INSURANCE RISK OF CARS

MGSC401 – Professor Juan SERPA – FINAL PROJECT

Karen Bou Daou (260957944)

DECEMBER 15TH, 2022

McGill | DESAUTELS

Table of Contents

I. Introduction

Have you ever wondered how insurance companies determine their risk rating and charge you for your car insurance? It is quite interesting to discover the reasoning behind it which is the goal of this project. The insurance risk rating of a car is an important factor to consider when one purchases a car. It measures the likelihood of a car being involved in an accident that may result in a claim being made by the insurance company. Hence, it is also used the determine the amount of premium one will be charged when purchasing car insurance. The higher the rate, the riskier the car is (and the higher the insurance premiums) and the lower the rate, the safer (which results in lower premiums). To determine a car's insurance risk rating, several factors are taken into consideration by insurance companies: the make and model of the car, its safety ratings but also its performance characteristics (overall reliability), but which one has more impact over the other? Do they all matter equally? How do insurance companies determine a car's risk rating?

Given the dataset provided, this project aims to analyze factors that affect the insurance risk rating of a car by focusing on its performance and characteristics. The goal is to uncover which variables have the greatest impact on the insurance risk rating to better understand the process by which insurance companies determine rates. To conduct this analysis, we will use the dataset on automobiles and their characteristics and build a predictive model to assess the insurance risk rating of cars. This can help us better understand the insurance market and help us make informed decisions when purchasing car insurance. We will then compare the results of this analysis to industry standards in order to draw meaningful conclusions about the insurance risk rating of cars.

II. Data Description

The dataset was collected from Kaggle and consists of diverse characteristics of an automobile, along with a symbol representing its risk rating and a variable of normalized losses. It was extracted from Ward's Automotive Yearbook in 1985 and contains a total of 26 variables and 205 observations. Our dependent variable is the symbolling of a car as it represents its insurance risk rating: the higher the symbol, the riskier it is and the lower, the safer.

Numerical Data Description

The important numerical variables are visualized through a PCA analysis. The data is first separated into labels and variables to allow the performance of PCA. The Principal Component Analysis technique was used to reduce the dimension of the dataset and visualize the relationship between these variables (Appendix 1).

Given the plot computed with the first two principal components, we have found a sense of how the characteristics of a car are related. Some variables are strongly correlated to one another such as the curb weight of a car, its engine size, length, width, wheelbase, and price. We could say that the most significant variables are in a car's dimensions and weights, which made up the first component. The heavier and bigger the car is, the pricier it is (this could be related to the engine that is pricier given its size). Then, we noticed a strong correlation between compression ratio and height. Hence, we can assume that the higher the car, the more engine power it contains. The second component also consists of a high correlation between the peak rpm (revolution per minute) of a car and its normalized losses. We can thus interpret that the higher the rpm of a car (acceleration and top speed), the more money an insurance company pays out in claims relative to the premium that it collects.

A further step was taken at visualizing the PCA plot by separating the observations in the PCA by the extreme levels of our dependent variable: symbolling (Appendix 2). We notice that safer cars are mostly impacted by their dimensions and weight (symbolling = -2) whereas riskier cars are mostly impacted by their horsepower, normalized losses, and peak rpm (symbolling = 3) which is pertinent: the more accelerated a car is the riskier it could be.

The important variables taken from the PCA analysis were then plotted in relationship with the target variable. As some variables were highly correlated, we decided to analyze the relationship of one variable from each correlation with the target variable (Appendix 3):

- Wheelbase: this variable is measured either in inches or millimetres and varies between a value of 85 and 120. It represents the distance between the front and rear wheels of a car. It also determines the size and weight of a car which is why we decide to choose it among the variables highly correlated. Given the scatter plot, we notice that the smaller the wheelbase of a car, the higher its insurance risk rating.

In fact, a shorter wheelbase makes a car more agile which is why it could be riskier than a longer one as it has less stability.

- Normalized losses: this variable represents the expected losses that an insurance company will incur. Its values vary between 50 and 150. We notice from the scatter plot that the higher the normalized losses for a car, the higher its insurance risk rating.

Categorical Data Description

Regarding the categorical variables, we created a ggplot for the different variables to visualize them.

The first distribution is the different rates of insurance risk. Symboling is a discrete variable. It is our dependent variable, and it ranges from -2 to 3 with negative values representing safer cars and positive values, riskier cars with a higher insurance risk rating. The plot highlights that our dataset is mostly comprised of cars at levels 0 and -1 (approximately 77 cars) followed by cars at levels 1 and 2 (count of approximately 75 cars). Hence, they are evenly distributed among different levels, and this is important in our analysis (Appendix 4).

Next is the distribution of car manufacturers (Appendix 5). We notice a higher amount of Toyota cars in the dataset (31 cars), followed by Nissan (18 cars). Then, we have an approximately equal amount of Honda (13 cars), Subaru (12 cars), Mazda and Volvo (11 cars each) and Mitsubishi (10 cars). More luxurious cars such as Audis, BMWs and Porsches are less significant in the dataset.

The distribution of the number of doors in a car (Appendix 6) depicts 95 cars with four doors and 64 cars with two doors. Moreover, the distribution of body styles illustrates 79 sedan cars, followed by 56 hatchbacks, 17 wagons, 5 hardtops and 2 convertibles (Appendix 7). Concerning the distribution of drive wheels, the dataset contains of 105 front-wheel drive cars, 46 rear-wheel drive cars and 8 four-wheel drive cars (Appendix 8). The car's fuel systems – which deliver fuel from the tank to the engine, are of different types: 64 cars correspond to the multi-point fuel injection 'mpfi' type, 63 to the two Barrels carburetor '2bbl' type, 15 to the indirect injection 'idi' type, 11 to the single barrel carburetor '1bbl' type and 5 to the plasma discharge inition 'spdi' type (Appendix 9).

Finally, the variables of fuel type, aspiration, engine location and type and number of cylinders (Appendix 10) were less varied than the other variables so I deemed them to be less relevant for the analysis.

III. Model selection and Methodology

Given our PCA results, we got many variables that were highly correlated. Hence, I decided to perform a Random Forest Model because it takes a random subset of predictors in each tree and reduces biases due to multicollinearity. It is also more effective than discriminant analysis algorithms because it does not make any assumptions about the data, it can model non-linear relationships and handle a large number of variables and classes in a more effective way to determine the important predictors. Hence, it does not require dummification of the categorical variables, therefore it enables me to interpret them efficiently.

I decided to insert all the variables into the model to visualize the relative predictive performance of my variables and plot them in importance order (Appendix 11). I ran 500 random trees on 500 bootstrapped samples, and I found an OOB estimate error rate of 10.69% which is suitable (the lower the OOB rate, the better it is for my model).

$$myforest = randomForest(symboling \sim normalized.losses + make + fuel.type + aspiration + num.of.doors + body.style + drive.wheels + engine.location + wheel.base + length + width + height + curb.weight + engine.type + num.of.cylinders + engine.size + fuel.system + bore + stroke + compression.ratio + horsepower + peak.rpm + city.mpg + highway.mpg + price, ntree = 500, data = data, importance = TRUE, na.action = na.omit)$$

Given the results, my important variables in order are the make of a car (its manufacturer company), followed by the normalized losses, the wheelbase, the width and the height.

I then performed another random forest and selected the five first important variables found to test if I would get a better predictive performance. I added a tree size function ‘cp’ to minimize the error rate. It was in fact decreased from 10.69% to 8.81% which is even more advantageous.

$$classifiedforest = randomForest(symboling \sim make + normalized.losses + wheel.base + width + height, cp = 0.01, na.action = na.omit)$$

After visualizing the important variables and performance of my tree, I decided to predict the symbol of a car by using the important variables from my random forest model and running a classification tree. I began by performing cross-validation to find the best tree and inserted a very small value of cp = 0.001 (Appendix 12).

$$myoverfittedtree = rpart(symboling \sim make + normalized.losses + wheel.base + width + height, control = rpart.control(cp = 0.001))$$

I then plotted the tree and studied the out-of-sample performance to find the optimal cp value – that minimizes the error and avoids overfitting (Appendix 13). I found an optimal value of 1e-06. My tree did not change in comparison with the overfitted tree (Appendix 14). This makes sense as I inserted the most important variables in my model which optimized its performance and results.

$$classifiedtree2 = rpart(symboling \sim make+normalized.losses+wheel.base+ width + height, cp = opt_cp, na.action = na.omit)$$

Finally, to make my project more interesting and visualize the accuracy of my predictions, I decided to predict a car's insurance risk rating with random values and found Chevrolet's symbol to be of level 2 (high insurance risk).

$$predict(classifiedtree2, data.frame(make =' chevrolet', normalized.losses = 89, wheel.base = 90, length = 125, width = 58, height = 51))$$

Throughout the analysis, I encountered many errors due to missing values which is why I initiated the code by replacing the '?' characters with 'NA' and proceeded with the removal of all missing values.

IV. Results

Given the results of my classification trees, I was able to predict a car's insurance risk rating with significant variables (Appendix 14). The risk rating ranges between -2 and 3 from safest

to riskiest. The most important predictor is the wheelbase and the second most important is the manufacturer of the car, followed by the normalized losses.

If the wheelbase is higher or equal to 95, the second most important predictor is the manufacturer of the car which consists of 70% of the total observations.

- If the car manufacturer is BMW, Honda, Jaguar, Mazda, Nissan, Peugeot or Subaru, then the risk of insurance is 0 (26% of observations).

If the observation does not belong to any of these manufacturers, then we look at the wheelbase again:

- If it's superior or equal to 101 and normalized losses are less than 99, we get 12% of the observations at an insurance risk rating of -1.
- However, if the normalized losses are superior to 99, the car is considered even safer with an insurance risk rating of -2 (8% of observations).

If the wheelbase is lower than 97 and the car manufacturer is a Toyota, the insurance risk rating is 0 and consists of 8% of the observations. On another hand, if the car is not a Toyota, the insurance risk rating is 3 for 5% of the observations which is the riskiest. However, if the wheelbase is bigger than 97, its insurance risk rating is 2 (second riskiest) with 13% of observations.

Looking back at the principal predictor, if the wheelbase is less than 95, we look at the manufacturer of the car (consisting of 30% of the total observations):

- If the car is manufactured by Dodge, Mazda, Nissan, Plymouth or Toyota, the insurance risk rating is 1 for 19% of the observations.
- If it is not manufactured by these companies, the insurance risk rating is higher with a value of 2 for 11% of the observations.

V. Interpretation and Conclusion

All in all, an interesting observation I have made throughout the analysis is that the wheelbase of a car is the most important predictor. One would not expect it to be a significant factor as it only represents the distance between the centers of the front and rear wheels. However, from a different point of view, the wheelbase of a car also affects its overall length, width and

height and thus can be related to risk because the wheelbase of a car can affect its handling and stability which can in turn affect the risk of an accident and increase the insurance's risk rating.

Given the plot, longer wheelbase cars would fall into an insurance risk rating of -2 and -1 which are the safest. This makes sense as a longer wheelbase provides more space between the front and rear wheels which enables better stability and easier control of the car, a smoother ride and helps distribute the impact of a collision which reduces the risk of damage to the car. Although a shorter wheelbase can make a car more agile, it also makes it less stable and is therefore more susceptible to rollover in certain situations. This explains why some of the short-wheelbase cars were found to be risky in our tree. However, it is not the only factor that determines the insurance risk, the make of the model comes in next.

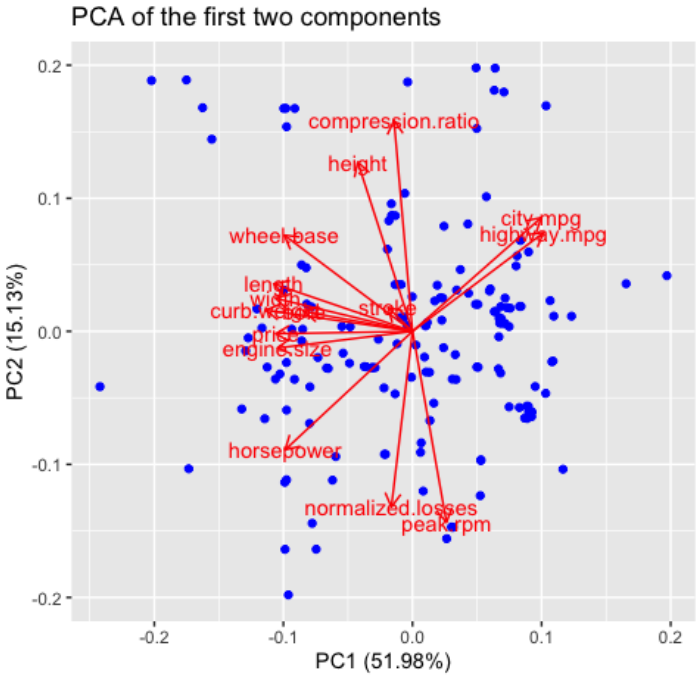
We now know that the make of the model affects its insurance risk rating. They differ in their models which is why they provide different levels of insurance risk rating in the tree. In fact, some cars may be more expensive to repair than others, some may be more prone to theft as well. These factors explain why different cars have different insurance risk ratings.

Finally, after taking into consideration the wheelbase and manufacturer of the car. The third predictor we should look at is the normalized loss of a car. In general, the higher the normalized losses for a car, the higher its insurance risk rating. Although in our tree we can see that a higher normalized loss leads to a lower insurance risk rating, we can assume that it was classified this way because the model has taken the two important predictors into consideration first. Nonetheless, it remains an important variable to acknowledge because it represents the expected losses that an insurance company anticipates for a given model of a car and thus explains why it affects its insurance risk rating: insurance companies typically charge a higher premium for the model of a car to compensate for the increased risk.

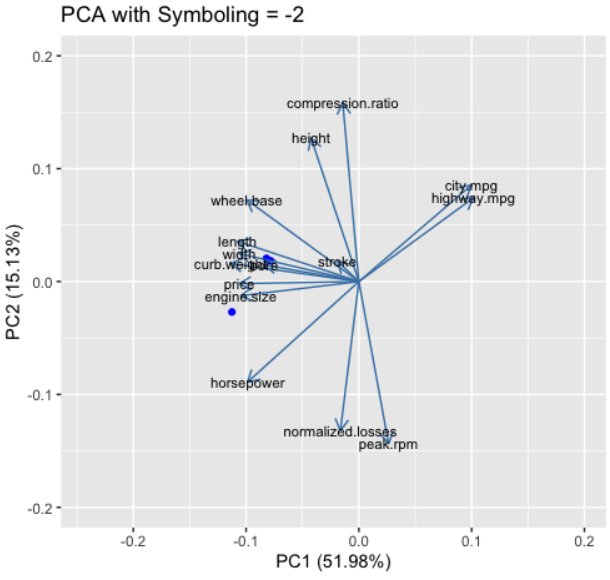
We can conclude that when evaluating the insurance risk rating for a car, its characteristics and more specifically its wheelbase, along with the manufacturer company and the normalized losses are the most important predictors to take into consideration. This is how insurance companies determine a car's insurance risk rating.

VI. Appendices

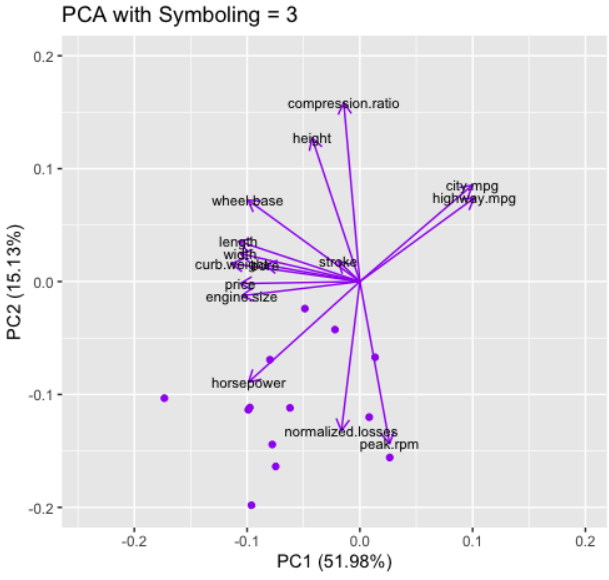
Appendix 1: PCA of the first two components



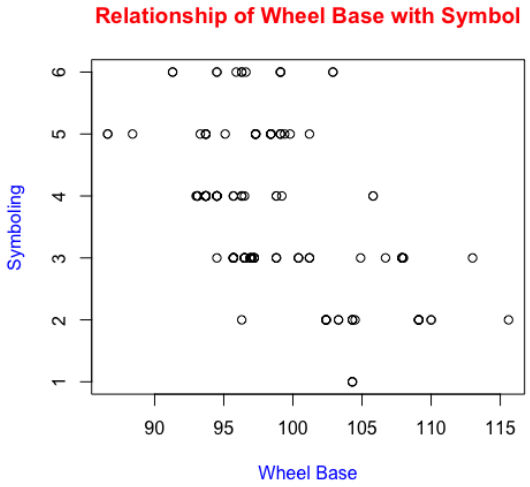
Appendix 2: PCA of symbol = -2 (safest)

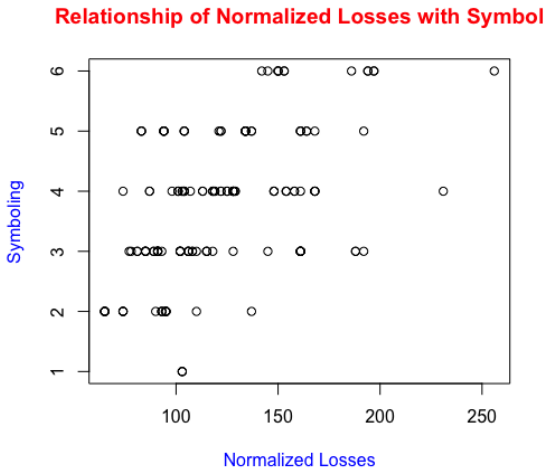


PCA of symbol = 3 (riskiest)

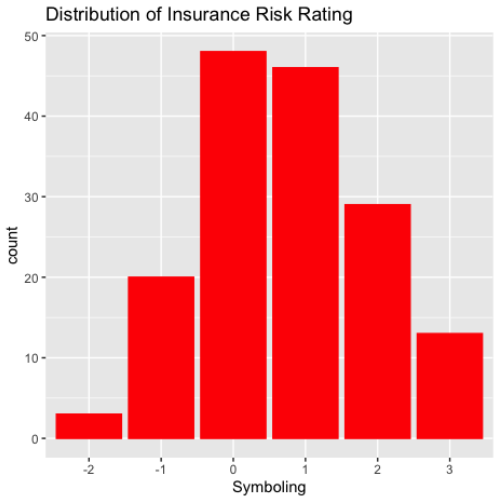


Appendix 3: Relationship of Wheel Base with Symbol and Normalized Losses with Symbol

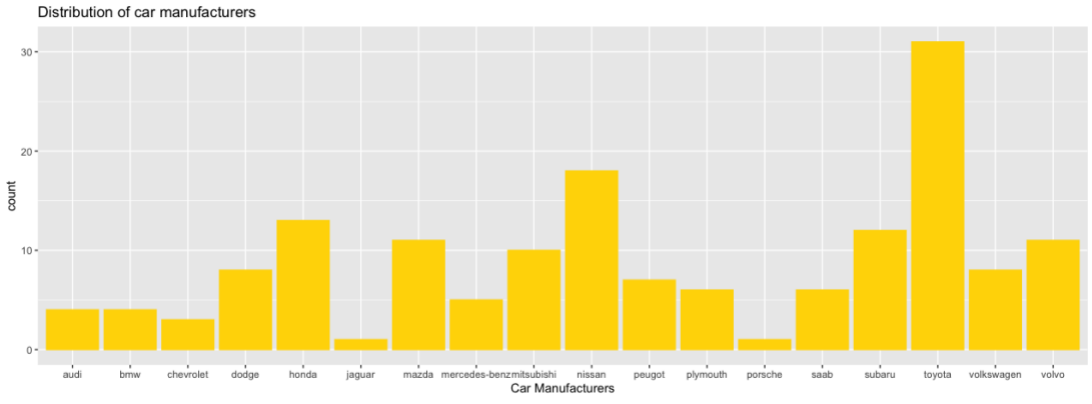




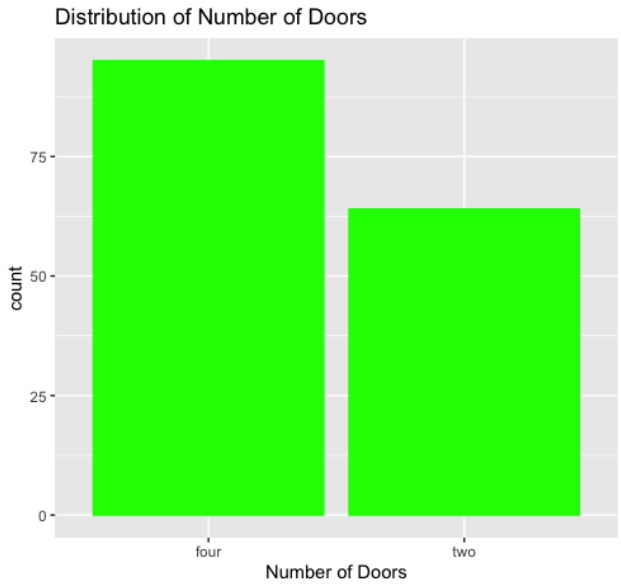
Appendix 4: Distribution of Insurance Risk Ratings (Symboling)



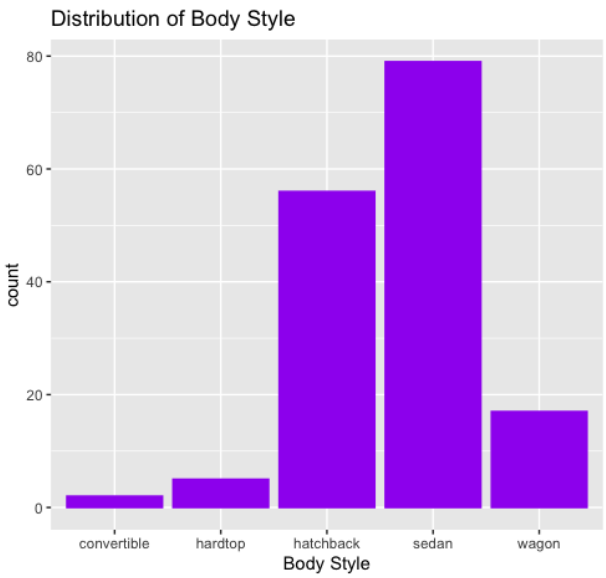
Appendix 5: Distribution of Car Manufacturers



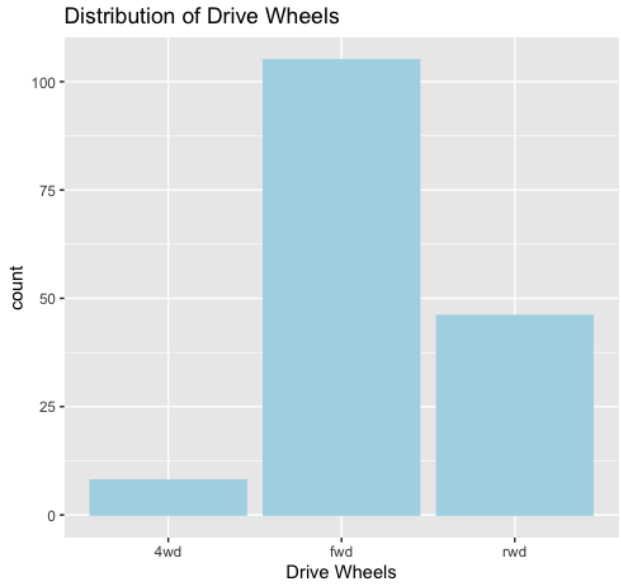
Appendix 6: Distribution of Number of Doors



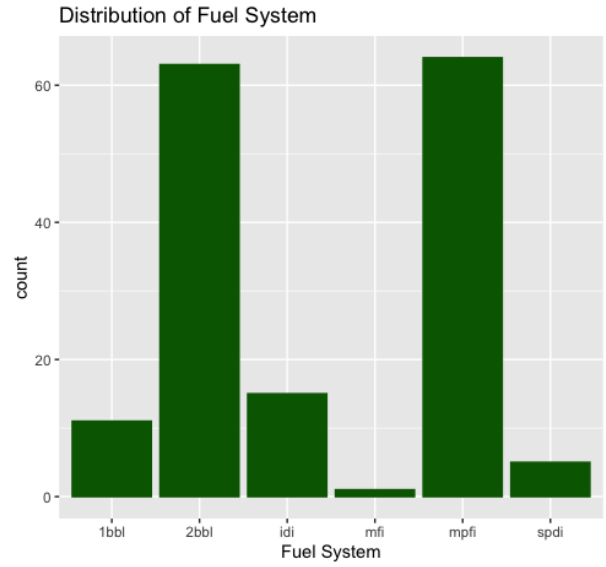
Appendix 7: Distribution of Body Style



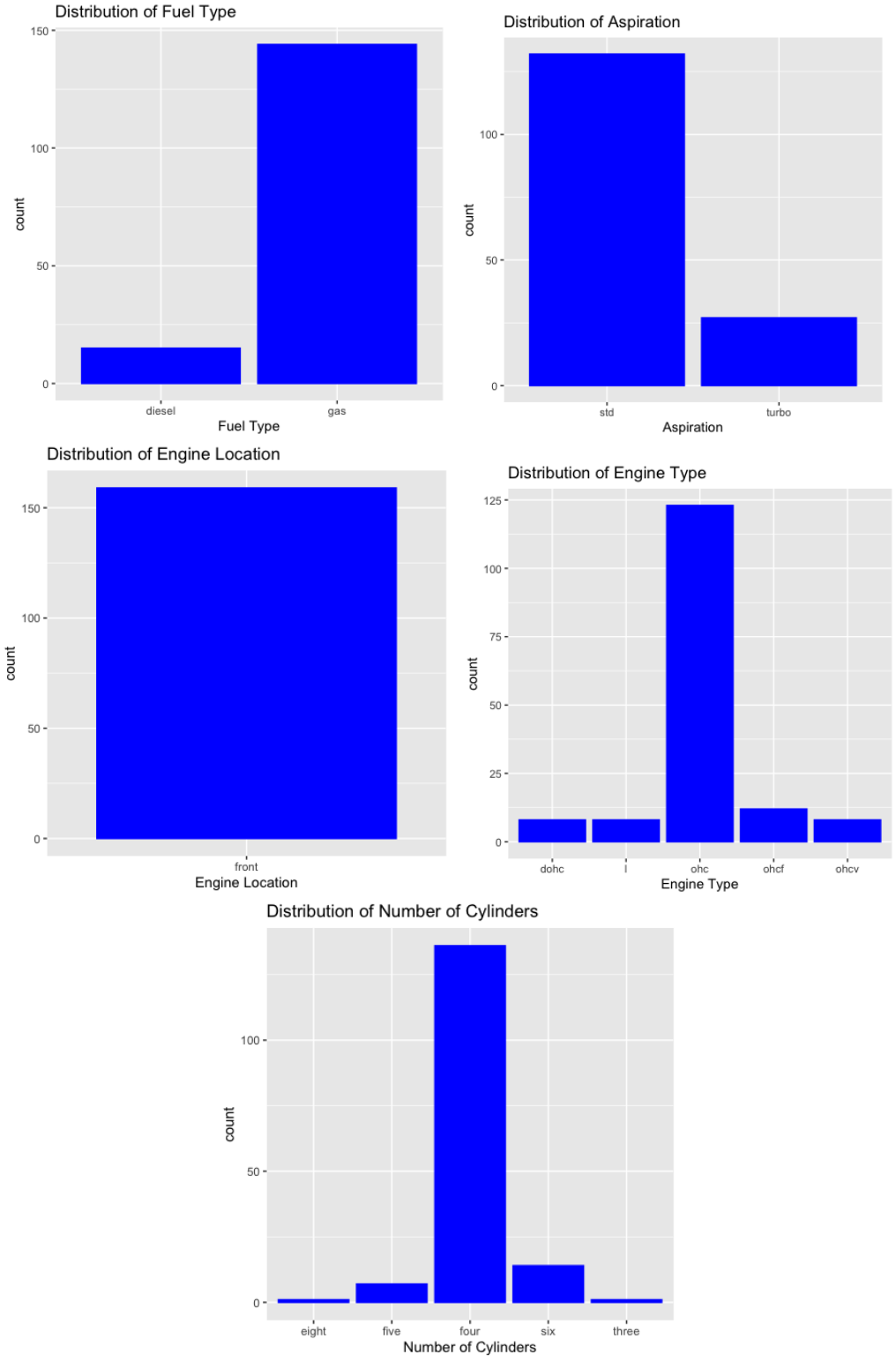
Appendix 8: Distribution of Drive Wheels



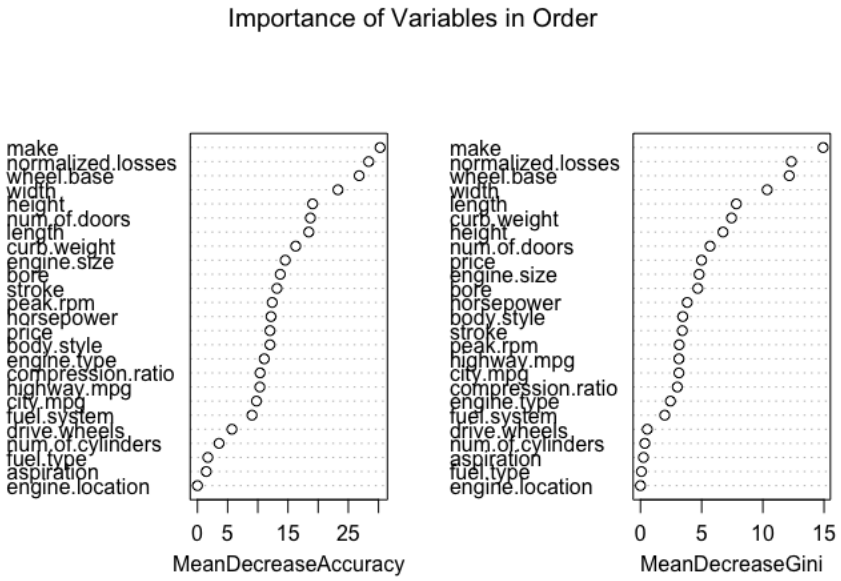
Appendix 9: Distribution of Fuel System



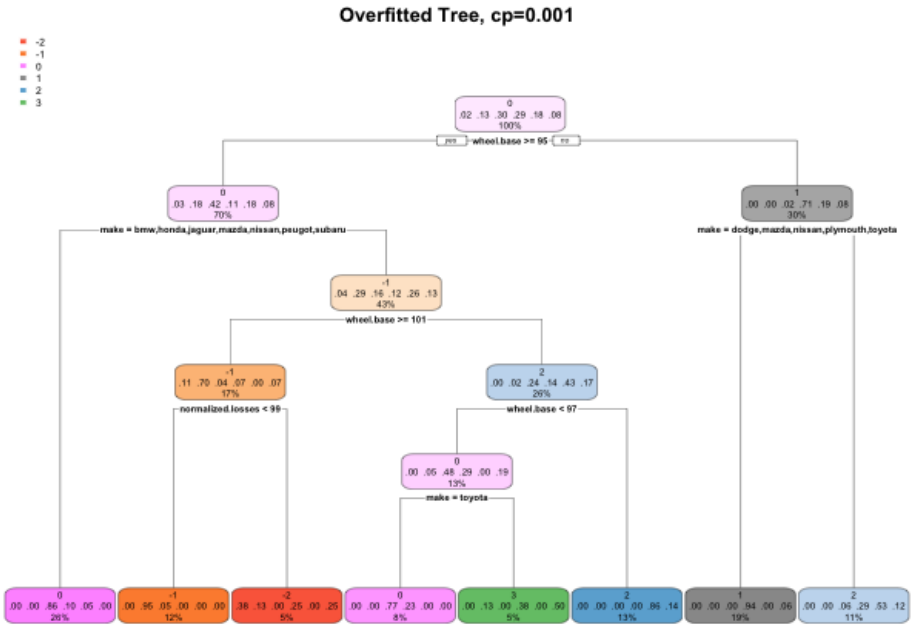
Appendix 10: Distribution of Fuel Type, Aspiration, Engine Location, Engine Type and Number of Cylinders



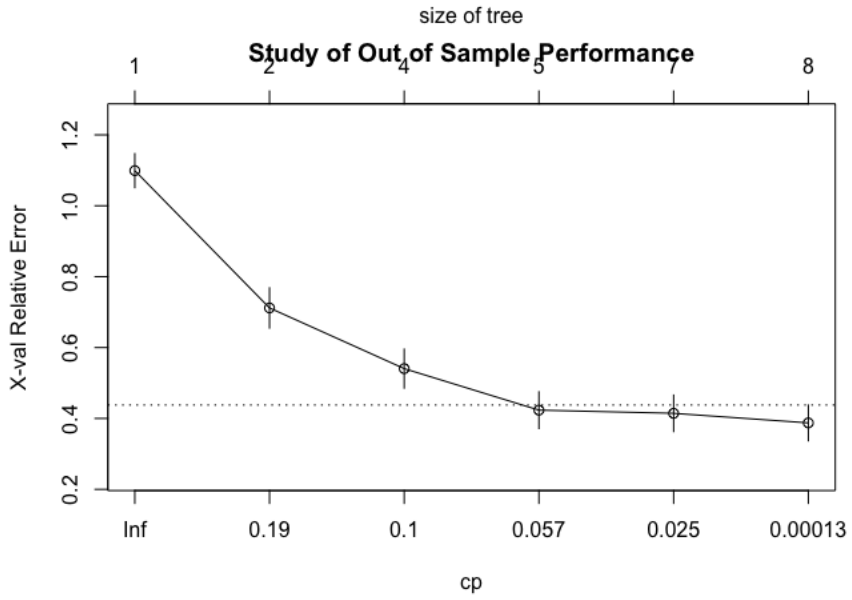
Appendix 11: Importance of Variables in Order



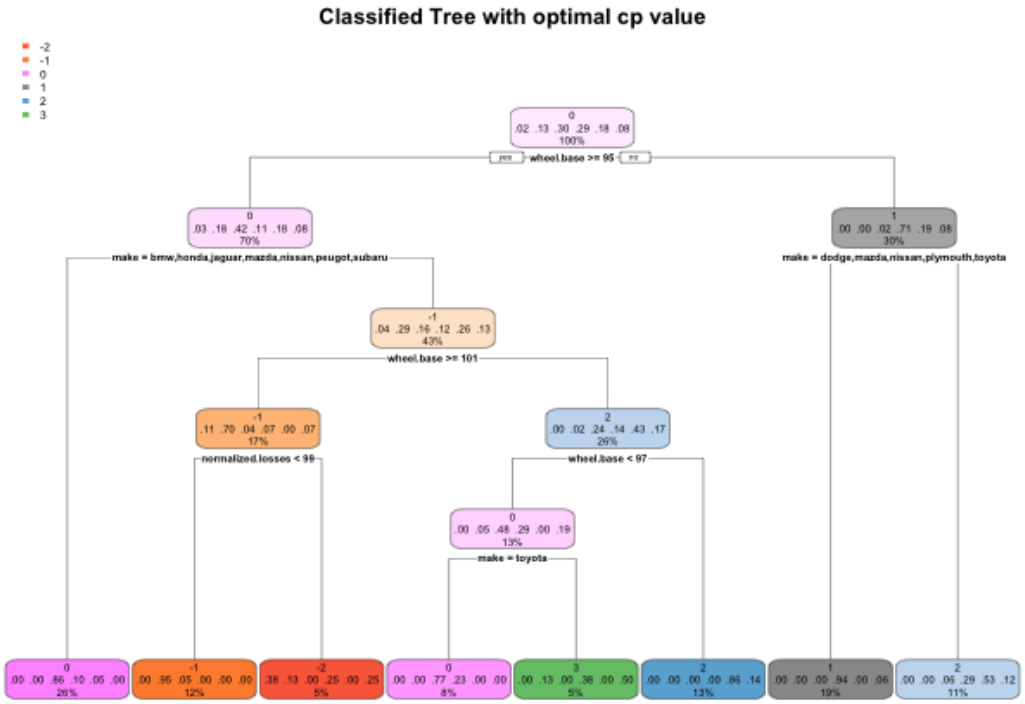
Appendix 12: Overfitted Tree



Appendix 13: Out of Sample performance



Appendix 14: Classified Tree with the optimal cp value



VII. Code

```
#loading the data and converting '?' to NA so it can be notified as a missing value
data <- read.csv("/Users/kaykaydaou/Desktop/MCGILL U3/FALL 22/MGSC 401 - stat founds of data analytics/Final Project/Fall 2022 - MGSC-401-001002 & MGSC-690-071 & RETL-603-095 - 1252022 - 1128 PM/Dataset 5 — Automobile data.csv",  na.strings="?")
data <- na.omit(data)
attach(data)
names(data)
class(data$symboling)
#viewing the variable
table(symboling) #3 cars are risky at a -2 level of insurance risk rating (considered more safe)
#               22 are risky at level -1
#               67 are risky at level 0
#               54 are risky at level 1
#               32 are risky at level 2
#               27 are risky at level 3

#### numerical variables converted for consistency
data$curb.weight <- as.numeric(data$curb.weight)
data$engine.size <- as.numeric(data$engine.size)
data$bore <- as.numeric(data$bore)
data$stroke <- as.numeric(data$stroke)
data$horsepower <- as.numeric(data$horsepower)
data$peak.rpm <- as.numeric(data$peak.rpm)
data$city.mpg <- as.numeric(data$city.mpg)
data$highway.mpg <- as.numeric(data$highway.mpg)
data$price <- as.numeric(data$price)
data$normalized.losses <- as.numeric(data$normalized.losses)
attach(data)

#### categorical variables
data$symboling <- as.factor(data$symboling)
data$make <-as.factor(data$make)
data$fuel.type <-as.factor(data$fuel.type)
data$aspiration <-as.factor(data$aspiration)

data$num.of.doors <-as.factor(data$num.of.doors)
#noticed it has 3 levels, one of them being '?' = missing values
#dropping the missing values in num.of.doors:
data$num.of.doors <- gsub("?", "", data$num.of.doors)
attach(data)
table(num.of.doors) #? is no longer in the data
data$num.of.doors <-as.factor(data$num.of.doors)

data$body.style <-as.factor(data$body.style)
```

```
data$drive.wheels <-as.factor(data$drive.wheels)
data$engine.location <-as.factor(data$engine.location)
data$engine.type <-as.factor(data$engine.type)
data$num.of.cylinders <-as.factor(data$num.of.cylinders)
data$fuel.system <-as.factor(data$fuel.system)
attach(data)

#### performing PCA to visualize the important features
#dividing data into labels and variables
data_labels=data[,c('make','fuel.type','aspiration','num.of.doors','body.style','drive.wheels',
                    'engine.location','engine.type','num.of.cylinders','fuel.system','symboling')]
data_vars=data[,c('wheel.base','length','width','height','curb.weight','engine.size','bore',
                  'stroke','compression.ratio','horsepower','peak.rpm','city.mpg',
                  'highway.mpg','price','normalized.losses')]

#visualizing the numerical data with PCA
pca=prcomp(na.omit(data_vars),scale=TRUE)
pca
#plotting the PCA using first two components
install.packages("ggfortify")
library(ggfortify)
plot=autoplot(pca, data = na.omit(data_vars), loadings = TRUE, col="blue", loadings.label =
TRUE )
plot+ggtitle("PCA of the first two components")
#from PC1, curb weight and engine size but also length and width are the most significant +
wheel base & price
#from PC2, compression ratio, height => engine power
#from PC2, normalized losses and peak rpm are highly correlated => acceleration and top speed

table(symboling)
#pca when symboling = -2
autoplot(pca, data = na.omit(data_vars), loadings = TRUE, loadings.label =
TRUE,loadings.label.size = 3,
         col=ifelse(data$symboling =="-2","blue","transparent"),loadings.colour = 'steelblue',
         loadings.label.colour="black", main= "PCA with Symboling = -2" )

#pca when symboling = -1
autoplot(pca, data = na.omit(data_vars), loadings = TRUE, loadings.label =
TRUE,loadings.label.size = 3,
         col=ifelse(data$symboling =="-1","red","transparent"),loadings.colour = 'red',
        loadings.label.colour="black", main= "PCA with Symboling = -1" )

#pca when symboling = 0
autoplot(pca, data = na.omit(data_vars), loadings = TRUE, loadings.label =
TRUE,loadings.label.size = 3,
         col=ifelse(data$symboling =="0","pink","transparent"),loadings.colour = 'pink',
```

```
        loadings.label.colour="black", main= "PCA with Symboling = 0" )

#pca when symboling = 1
autoplot(pca, data = na.omit(data_vars), loadings = TRUE, loadings.label =
TRUE,loadings.label.size = 3,
        col=ifelse(data$symboling =="1","orange","transparent"),loadings.colour = 'orange',
         loadings.label.colour="black", main= "PCA with Symboling = 1" )

#pca when symboling = 2
autoplot(pca, data = na.omit(data_vars), loadings = TRUE, loadings.label =
TRUE,loadings.label.size = 3,
        col=ifelse(data$symboling =="2","green","transparent"),loadings.colour = 'green',
         loadings.label.colour="black", main= "PCA with Symboling = 2" )

#pca when symboling = 3
autoplot(pca, data = na.omit(data_vars), loadings = TRUE, loadings.label =
TRUE,loadings.label.size = 3,
        col=ifelse(data$symboling =="3","purple","transparent"),loadings.colour = 'purple',
         loadings.label.colour="black", main= "PCA with Symboling = 3" )

#determining the optimal number of components - minimizes variance - could be useful for
clustering
pve=(pca$sdev^2)/sum(pca$sdev^2)
par(mfrow=c(1,2))
plot(pve, ylim=c(0,1))
plot(cumsum(pve), ylim=c(0,1))
# with 4 components, accuracy is 80%: optimal # of components is 4

#visualize our numerical variables that are deemed significant given PCA - took the relevant
ones in the report for analysis
#curb weight, engine size, length, width, wheel base, price
#compression ratio, highway mpg, city mpg & height
plot(curb.weight,symboling, xlab="Curb Weight", ylab="Symboling", main="Relationship of
Curb Weight with Symbol",col.main='red',col.lab='blue',col.lab='blue')
plot(engine.size,symboling, main="Relationship of Engine Size with
Symbol",col.main='red',col.lab='blue',col.lab='blue')
plot(length,symboling, main="Relationship of Engine Size with
Symbol",col.main='red',col.lab='blue',col.lab='blue')
plot(width,symboling, main="Relationship of Width with
Symbol",col.main='red',col.lab='blue',col.lab='blue')
plot(wheel.base,symboling, xlab="Wheel Base", ylab="Symboling",main="Relationship of
Wheel Base with Symbol",col.main='red',col.lab='blue',col.lab='blue')
plot(price,symboling, main="Relationship of Price with
Symbol",col.main='red',col.lab='blue',col.lab='blue')
plot(compression.ratio,symboling, main="Relationship of Compression Ratio with
Symbol",col.main='red',col.lab='blue',col.lab='blue')
```

```
plot(height,symboling, main="Relationship of Height with
Symbol",col.main='red',col.lab='blue',col.lab='blue')
plot(normalized.losses,symboling, xlab="Normalized Losses", ylab="Symboling",
main="Relationship of Normalized Losses with
Symbol",col.main='red',col.lab='blue',col.lab='blue')
plot(peak.rpm,symboling, main="Relationship of Peak Rpm with
Symbol",col.main='red',col.lab='blue',col.lab='blue')

###visualize categorical data
table(data$symboling)
#dist of symboling
q <- ggplot(data=na.omit(data),aes(x=symboling))+
geom_bar(fill="red",color="red")+ggtitle("Distribution of Insurance Risk Rating")
q + xlab("Symboling")

table(data$make)
#distribution of make
d <- ggplot(data=na.omit(data),aes(x=make))+
geom_bar(fill="gold",color="gold")+ggtitle("Distribution of car manufacturers")
d + xlab("Car Manufacturers")

table(data$fuel.type) #only 20 are diesel compared to 185 gas => not too significant
#distribution of fuel type
w <- ggplot(data=na.omit(data),aes(x=fuel.type))+
geom_bar(fill="blue",color="blue")+ggtitle("Distribution of Fuel Type")
w + xlab("Fuel Type")

table(data$aspiration) #only 37 turbo compared to 168 gas => not that significant
#distribution of aspiration
t <- ggplot(data=na.omit(data),aes(x=aspiration))+
geom_bar(fill="blue",color="blue")+ggtitle("Distribution of Aspiration")
t + xlab("Aspiration")

table(data$num.of.doors)
#distribution of num.of.doors
p <- ggplot(data=na.omit(data),aes(x=num.of.doors))+
geom_bar(fill="green",color="green")+ggtitle("Distribution of Number of Doors")
p + xlab("Number of Doors")

table(data$body.style)
#dist of body style
o<- ggplot(data=na.omit(data),aes(x=body.style))+
geom_bar(fill="purple",color="purple")+ggtitle("Distribution of Body Style")
o + xlab("Body Style")

table(data$drive.wheels)
```

```
#dist of drive wheels
a <- ggplot(data=na.omit(data),aes(x=drive.wheels))+
geom_bar(fill="lightblue",color="lightblue")+ggtitle("Distribution of Drive Wheels")
a + xlab("Drive Wheels")

table(data$engine.location) #only 3 in rear => not too significant compared to 202 in front and
isn't shown in the plot
#dist of engine location
s <- ggplot(data=na.omit(data),aes(x=engine.location))+
geom_bar(fill="blue",color="blue")+ggtitle("Distribution of Engine Location")
s + xlab("Engine Location")

table(data$engine.type) #mainly ohc => not too significant
#dist of engine type
u <- ggplot(data=na.omit(data),aes(x=engine.type))+
geom_bar(fill="blue",color="blue")+ggtitle("Distribution of Engine Type")
u + xlab("Engine Type")

table(data$num.of.cylinders) #=> mainly four cylinders, not too significant
#dist of num.of.cylinders
f <- ggplot(data=na.omit(data),aes(x=num.of.cylinders))+
geom_bar(fill="blue",color="blue")+ggtitle("Distribution of Number of Cylinders")
f + xlab("Number of Cylinders")

table(data$fuel.system)
#dist of fuel.system
m <- ggplot(data=na.omit(data),aes(x=fuel.system))+
geom_bar(fill="darkgreen",color="darkgreen")+ggtitle("Distribution of Fuel System")
m + xlab("Fuel System")

###Random forest
install.packages("randomForest")
library(randomForest)
myforest=randomForest(symboling~normalized.losses+make+fuel.type+aspiration+
                      num.of.doors+body.style+drive.wheels+engine.location+
                      wheel.base+length+width+height+curb.weight+engine.type+
                      num.of.cylinders+engine.size+fuel.system+bore+stroke+
                      compression.ratio+horsepower+peak.rpm+city.mpg+highway.mpg+
                      price,ntree=500,data=data,importance=TRUE,na.action=na.omit)

myforest #i get an OOB estimate of error rate of 10.69%% which is good - avoids overfitting

##visualize relative predictive performance of my variables
importance(myforest) # most important variables in order are make, normalized losses, wheel
base, width
```

```
#also engine location seems completely useless, if we remove it, MSE / mean decrease accuracy
wont change

varImpPlot(myforest, main= "Importance of Variables in Order") #visualize importance in order
#this provides a good guideline of which predictors are important: make, normalized losses,
wheel base, width

#testing oob performance
myforest=randomForest(symboling~normalized.losses+make+fuel.type+aspiration+
                      num.of.doors+body.style+drive.wheels+engine.location+
                      wheel.base+length+width+height+curb.weight+engine.type+
                      num.of.cylinders+engine.size+fuel.system+bore+stroke+
                      compression.ratio+horsepower+peak.rpm+city.mpg+highway.mpg+
                      price,ntree=500,data=data,importance=TRUE,na.action=na.omit, do.trace=50)

#trying random forest with most important variables
classifiedforest=randomForest(symboling~make+normalized.losses+
                              wheel.base+width+height, cp=0.01, na.action=na.omit)
summary(classifiedforest)
classifiedforest #OOB error rate becomes 8.81%  which is good - better than first random forest

#predicting the symbol of a car  using important variables - classification tree
table(symboling)
#performing cross validation to find the best-tree
myoverfittedtree=rpart(symboling~make+normalized.losses+
                       wheel.base+width+height,control=rpart.control(cp=0.001))
rpart.plot(myoverfittedtree, main="Overfitted Tree, cp=0.001")
#studying out of sample performance
printcp(myoverfittedtree)
plotcp(myoverfittedtree, main="Study of Out of Sample Performance")
#finding cp value that minimizes the error
opt_cp=myoverfittedtree$cptable[which.min(myoverfittedtree$cptable[,"xerror"]),"CP"]
opt_cp
#best tree is one that has a cp of 1e-06

#trying with cp = 1e-06
classifiedtree2=rpart(symboling~make+normalized.losses+
                      wheel.base+width+height, cp=opt_cp, na.action=na.omit)
rpart.plot(classifiedtree2, main="Classified Tree with optimal cp value")

##idea! predict a car's symbol - change values at random
table(make)
#let's predict chevrolet's symbol with some random variables
predict(classifiedtree2,data.frame(make='chevrolet',normalized.losses=89,wheel.base=90,
                        length=125,width=58,height=51))
```