# Introduction

The data from the automobile industry is to be examined for the purposes of determining if there is a relation present between the “popularity” of a car and its MSRP. We were given the task to determine and predict the retail price of a vehicle using various attributes. Before proceeding it is very important to note that this is an observational study and the conclusions are limited in scope to the data from our analysis.

# Data Description

The dataset comes to us as an extension of homework two and includes 16 various factors related to cars. This is a common dataset and contains many useful parameters for analysis.

# Exploratory Data Analysis

The data was thoroughly examined before analysis, and a few observations on the data can be taken. The data was first combed for missing and incorrect data. The missing data consisted of 99 entries, and most of these entries were due to electric cars not being measured on the same metrics as traditional gasoline vehicles. As you can see in the plot below <0.1% of data from HP and horsepower was missing.

Text

Description automatically generatedThe columns with missing data were, “Number of Cylinders” and, “HP.” The electric cars cannot be measured on these metrics, so for all electric cars we inserted a 0 value, and later on in the analysis this will be considered to be factors so analysis, thus resulting in no adverse effects. This was done for both columns. Additionally, we noticed a few vehicles with missing HP data, namely Mazda, Ford, and Lincoln. Luckily all the general information on these cars is available online and the end result was the data being updated by hand since there was such a small number of missing data points. Since there was such a small number of missing records, we did not notice a significant shift in the summary statistics after performing this modification <1% difference. As a note all horsepower data was pulled from Motor trend for consistency.

Text

Description automatically generatedAs you can see in the missingness plot, after the data was fixed we achieved a 100% present data rate.

After data cleaning and verification was complete. We began to look at the different distributions of variables and how they could potentially play into price. After data cleaning, we examined the distribution of our outcome variable MSRP. There is quite a range of values on MSRP, with some cars being around $2000 and some going all the way to over $200000. This seems like it could be a problem, and sure enough after plotting, non-normality seemed to present itself.Chart, histogram

Description automatically generatedChart, histogram

Description automatically generated

As seen in the plots above, MSRP greatly benefits from a log transformation. This will however change the inferences we make on the outcome. The log transform seems to sort out many of the issues with this data, since this is somewhat like income data it tends to be very right skewed.

With many variables, and many levels of each factor, correlation charts were created. In the appendix the primary measure of correlation is viewable (Exhibit A.1). When choosing between variables, we looked at the most highly correlated variables to MSRP. Some very interesting trends presented themselves, these included Horse Power, vehicle style, and engine cylinders. Exhibit B features the plots of the relationships between these variables and our outcome variable of interest. The various variables plotted in appendix b, show a few trends between MSRP and other predictor variables from the dataset.

Within exhibit B we can see evidence of a relationship between a few of the predictors. There is strong evidence of a linear relationship between HP and MSRP. Engine cylinders also shows evidence of a linear relationship as well.

As a final point on EDA, all of it was performed on the training dataset which comprised of about 80% of the data. An additional point is that the data is further split between testing and validation of 10% each. It should also be stated that no points were removed from the data after viewing Cook’s D and checking for outliers. The data is fairly uniform.

# Objective 1: Regression Analysis and Key Relationships

We have been required to identify relationships between and interpreting those relationships. Some key ideas to address are the importance of the popularity variable and how much of a role it plays in MSRP.

First and foremost we must address assumptions for multiple linear regression. We need to first check to see if the residuals are normally distributed, then we need to check for constant variance, followed by homoscedasticity, finally we must assume homoscedasticity. We can confirm that these assumptions are met in the appendix in section C. We will assume the data are independent of each other since we have no further information.

# Variable Selection

We decided on which variables in place into the model based on correlation matrices and various plots of the relationships. The correlation plot is available in exhibit A and shows that there is strong correlation between engine horsepower and engine cylinders along with a few other numerical variables. Variables which had high multicollinearity were not included in the model.

# Competing Models

Model A

(Simple)

MSRP =

Model B

(Complex)

MSRP =

Review of model assumptions for Simple Model. The residual plot for one iteration appeared to meet the model assumption of no clear evidence of violation. The data doesn’t have a clear pattern and roughly has the residuals evenly distributed between both sides of the line. The quantile-quantile plot looks very, very roughly passable, but the tail ends deviate away from normality. There is good evidence of outliers and high leverage points, however there were 100 iterations of the model done, so we cannot speak to specifics due to the changing nature of the plots.

The same is true of the complex model. The residual plot for one iteration appeared to meet the model assumption of no clear evidence of violation. The data doesn’t have a clear pattern and roughly has the residuals evenly distributed between both sides of the line. The quantile-quantile plot looks very normal and fits the assumption quite nicely. There is no evidence of outliers and high leverage points, however there were 100 iterations of the model done, so we cannot speak to specifics due to the changing nature of the plots.

In general the complex model seems to have a better fit in almost all metrics. The simple model did not perform nearly as well as the complete model.

# Coefficient Interpretation

The simple model was created based on human intuition mainly by examining the data and visualizing the data and searching for relationships and trends. Variable selection was decided on by different metrics mainly BIC and Rsqr. In exhibit F, we can see that Rsqr and BIC were used to select number of variables. Based on repeated sampling, we felt our model was optimal with around 15 predictors.

We have only one categorical variable which has 16 different levels. The variable was Vehicle.Style with levels: 2dr Hatchback, 2dr SUV,4dr Hatchback,4dr SUV, Cargo Minivan, Cargo Van, Convertible SUV, Coupe,Crew Cab pickup, Extended Cab Pickup, Passenger Minivan, Passenger Van, Regular Cab Pickup

In the end the simpler model was chosen, this was due to similar performance despite being much simpler.

Despite the RMSE of the full model being much lower, the complexity of the model did not make up for the worse fit of the simple model. This is an excellent statistical case of a tradeoff between a good model and an overly complicated one.

MSRP =

2 dr hatchback is the reference category.

Interpretation:

MSRP = 8.7964645+0.0094268\*Engine.HP-0.2215210\*Engine.Cylinders

Interpretation for 2dr SUV

MSRP = (8.7964645-0.6808094)+0.0094268\*Engine.HP-0.2215210\*Engine.Cylinders

Interpretation for 4dr HatchBack

MSRP = (8.7963645+0.4460832)+0.0094268\*Engine.HP-0.2215210\*Engine.Cylinders

Interpretation for Cargo Minivan

MSRP = (8.7963645+.5003414)+0.0094268\*Engine.HP-0.2215210\*Engine.Cylinders

The rest of the interpretations can be generalized to this approach and can be calculated using regression coefficients. The rest of the coefficients can be found in exhibit G.

, are the intercept, engine horse power and the engine cylinders.

If the cost of the car is 0, this is not in the scope of the observation.

interpretation, the estimated horse power associated with 2 dr hatchback has a point estimate of 0.0094268 and a 95% CI of (0.009214652,0.009638934). Additionally, we will say that a one-unit increase in engine horse power, holding all else equal would result in an expected increase in MSRP of 0.0094268 increase.

interpretation, the estimated engine cylinders associated with a 2 dr hatchback has a point estimate of -0.2215210 and a 95% CI of (-0.234483849,-0.208558120). Additionally, we will say that a one unit increase in engine horse power, holding all else equal would result in an expected decrease in MSRP of -0.2215210.

Popularity is a very difficult variable to pin down. There are many loose associations, but there are many variables tied to popularity. Certain vehicles seem to be tied to different levels of popularity with Ford trucks reaching the top of the popularity scale.

Chart, bar chart

Description automatically generated

# Objective Two:

Restatement of problem and overall objective:

We will use the training and test set to build another multiple linear regression model with increased complexity. We will accomplish this using K-nearest neighbor’s regression. The training dataset that was created will be used to determine the optimal number for “K.”

The data from the automobile industry is to be examined for the purposes of determining if there is a relation present between the “popularity” of a car and its MSRP. We were given the task to determine and predict the retail price of a vehicle using various attributes. Before proceeding it is very important to note that this is an observational study and the conclusions are limited in scope to the data from our analysis.

KNN is a useful technique, however KNN required numeric values. This model used Engine.HP, Engine.Cylinders,Highway.MPG, CityMPG,Popularity. In addition, we used a random forest to see if we could improve the fit of the KNN. One additional issue with this approach is that KNN is quite difficult to interpret, the model also suffers from high variance and low bias.

In general KNN is a non-parametric classification and can be implemented for regression. In the regression context, the output is the property value for the object. And this value is the average of the values of k nearest neighbors. The KNN algorithm relies on distance, so normalizing is very important.

|  |  |  |  |
| --- | --- | --- | --- |
| Model Type | Regression (Simple) | KNN | Random Forest |
| Average RMSE | 0.738 | 0.8076 | 0.7231 |

KNN has an optimal K of 5. Random forest also has an optimal value of 5. Appendix A, section I has references to the output from the random forest and the KNN approach. As well as the plots.

# Final Summary:

Objective 1 and objective two approached the same problem with vastly different approaches. Objective one used different types of regression to determine important relationships by way of forward selection, backward selection, and manual selection of variables. The model assumptions were checked and evaluated, the residual plots, and the rest of the assumptions were met and validated. The models were run multiple times to tune the parameters and the results were outputted into both the document and the appendix. In the end, objective 1 was completed with two different models created. There was a simple model, and a complex model. The complex model, at least on paper, fit the data better, but there were large concerns of overfitting in this case. Because of this concern a simpler model was selected and used in the end. The model highlights the importance of the car type, engine hp and engine cylinders in terms of MSRP and popularity. Of course, as popularity was very vague and did not leave many details we can only assume thins about it. In general, we can use the aforementioned variables to better predict the MSRP of cars and their popularity. This was an observational study, and many things could be happening behind the scenes, and this further study is needed to determine any further relationships.

Based on the outcome, and depending on the goals of someone who is using this data, I feel that using the simpler regression model is better for prediction and ease of interpretation. However, if the lowest RMSE is your goal, use the random forest on top of the KNN sequence. All in all, it would be good to have a bit more insight on the popularity factor, as this could be a very powerful variable to understand.

# Appendix A:

Data Columns Listed in the Data File:

|  |  |  |
| --- | --- | --- |
| **Variable Name** | **Data Type** | **Description** |
| MSRP | Numeric | The response variable |
| Car Make | Factor | The company that made the car. Ex: Honda, Toyota, etc. |
| Car Model | Factor | The model of the car. Ex: 4Runner, Accord, etc. |
| Year | Numeric | Year the car was produced |
| Engine Fuel Type | Factor | Type of fuel the car accepts. Ex: Regular unleaded, Premium unleaded, Diesel |
| Engine HP | Numeric | Horsepower of the car’s engine. |
| Engine Cylinders | Numeric | Number of cylinders in the car’s engine. |
| Transmission Type | Factor | Type of transmission in the car. Usually manual or automatic, but there are a few specialty transmission types in the data. |
| Driven\_Wheels | Numeric | The wheels that are powered by the engine. Ex: Front Wheel, Rear Wheel, Four Wheel Drive |
| Number of Doors | Numeric | The number of doors that the car has. Usually 2 or 4 |
| Market Category | Factor | Various special factors for each car. Ex: Exotic, Luxury, High-Performance, Flex Fuel. Note: we created a new feature using Exotic/Not Exotic for our analysis |
| Vehicle Size | Factor | The size of the vehicle. Ex: Midsize, Large, Compact |
| Vehicle Style | Factor | Body type of the vehicle. Ex: Coupe, Convertible, etc. |
| Highway MPG | Numeric | Fuel efficiency on the highway in MPG |
| City MPG | Numeric | Fuel efficiency in the city in MPG |
| Popularity | Numeric | A popularity score for each car. The dataset does not detail how the popularity score is calculated. |

Exhibit A:

Correlation between numerical variables

(A.1)

Chart, bubble chart

Description automatically generated

Exhibit B:

(B.1) Engine horsepower vs MSRP

Chart, scatter chart

Description automatically generated

Popularity vs MSRP

(B.2)

Chart

Description automatically generated

Vehicle Style vs MSRP

(B.3)

Chart, box and whisker chart

Description automatically generated

Engine Cylinders vs MSRP

(B.4)

Chart, line chart

Description automatically generated

Exhibit C:

Assumption plots for simple model

Chart, scatter chart

Description automatically generated

Chart, scatter chart

Description automatically generated

Chart, line chart

Description automatically generated

Chart, histogram

Description automatically generated

Chart, scatter chart

Description automatically generated

Assumption plots for complex model

Chart, scatter chart

Description automatically generated

Chart

Description automatically generated

Chart, scatter chart

Description automatically generated

Chart, line chart

Description automatically generated

Chart, histogram

Description automatically generated

Exhibit D:

GG Pairs Plot

Diagram

Description automatically generated with low confidence

Exhibit E:

Chart, scatter chart

Description automatically generated

Chart, bar chart

Description automatically generated

Exhibit F:

Chart, line chart

Description automatically generated

Chart, line chart

Description automatically generated

Exhibit G:

Table

Description automatically generated with medium confidence

G.2

Text, table

Description automatically generated with medium confidence

Exhibit H:

A picture containing text, receipt

Description automatically generated

Exhibit I:

Chart, line chart

Description automatically generated

Chart, line chart, box and whisker chart

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated

# Appendix B: Code Bank, R was used for all analysis

---

title: "DS6372 Project 1"

output: html\_notebook

editor\_options:

chunk\_output\_type: inline

---

## Project 1, MSDS6372

# By Group 2: Helene, Ben, Will

```{r}

#Project Libraries

library(naniar)

library(caret)

library(ggplot2)

library(mlbench)

library(glmnet)

library(olsrr)

library(fmsb)

library(corrplot)

library(RColorBrewer)

library(funModeling)

library(tidyverse)

library(Hmisc)

library(kableExtra)

```

## Objective 1 Overview

Objective 1: Display the ability to build regression models using the skills and discussions from Unit 1 and 2 with the purpose of identifying key relationships and interpreting those relationships. A key question of interest that must be addressed in this analysis is the importance of the “Popularity” variable. While the details of this variable are vague, it was created from social media, and the “higher ups” are curious how much general popularity can play a role in the retail price of a vehicle.

# Objective 1, Prelimenary Data Assessment and data parsing

```{r}

#Read in the data file

proj1Dat <- read.csv("data1.csv")

vis\_miss(proj1Dat)

#Make df to work with, so orginal data is untouched

carDat <- proj1Dat

carDat$MSRP <- log(carDat$MSRP)

#Examine data

#View(proj1Dat)

#See problems with missing data

new\_DF <- proj1Dat[rowSums(is.na(proj1Dat)) > 0,]

#View(new\_DF)

#Looks like data is pretty good, need to handle NA's for electric cars (cylinders)

#Proposed solution is to change electric cars to 0 cylinders and convert to factor

#Also need to handle NA's for electric cars (horsepower)

#Proposed solution is to change horsepower to 0 and convert to factor

#Some mazda rx7 missing values

#Some mazda rx8 missing values

#Change NA's

carDat[is.na(carDat)] <- 0

#Now fix 8696-8698 with 2 cylinders

carDat[c(8696:8698),6] <- 2

#And fix 8699-8715 with 4 cylinders

carDat[c(8699:8715),6] <- 4

#And fix 4204-4207 with 168

carDat[c(4204:4207),5] <- 168

#And fix 4915-4920 with 193

carDat[c(4915:4920),5] <- 193

#And fix 5826,5831,5832,5834,5840,5841 with 301

carDat[c(5826,5831,5832,5840,5841),5] <- 301

#And fix 6909, 6911, 6917,6919 with 305

carDat[c(6909,6911,6917,6919),5] <- 305

#Convert to factors

carDat$Make <- as.factor(carDat$Make)

carDat$Year <- as.factor(carDat$Year)

carDat$Engine.Fuel.Type <- as.factor(carDat$Engine.Fuel.Type)

carDat$Transmission.Type <- as.factor(carDat$Transmission.Type)

carDat$Number.of.Doors <- as.factor(carDat$Number.of.Doors)

carDat$Market.Category <- as.factor(carDat$Market.Category)

carDat$Vehicle.Size <- as.factor(carDat$Vehicle.Size)

carDat$Vehicle.Style <- as.factor(carDat$Vehicle.Style)

#Look at missing data

vis\_miss(carDat)

#Looks good

```

# Split up data into 80% training, 10% testing, and 10% validation

```{r}

#time to split up the data .8 train, .1 test, .1 validate

ss <- sample(1:3,size=nrow(carDat),replace=TRUE,prob=c(0.8,0.1,0.1))

train <- carDat[ss==1,]

test <- carDat[ss==2,]

cvr <- carDat[ss==3,]

```

## Objective 1.1

Build a model with the main goal to identify key relationships and is highly interpretable. Provide detailed information on summary statistics, EDA, and your model building process.

```{r}

#Summary Statistics

mean(carDat$Popularity)

min(carDat$Popularity)

max(carDat$Popularity)

sd(carDat$Popularity)

#General summary stats

summary(carDat)

summary(proj1Dat)

#Summary of vehicle style

summary(carDat$Vehicle.Style)

plot(carDat$Vehicle.Style)

p <- ggplot(carDat, aes(fill=Popularity, y=Vehicle.Style, x=MSRP)) +

geom\_bar(position="dodge", stat="identity")

p + theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))

#EDA

ggplot(carDat, aes(x = MSRP, y = Popularity,color=Vehicle.Style))

geom\_point()

ggplot(data = carDat, mapping = aes(x = MSRP, y = Popularity)) +

geom\_point() +

geom\_smooth(aes(color = Vehicle.Style)) +

facet\_wrap( ~Vehicle.Style)

#Histogram of MSRP

ggplot(carDat, aes(x=(MSRP))) +

geom\_histogram(aes(y=..density..), colour="black", fill="white")+

geom\_density(alpha=.2, fill="#FF6666")

#Log transformed history of MSRP

ggplot(carDat, aes(x=log(MSRP))) +

geom\_histogram(aes(y=..density..), colour="black", fill="white")+

geom\_density(alpha=.2, fill="#FF6666")

#Histogram of popularity

ggplot(carDat, aes(x=(Popularity))) +

geom\_histogram(aes(y=..density..), colour="black", fill="white")+

geom\_density(alpha=.2, fill="#FF6666")

M <-cor(carDat[,c(5,6,13,14,15,16)])

corrplot(M, type="upper", order="hclust",

col=brewer.pal(n=8, name="RdYlBu"))

plotmatrix((carDat[,c(5,6,13,14,15,16)]))

ggpairs(carDat[,c(5,6,13,14,15,16)])

plot\_num(carDat)

ggplot(carDat, aes(x=Engine.HP, y=MSRP)) +

geom\_point()+

geom\_smooth(method = "lm")

ggplot(carDat, aes(x=Popularity, y=MSRP)) +

geom\_point()+

geom\_smooth()

ggplot(carDat, aes(x=Vehicle.Style, y=MSRP)) +

geom\_point()+

geom\_smooth() + theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))

ggplot(carDat, aes(x=Engine.Cylinders, y=MSRP)) +

geom\_point()+

geom\_smooth() + theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))

```

```{r}

#Model Selection

reg.fwd=regsubsets(MSRP~.,data=train,method="forward",nvmax=15)

bics<-summary(reg.fwd)$bic

plot(bics,type="l",ylab="BIC",xlab="# of predictors")

reg.fwd$nbest

regfwdMdl <- lm(MSRP~.,data=train)

summary(regfwdMdl)

reg.bwd=regsubsets(MSRP~.,data=train,method="backward",nvmax=15)

bics<-summary(reg.bwd)$bic

# Adjr2

adjr2<-summary(reg.fwd)$adjr2

plot(adjr2,type="l",ylab="Adjusted R-squared",xlab="# of predictors")

MallowCP <- summary(reg.fwd)$cp

plot(MallowCP,type="l",ylab="Mallow's CP",xlab="# of predictors")

##Human model:

edaModel <- lm(MSRP~Engine.HP+Vehicle.Style+Engine.Cylinders,data=carDat)

summary(edaModel)

ols\_plot\_resid\_fit(edaModel)

ols\_plot\_resid\_lev(edaModel)

ols\_plot\_resid\_qq(edaModel)

ols\_plot\_resid\_hist(edaModel)

ols\_plot\_cooksd\_bar(edaModel)

p <- predict(edaModel, cvr)

error <- (p- cvr$MSRP)

RMSE\_Model <- sqrt(mean(error^2))

ptest <- predict(edaModel, test)

error1 <- (ptest- cvr$MSRP)

RMSE\_NewData <- sqrt(mean(error1^2))

Method <- c("Train/Test Split")

ModelRMSE <- c(RMSE\_Model)

RMSENewData <- c(RMSE\_NewData)

table1 <- data.frame(Method, ModelRMSE, RMSENewData)

kable(table1) %>% kable\_styling(c("striped", "bordered")) %>%column\_spec(2:3, border\_left = T)

#Full model

fullModel <- lm(MSRP~.,data = carDat)

summary(fullModel)

ols\_plot\_resid\_fit(fullModel)

ols\_plot\_resid\_lev(fullModel)

ols\_plot\_resid\_qq(fullModel)

ols\_plot\_resid\_hist(fullModel)

ols\_plot\_cooksd\_bar(fullModel)

p <- predict(fullModel, test)

error <- (p- test$MSRP)

RMSE\_Model <- sqrt(mean(error^2))

ptest <- predict(fullModel, test)

error1 <- (ptest- carDat$MSRP)

RMSE\_NewData <- sqrt(mean(error1^2))

Method <- c("Train/Test Split")

ModelRMSE <- c(RMSE\_Model)

RMSENewData <- c(RMSE\_NewData)

table1 <- data.frame(Method, ModelRMSE, RMSENewData)

kable(table1) %>% kable\_styling(c("striped", "bordered")) %>%column\_spec(2:3, border\_left = T)

# Set number of times you would like to repeat the sampling/testing

iterations = 1:100

# the initial values for the columns (might not need these now that ive switched to building columns)

rmseSimple = c()

rmseComplex = c()

# Start of Loop

for(i in iterations){

# Resets sample every iteration

index<- sample(1:dim(carDat)[1],128,replace=F)

train<- carDat[index,]

test<- carDat[-index,]

# the model runs

edaModel

fullModel

# predictors and column building

predictions1 <- edaModel %>% predict(test)

d1 = data.frame(R2 = R2(predictions1,test$MSRP),

RMSE = RMSE(predictions1,test$MSRP), MAE = MAE(predictions1, test$MSRP))

rmseSimple = c(rmseSimple,d1$RMSE)

predictions2 <- fullModel %>% predict(test)

d2 = data.frame(R2 = R2(predictions2,test$MSRP),

RMSE = RMSE(predictions2,test$MSRP), MAE = MAE(predictions2, test$MSRP))

rmseComplex = c(rmseComplex, d2$RMSE)

# End for

}

# putting the dataframe together and outputting relevant statistics

Model.Average.RMSE = cbind(rmseSimple, rmseComplex)

rmsedf = as.data.frame(Model.Average.RMSE)

Means = colMeans(Model.Average.RMSE)

SDs = round(colSds(Model.Average.RMSE), 3)

range1 = max(rmsedf$rmseSimple) - min(rmsedf$rmseSimple)

range2 = max(rmsedf$rmseComplex) - min(rmsedf$rmseComplex)

rmsedf1 = melt(rmsedf,rmse = c("n", "rmse"))

summary(Model.Average.RMSE)

Pred1 <- data.frame(Value = predictions1, Model = "Simple")

Pred2 <- data.frame(Value = predictions2, Model = "Complex")

PredActual <- data.frame(ActualValue = test$MSRP)

PredAll <- rbind(Pred1, Pred2)

PredActual <- rbind(PredActual,PredActual)

PredAll <- cbind(PredAll, PredActual)

PredAll %>% ggplot(aes(x = Value, y = ActualValue, fill = Model)) + geom\_point(aes(color = Model)) + geom\_smooth(formula = y~x)+theme\_minimal()

# Column

rmsedf1 %>% group\_by(variable) %>% summarise(mean = (mean(value))) %>%

ggplot(aes(x = reorder(variable, -mean), y = mean, fill = variable)) + geom\_col(width = 0.75) + geom\_text(aes(label = round(mean,3), vjust = -0.5)) +

ggtitle("Average RMSE over 100 Shuffles (Linear Models)") + xlab("Model #") + ylab("Mean RMSE")+theme\_minimal()

# Boxplot

rmsedf1 %>% ggplot(aes(x = variable, y = value)) + geom\_boxplot(aes(fill = variable)) + facet\_wrap(~variable,ncol = TRUE) +

ggtitle("Mean RMSE Distribution by Model") + ylab("Mean RMSE") + coord\_flip() +

theme(axis.title.y = element\_blank(), axis.text.y = element\_blank(), axis.ticks.y = element\_blank())

# Histogram

rmsedf1 %>% ggplot(aes(x = value)) + geom\_histogram(aes(fill = variable)) + facet\_wrap(~variable,ncol = TRUE) +

ggtitle("Mean RMSE Distribution by Model") + xlab("Mean RMSE") +

theme(axis.title.y = element\_blank(), axis.text.y = element\_blank(), axis.ticks.y = element\_blank())

# Here we can see there is no significant difference between the models in terms of RMSE

t.test(rmseSimple,rmseComplex, var.equal = FALSE)

```

## Objective 1.2

Provide interpretation of the regression coefficients of your final model including hypothesis testing, interpretation of regression coefficients, and confidence intervals. It’s also good to mention the Practical vs Statistical significance of the predictors. Answer any additional questions using your model that you deem are relevant.

## Objective 1.3

The training data set can be used for EDA and model fitting while the test set can be used to help compare models to make a final call. There is no need to use the validation data set for this objective.

# Practical Consideration for Objective 1:

EDA, EDA, EDA! It helps you on so many fronts so use it to your advantage. When writing a concise report, you do not have to literally step out every single step of your model building process. I know you guys are going to being iterating on things many many times. That does not all have to be there. You can summarize that iteration stuff in a paragraph.

What is key in the report is that you develop a “story” of your analysis. Keep in mind that when you are finished with your analysis. You know how it is going to end (what the final models look like). You can use this to your advantage when selecting what parts of the EDA and additional information to show. For example, if you know that predictor X7 is in your final model and it is one of the stronger relationships, that is probably a good one to show and discuss in the EDA part. You would show the reader, “Hey look at these interesting trends”, “Hey look at these that are not”, etc. When you report your final model and you are bringing back up the predictors discussed in EDA, it helps build the confidence of the reader in what you are doing is making sense.

```{r}

#Objective 2

trctrl <- trainControl(method = "repeatedcv", number = 10, repeats = 3)

set.seed(3333)

knn\_fit <- train(MSRP ~., data = train[,c(5,6,13,14,15,16)], method = "knn",

trControl=trctrl,

preProcess = c("center", "scale"),

tuneLength = 10)

(knn\_fit)

test\_pred <- predict(knn\_fit, newdata = test)

test\_pred

plot(knn\_fit)

plot(knn\_fit, print.thres = 0.5, type="S")

set.seed(400)

ctrl <- trainControl(method="repeatedcv",repeats = 3) #,classProbs=TRUE,summaryFunction = twoClassSummary)

# Random forrest

rfFit <- train(MSRP ~ ., data = train[,c(5,6,13,14,15,16)], method = "rf", trControl = ctrl, preProcess = c("center","scale"), tuneLength = 20)

rfFit

plot(rfFit)

```