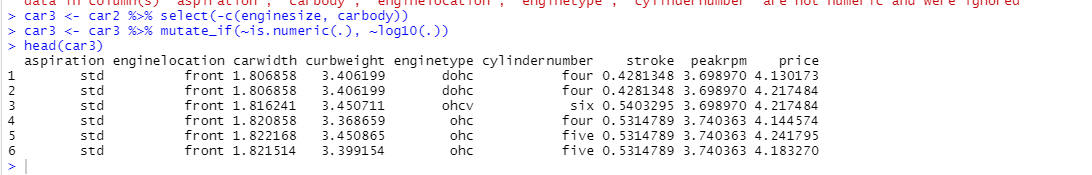
Below is the second model (with removed variables) performance.

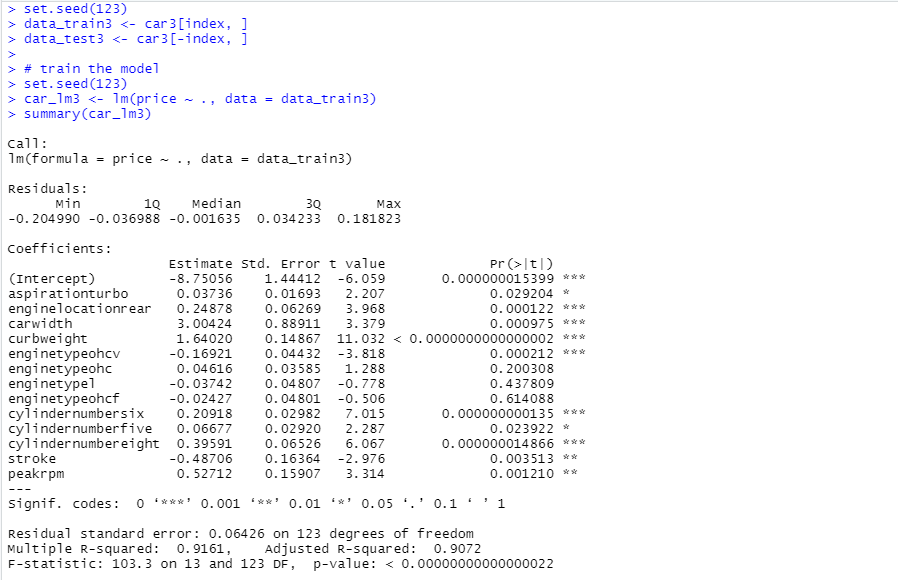


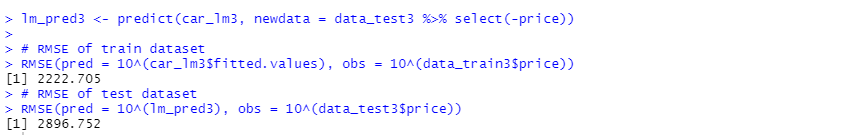
The second model is better than the first one in predicting the testing dataset, even only by a small margin. On both models, the error of the training dataset is lower than the test dataset, suggesting that our model may be overfit.











***CONCLUSION-***

The variables useful in significantly predicting the car price are:

* Aspiration
* Engine location
* Car width
* Car height
* Engine type
* Number of cyclinders
* Stroke
* Peak rpm

