```
#-------------------------------------DATA UNDERSTANDING, PREPARATION &
EDA-----------------------------------------------------------------------------------
# Read the data
carPrices <- read.csv("CarPrice_Assignment.csv",stringsAsFactors = F)

# View the data frame
View(carPrices)

# Understand structure of the data
str(carPrices)

# Chencking if data has any NA values
sum(is.na(carPrices))
# Conclusion: data has no NA values

# Checking if data has any missing values
sapply(carPrices,function(y) length(which(y=="")))
# Conclusion: data has no missing values


# Extracting the car company name and correcting the spelling errors
carPrices$CarName <- sapply(strsplit(tolower(carPrices$CarName), "\\s+"), "[",
1)
carPrices$CarName[which(carPrices$CarName=="maxda")] <- "mazda"
carPrices$CarName[which(carPrices$CarName=="porcshce")] <- "porsche"
carPrices$CarName[which(carPrices$CarName=="toyouta")] <- "toyota"
carPrices$CarName[which(carPrices$CarName=="vokswagen" |
carPrices$CarName=="vw")] <- "volkswagen"

library(ggplot2)
# Plotting the count based on car name
ggplot(carPrices,aes(CarName)) + geom_bar(col="black") + coord_flip()

# Plotting the price box plot based on car name
ggplot(carPrices,aes(CarName,price)) + geom_boxplot() + coord_flip()

# Displaying the count based on car prices
ggplot(carPrices,aes(price)) + geom_histogram(col="black")

# Displaying the count based on car body type
ggplot(carPrices,aes(carbody)) + geom_bar(col="black")

# Displaying the price box plot based on car body type
ggplot(carPrices,aes(carbody,price)) + geom_boxplot()



#----------------------------Outlier
Treatment----------------------------------
# Looking at all continuous variables

# wheelbase variable
```

```
# sudden jump from 99% - 100%
quantile(carPrices$wheelbase,seq(0,1,0.01))
boxplot.stats(carPrices$wheelbase)
carPrices$wheelbase <-
ifelse(carPrices$wheelbase>115.544,115.544,carPrices$wheelbase)

# carlength variable
# sudden jump from 0%-1%-2%-3% and 99% - 100%
quantile(carPrices$carlength,seq(0,1,0.01))
boxplot.stats(carPrices$carlength)
carPrices$carlength <-
ifelse(carPrices$carlength<155.9,155.9,carPrices$carlength)
carPrices$carlength <-
ifelse(carPrices$carlength>202.48,202.48,carPrices$carlength)

# carwidth variable
# outliers found at the higher end
quantile(carPrices$carwidth,seq(0,1,0.01))
boxplot.stats(carPrices$carwidth)
carPrices$carwidth <- ifelse(carPrices$carwidth>70.9,70.9,carPrices$carwidth)

# carheight variable
# no outliers
quantile(carPrices$carheight,seq(0,1,0.01))
boxplot.stats(carPrices$carheight)

# curbweigth variable
# no outliers
quantile(carPrices$curbweight,seq(0,1,0.01))
boxplot.stats(carPrices$curbweight)

# enginesize variable
# sudden jump from 94%-100%
quantile(carPrices$enginesize,seq(0,1,0.01))
boxplot.stats(carPrices$enginesize)
carPrices$enginesize <-
ifelse(carPrices$enginesize>194.0,194.0,carPrices$enginesize)

# boreratio variable
# no outliers
quantile(carPrices$boreratio,seq(0,1,0.01))
boxplot.stats(carPrices$boreratio)

# stroke variable
# outliers both ends
quantile(carPrices$stroke,seq(0,1,0.01))
boxplot.stats(carPrices$stroke)
carPrices$stroke <- ifelse(carPrices$stroke<2.68,2.68,carPrices$stroke)
carPrices$stroke <- ifelse(carPrices$stroke>3.86,3.86,carPrices$stroke)

# compressioratio variable
# Outliers are found. But, the ones at the higher end correspond to
compression ratio of diesel engines. Treating them will remove data pertaining
to diesel engines.
```

```
# hence not treating the outliers
quantile(carPrices$compressionratio,seq(0,1,0.01))
boxplot.stats(carPrices$compressionratio)

# horsepower variable
# treating outliers beyond 97%
quantile(carPrices$horsepower,seq(0,1,0.01))
boxplot.stats(carPrices$horsepower)
carPrices$horsepower <-
ifelse(carPrices$horsepower>184,184,carPrices$horsepower)

# peakrpm variable
# treating outliers beyond 99%
quantile(carPrices$peakrpm,seq(0,1,0.01))
boxplot.stats(carPrices$peakrpm)
carPrices$peakrpm <- ifelse(carPrices$peakrpm>6000,6000,carPrices$peakrpm)

# citympg variable
# treating outliers beyond 99%
quantile(carPrices$citympg,seq(0,1,0.01))
boxplot.stats(carPrices$citympg)
carPrices$citympg <- ifelse(carPrices$citympg>45,45,carPrices$citympg)

# highwaympg variable
# treating outliers beyond 99%
quantile(carPrices$highwaympg,seq(0,1,0.01))
boxplot.stats(carPrices$highwaympg)
carPrices$highwaympg <-
ifelse(carPrices$highwaympg>47,47,carPrices$highwaympg)

#----------------------------End of outlier
treatment--------------------------------------

#-----------------------------Derived
Metrics-----------------------------------------------

# Derived variable: totalLength = carlength+carwidth+carheight
carPrices$totalLength <- rowSums(carPrices[,11:13])
# Removing carlength, carwidth and carheight. These are highly correlated
variables and is captured in totalLength.
carPrices <- carPrices[,-(11:13)]

# Derived variable: powerToWeightRatio (https://en.wikipedia.org/wiki/Power-
to-weight_ratio)
carPrices$powerToWeightRatio <- carPrices$horsepower/carPrices$curbweight
# Removing horsepower and curbweight columns
carPrices <- carPrices[,-c(11,19)]


# Derived variable: strokeToBoreRatio (https://en.wikipedia.org/wiki/
Stroke_ratio)
carPrices$strokeToBoreRatio <- carPrices$stroke/carPrices$boreratio
# Removing stroke and boreratio columns
carPrices <- carPrices[,-(15:16)]
```

```
#----------------------------Creating dummy
variables--------------------------------------

# CarName variable
colnames(carPrices)[which(names(carPrices) == "CarName")] <- "CarName."
dummy_1 <- data.frame(model.matrix( ~CarName., data = carPrices))
carPrices <- cbind(carPrices[,-3], dummy_1[,-1])

# fueltype variable
carPrices$fueltype <- factor(carPrices$fueltype)
levels(carPrices$fueltype) <- c(0,1)
carPrices$fueltype <- as.numeric(levels(carPrices$fueltype))
[carPrices$fueltype]

# aspiration variable
carPrices$aspiration <- factor(carPrices$aspiration)
levels(carPrices$aspiration) <- c(1,0)
carPrices$aspiration <- as.numeric(levels(carPrices$aspiration))
[carPrices$aspiration]

# doornumber variable
carPrices$doornumber <- factor(carPrices$doornumber)
levels(carPrices$doornumber) <- c(1,0)
carPrices$doornumber <- as.numeric(levels(carPrices$doornumber))
[carPrices$doornumber]

# carbody variable
colnames(carPrices)[which(names(carPrices) == "carbody")] <- "carbody."
dummy_2 <- data.frame(model.matrix( ~carbody., data = carPrices))
carPrices <- cbind(carPrices[,-6], dummy_2[,-1])

# drivewheel variable
colnames(carPrices)[which(names(carPrices) == "drivewheel")] <- "drivewheel."
dummy_3 <- data.frame(model.matrix( ~drivewheel., data = carPrices))
carPrices <- cbind(carPrices[,-6], dummy_3[,-1])

# enginelocation variable
carPrices$enginelocation <- factor(carPrices$enginelocation)
levels(carPrices$enginelocation) <- c(1,0)
carPrices$enginelocation <- as.numeric(levels(carPrices$enginelocation))
[carPrices$enginelocation]

# enginetype variable
# reducing the number of levels by consolidating it into 2 levels as the count
for some are very low
summary(factor(carPrices$enginetype))
carPrices$enginetype[which(carPrices$enginetype=="dohcv")] <- "dohc"
carPrices$enginetype[which(carPrices$enginetype=="ohcf" |
carPrices$enginetype=="ohcv")] <- "ohc"
carPrices$enginetype[which(carPrices$enginetype=="rotor")] <- "dohc"
carPrices$enginetype[which(carPrices$enginetype=="l")] <- "dohc"
carPrices$enginetype <- factor(carPrices$enginetype)
```

```
levels(carPrices$enginetype) <- c(0,1)
carPrices$enginetype <- as.numeric(levels(carPrices$enginetype))
[carPrices$enginetype]

# cylindernumber variable
# reducing the number of levels by consolidating it into 3 levels as the count
for some are very low
summary(factor(carPrices$cylindernumber))
carPrices$cylindernumber[which(carPrices$cylindernumber=="two" |
carPrices$cylindernumber=="three")] <- "four"
carPrices$cylindernumber[which(carPrices$cylindernumber=="five")] <- "six"
carPrices$cylindernumber[which(carPrices$cylindernumber=="twelve")] <- "eight"
colnames(carPrices)[which(names(carPrices) == "cylindernumber")] <-
"cylindernumber."
dummy_4 <- data.frame(model.matrix( ~cylindernumber., data = carPrices))
carPrices <- cbind(carPrices[,-9], dummy_4[,-1])

# fuelsystem variable
# reducing the number of levels by consolidating it into 3 levels as the count
for some are very low
summary(factor(carPrices$fuelsystem))
carPrices$fuelsystem[which(carPrices$fuelsystem=="1bbl" |
carPrices$fuelsystem=="2bbl" | carPrices$fuelsystem=="4bbl")] <- "bbl"
carPrices$fuelsystem[which(carPrices$fuelsystem=="mfi" |
carPrices$fuelsystem=="spdi" | carPrices$fuelsystem=="spfi")] <- "mpfi"
colnames(carPrices)[which(names(carPrices) == "fuelsystem")] <- "fuelsystem."
dummy_5 <- data.frame(model.matrix( ~fuelsystem., data = carPrices))
carPrices <- cbind(carPrices[,-10], dummy_5[,-1])

#------------------------End of dummy variable
creation-----------------------------------------------------


# citympg and highwaympg are highly correlated (0.97) and represent the same
thing mpg. Hence only using citympg for analysis and removing highwaympg
cor(carPrices$highwaympg,carPrices$citympg)
carPrices <- carPrices[,-which(names(carPrices)=="highwaympg")]

# removing car_ID
carPrices <- carPrices[,-which(names(carPrices)=="car_ID")]


#-------------------------------------------------------------MODEL
BUILDING-------------------------------------------------------------------------------------

# separate training and testing data
set.seed(100)
trainindices= sample(1:nrow(carPrices), 0.7*nrow(carPrices))
train = carPrices[trainindices,]
test = carPrices[-trainindices,]

# Linear Regression
model_1 <-lm(price~.,data=train)
summary(model_1)
```

```
library(MASS)
library(car)

# Using stepAIC function
step <- stepAIC(model_1, direction="both")

# using variables suggested by stepAIC
model_2 <- lm(formula = price ~ symboling + fueltype + aspiration +
enginelocation +
                wheelbase + citympg + totalLength + strokeToBoreRatio +
CarName.audi +
                CarName.bmw + CarName.dodge + CarName.honda + CarName.isuzu +
                CarName.jaguar + CarName.mazda + CarName.mercury +
CarName.mitsubishi +
                CarName.nissan + CarName.peugeot + CarName.plymouth +
CarName.porsche +
                CarName.renault + CarName.saab + CarName.subaru +
CarName.toyota +
                CarName.volkswagen + CarName.volvo + carbody.hardtop +
carbody.hatchback +
                carbody.sedan + carbody.wagon + cylindernumber.four +
cylindernumber.six,
              data = train)
summary(model_2)
sort(vif(model_2),decreasing = T)


# removing wheelbase
model_3 <- lm(formula = price ~ symboling + fueltype + aspiration +
enginelocation +
                citympg + totalLength + strokeToBoreRatio + CarName.audi +
                CarName.bmw + CarName.dodge + CarName.honda + CarName.isuzu +
                CarName.jaguar + CarName.mazda + CarName.mercury +
CarName.mitsubishi +
                CarName.nissan + CarName.peugeot + CarName.plymouth +
CarName.porsche +
                CarName.renault + CarName.saab + CarName.subaru +
CarName.toyota +
                CarName.volkswagen + CarName.volvo + carbody.hardtop +
carbody.hatchback +
                carbody.sedan + carbody.wagon + cylindernumber.four +
cylindernumber.six,
              data = train)
summary(model_3)
sort(vif(model_3),decreasing = T)

# removing carbody.hatchback
model_4 <- lm(formula = price ~ symboling + fueltype + aspiration +
enginelocation +
                citympg + totalLength + strokeToBoreRatio + CarName.audi +
                CarName.bmw + CarName.dodge + CarName.honda + CarName.isuzu +
                CarName.jaguar + CarName.mazda + CarName.mercury +
CarName.mitsubishi +
```

```
                CarName.nissan + CarName.peugeot + CarName.plymouth +
CarName.porsche +
                CarName.renault + CarName.saab + CarName.subaru +
CarName.toyota +
                CarName.volkswagen + CarName.volvo + carbody.hardtop +
carbody.sedan + carbody.wagon + cylindernumber.four + cylindernumber.six,
              data = train)
summary(model_4)
sort(vif(model_4),decreasing = T)


# removing CarName.honda
model_5 <- lm(formula = price ~ symboling + fueltype + aspiration +
enginelocation +
                citympg + totalLength + strokeToBoreRatio + CarName.audi +
                CarName.bmw + CarName.dodge + CarName.isuzu +
                CarName.jaguar + CarName.mazda + CarName.mercury +
CarName.mitsubishi +
                CarName.nissan + CarName.peugeot + CarName.plymouth +
CarName.porsche +
                CarName.renault + CarName.saab + CarName.subaru +
CarName.toyota +
                CarName.volkswagen + CarName.volvo + carbody.hardtop +
carbody.sedan + carbody.wagon + cylindernumber.four + cylindernumber.six,
              data = train)
summary(model_5)
sort(vif(model_5),decreasing = T)


# removing symboling
model_6 <- lm(formula = price ~ fueltype + aspiration + enginelocation +
                citympg + totalLength + strokeToBoreRatio + CarName.audi +
                CarName.bmw + CarName.dodge + CarName.isuzu +
                CarName.jaguar + CarName.mazda + CarName.mercury +
CarName.mitsubishi +
                CarName.nissan + CarName.peugeot + CarName.plymouth +
CarName.porsche +
                CarName.renault + CarName.saab + CarName.subaru +
CarName.toyota +
                CarName.volkswagen + CarName.volvo + carbody.hardtop +
carbody.sedan + carbody.wagon + cylindernumber.four + cylindernumber.six,
              data = train)
summary(model_6)
sort(vif(model_6),decreasing = T)


# removing carbody.hardtop
model_7 <- lm(formula = price ~ fueltype + aspiration + enginelocation +
                citympg + totalLength + strokeToBoreRatio + CarName.audi +
                CarName.bmw + CarName.dodge + CarName.isuzu +
                CarName.jaguar + CarName.mazda + CarName.mercury +
CarName.mitsubishi +
                CarName.nissan + CarName.peugeot + CarName.plymouth +
CarName.porsche +
```

```
                CarName.renault + CarName.saab + CarName.subaru +
CarName.toyota +
                CarName.volkswagen + CarName.volvo + carbody.sedan +
carbody.wagon + cylindernumber.four + cylindernumber.six,
              data = train)
summary(model_7)
sort(vif(model_7),decreasing = T)



# removing CarName.volvo
model_8 <- lm(formula = price ~ fueltype + aspiration + enginelocation +
                citympg + totalLength + strokeToBoreRatio + CarName.audi +
                CarName.bmw + CarName.dodge + CarName.isuzu +
                CarName.jaguar + CarName.mazda + CarName.mercury +
CarName.mitsubishi +
                CarName.nissan + CarName.peugeot + CarName.plymouth +
CarName.porsche +
                CarName.renault + CarName.saab + CarName.subaru +
CarName.toyota +
                CarName.volkswagen + carbody.sedan + carbody.wagon +
cylindernumber.four + cylindernumber.six,
              data = train)
summary(model_8)
sort(vif(model_8),decreasing = T)


# removing CarName.audi
model_9 <- lm(formula = price ~ fueltype + aspiration + enginelocation +
                citympg + totalLength + strokeToBoreRatio + CarName.bmw +
CarName.dodge + CarName.isuzu +
                CarName.jaguar + CarName.mazda + CarName.mercury +
CarName.mitsubishi +
                CarName.nissan + CarName.peugeot + CarName.plymouth +
CarName.porsche +
                CarName.renault + CarName.saab + CarName.subaru +
CarName.toyota +
                CarName.volkswagen + carbody.sedan + carbody.wagon +
cylindernumber.four + cylindernumber.six,
              data = train)
summary(model_9)
sort(vif(model_9),decreasing = T)


# removing CarName.mercury
model_10 <- lm(formula = price ~ fueltype + aspiration + enginelocation +
                citympg + totalLength + strokeToBoreRatio + CarName.bmw +
CarName.dodge + CarName.isuzu +
                CarName.jaguar + CarName.mazda + CarName.mitsubishi +
                CarName.nissan + CarName.peugeot + CarName.plymouth +
CarName.porsche +
                CarName.renault + CarName.saab + CarName.subaru +
CarName.toyota +
```

```
                CarName.volkswagen + carbody.sedan + carbody.wagon +
cylindernumber.four + cylindernumber.six,
              data = train)
summary(model_10)
sort(vif(model_10),decreasing = T)


# removing CarName.saab
model_11 <- lm(formula = price ~ fueltype + aspiration + enginelocation +
                 citympg + totalLength + strokeToBoreRatio + CarName.bmw +
CarName.dodge + CarName.isuzu +
                 CarName.jaguar + CarName.mazda + CarName.mitsubishi +
                 CarName.nissan + CarName.peugeot + CarName.plymouth +
CarName.porsche +
                 CarName.renault + CarName.subaru + CarName.toyota +
                 CarName.volkswagen + carbody.sedan + carbody.wagon +
cylindernumber.four + cylindernumber.six,
               data = train)
summary(model_11)
sort(vif(model_11),decreasing = T)


# removing CarName.renault
model_12 <- lm(formula = price ~ fueltype + aspiration + enginelocation +
                 citympg + totalLength + strokeToBoreRatio + CarName.bmw +
CarName.dodge + CarName.isuzu +
                 CarName.jaguar + CarName.mazda + CarName.mitsubishi +
                 CarName.nissan + CarName.peugeot + CarName.plymouth +
CarName.porsche +
                 CarName.subaru + CarName.toyota + CarName.volkswagen +
carbody.sedan + carbody.wagon + cylindernumber.four + cylindernumber.six,
               data = train)
summary(model_12)
sort(vif(model_12),decreasing = T)


# removing CarName.dodge
model_13 <- lm(formula = price ~ fueltype + aspiration + enginelocation +
                 citympg + totalLength + strokeToBoreRatio + CarName.bmw +
CarName.isuzu +
                 CarName.jaguar + CarName.mazda + CarName.mitsubishi +
                 CarName.nissan + CarName.peugeot + CarName.plymouth +
CarName.porsche +
                 CarName.subaru + CarName.toyota + CarName.volkswagen +
carbody.sedan + carbody.wagon + cylindernumber.four + cylindernumber.six,
               data = train)
summary(model_13)
sort(vif(model_13),decreasing = T)


# removing CarName.nissan
model_14 <- lm(formula = price ~ fueltype + aspiration + enginelocation +
                 citympg + totalLength + strokeToBoreRatio + CarName.bmw +
CarName.isuzu +
```

```
                 CarName.jaguar + CarName.mazda + CarName.mitsubishi +
                 CarName.peugeot + CarName.plymouth + CarName.porsche +
                 CarName.subaru + CarName.toyota + CarName.volkswagen +
carbody.sedan + carbody.wagon + cylindernumber.four + cylindernumber.six,
               data = train)
summary(model_14)
sort(vif(model_14),decreasing = T)


# removing CarName.plymouth
model_15 <- lm(formula = price ~ fueltype + aspiration + enginelocation +
                 citympg + totalLength + strokeToBoreRatio + CarName.bmw +
CarName.isuzu +
                 CarName.jaguar + CarName.mazda + CarName.mitsubishi +
                 CarName.peugeot + CarName.porsche + CarName.subaru +
CarName.toyota + CarName.volkswagen + carbody.sedan + carbody.wagon +
cylindernumber.four + cylindernumber.six,
               data = train)
summary(model_15)
sort(vif(model_15),decreasing = T)


# removing CarName.mitsubishi
model_16 <- lm(formula = price ~ fueltype + aspiration + enginelocation +
                 citympg + totalLength + strokeToBoreRatio + CarName.bmw +
CarName.isuzu +
                 CarName.jaguar + CarName.mazda +
                 CarName.peugeot + CarName.porsche + CarName.subaru +
CarName.toyota + CarName.volkswagen + carbody.sedan + carbody.wagon +
cylindernumber.four + cylindernumber.six,
               data = train)
summary(model_16)
sort(vif(model_16),decreasing = T)


# removing CarName.isuzu
model_17 <- lm(formula = price ~ fueltype + aspiration + enginelocation +
                 citympg + totalLength + strokeToBoreRatio + CarName.bmw +
CarName.jaguar + CarName.mazda +
                 CarName.peugeot + CarName.porsche + CarName.subaru +
CarName.toyota + CarName.volkswagen + carbody.sedan + carbody.wagon +
cylindernumber.four + cylindernumber.six,
               data = train)
summary(model_17)
sort(vif(model_17),decreasing = T)


# removing CarName.mazda
model_18 <- lm(formula = price ~ fueltype + aspiration + enginelocation +
                 citympg + totalLength + strokeToBoreRatio + CarName.bmw +
CarName.jaguar +
                 CarName.peugeot + CarName.porsche + CarName.subaru +
CarName.toyota + CarName.volkswagen + carbody.sedan + carbody.wagon +
cylindernumber.four + cylindernumber.six,
```

```
               data = train)
summary(model_18)
sort(vif(model_18),decreasing = T)


# removing CarName.volkswagen
model_19 <- lm(formula = price ~ fueltype + aspiration + enginelocation +
                 citympg + totalLength + strokeToBoreRatio + CarName.bmw +
CarName.jaguar +
                 CarName.peugeot + CarName.porsche + CarName.subaru +
CarName.toyota + carbody.sedan + carbody.wagon + cylindernumber.four +
cylindernumber.six,
               data = train)
summary(model_19)
sort(vif(model_19),decreasing = T)


# removing CarName.porsche
model_20 <- lm(formula = price ~ fueltype + aspiration + enginelocation +
                 citympg + totalLength + strokeToBoreRatio + CarName.bmw +
CarName.jaguar +
                 CarName.peugeot + CarName.subaru + CarName.toyota +
carbody.sedan + carbody.wagon + cylindernumber.four + cylindernumber.six,
               data = train)
summary(model_20)
sort(vif(model_20),decreasing = T)

# removing CarName.subaru
model_21 <- lm(formula = price ~ fueltype + aspiration + enginelocation +
                 citympg + totalLength + strokeToBoreRatio + CarName.bmw +
CarName.jaguar +
                 CarName.peugeot + CarName.toyota + carbody.sedan +
carbody.wagon + cylindernumber.four + cylindernumber.six,
               data = train)
summary(model_21)
sort(vif(model_21),decreasing = T)


# removing strokeToBoreRatio
model_22 <- lm(formula = price ~ fueltype + aspiration + enginelocation +
                 citympg + totalLength + CarName.bmw + CarName.jaguar +
                 CarName.peugeot + CarName.toyota + carbody.sedan +
carbody.wagon + cylindernumber.four + cylindernumber.six,
               data = train)
summary(model_22)
sort(vif(model_22),decreasing = T)


# removing fueltype
model_23 <- lm(formula = price ~ aspiration + enginelocation +
                 citympg + totalLength + CarName.bmw + CarName.jaguar +
                 CarName.peugeot + CarName.toyota + carbody.sedan +
carbody.wagon + cylindernumber.four + cylindernumber.six,
               data = train)
```

```
summary(model_23)
sort(vif(model_23),decreasing = T)


# removing citympg
model_24 <- lm(formula = price ~ aspiration + enginelocation +
                 totalLength + CarName.bmw + CarName.jaguar +
                 CarName.peugeot + CarName.toyota + carbody.sedan +
carbody.wagon + cylindernumber.four + cylindernumber.six,
               data = train)
summary(model_24)
sort(vif(model_24),decreasing = T)


# removing CarName.toyota
model_25 <- lm(formula = price ~ aspiration + enginelocation +
                 totalLength + CarName.bmw + CarName.jaguar +
                 CarName.peugeot + carbody.sedan + carbody.wagon +
cylindernumber.four + cylindernumber.six,
               data = train)
summary(model_25)
sort(vif(model_25),decreasing = T)


# removing CarName.peugeot
model_26 <- lm(formula = price ~ aspiration + enginelocation +
                 totalLength + CarName.bmw + CarName.jaguar +
                 carbody.sedan + carbody.wagon + cylindernumber.four +
cylindernumber.six,
               data = train)
summary(model_26)
sort(vif(model_26),decreasing = T)

# Conclusion: All the variables are p-value significant. Henc model_26 is the
final model.
# Driver variables: aspiration, enginelocation, totalLength, CarName.bmw,
CarName.jaguar, carbody.sedan, carbody.wagon, cylindernumber.four &
cylindernumber.six

# Final Adjusted R-squared value: 0.9339

#-----------------------------------------------------------------------MODEL
EVALUATION-------------------------------------------------------------------------------

# predicting the results in test dataset
Predict_test <- predict(model_26,test[,-12])
r <- cor(test$price,Predict_test)
rsquared <- cor(test$price,Predict_test)^2
rsquared
# Conclusion: R-squared value with the test data is 0.78

# Plotting original price & predicted price
ggplot(test, aes(1:nrow(test),price)) + geom_line(colour="red") +
geom_line(colour="blue",aes(1:nrow(test),Predict_test))
```

```
# Conclusion: Overall the model is not accurate with the peaks.


# Plotting error
ggplot(test, aes(1:nrow(test), price-Predict_test)) + geom_point(colour =
"blue" ) + geom_hline(yintercept = 0) + xlab("Index") + ylab("Error")
# Conclusion: The error's are randomly distributed. This confirms that there
are no variables that could have helped explain the model better.


#-----------------------------------------------------------------MODEL EXPLANATION &
INTERPRETATION-----------------------------------------------------------------------------



# Driver variable        Beta values
#-------------------------------------
#  aspiration              -3313.4
#  enginelocation          -19688.4
#   totalLength             254.1
#  CarName.bmw              8669.3
#  CarName.jaguar           11475.0
#  carbody.sedan           -1690.8
#  carbody.wagon           -3105.1
#  cylindernumber.four     -22319.2
#  cylindernumber.six      -16653.5

# The -ve beta value means it reduces the price of the car
# The +ve beta value means it increases the price of the car
# aspiration: "std" option reduces the price of car and "turbo" option makes
this variable 0
# enginelocation: "front" option reduces the price of car and "rear" option
makes this variable 0
# totalLength: increases the price of the car
# CarName.bmw: increases the price of the car
# CarName.jaguar: increases the price of the car
# carbody.sedan: reduces the price of the car when its a sedan, 0 otherwise
# carbody.wagon: reduces the price of the car when its a wagon, 0 otherwise
# cylindernumber.four: reduces the price of the car when it has four
cylinders, 0 otherwise
# cylindernumber.six: reduces the price of the car when it has six cylinders,
0 otherwise

# Based on the above info Geely Auto should design and build cars to target a
particular price range
```