S&DS 230: Car Price Prediction

5/3/2021

Introduction

Being a car enthusiast has made me curious about what characteristics of a car exactly influence its MSRP. I have come to learn a car's price is heavily dependent on categorical variables such as brand as well as continuous variables such as its miles per gallon and horsepower. In this project, I will be determining what exactly contributes to the differences in prices between cars, and how these differences can help us build a model to predict a car's price. With this goal, I have found a Kaggle data set which has 20+ brands' car models dating back since 2001 with each row containing characteristics and details of the car as well as the current day MSRP for that car.

Data Cleaning

First, I removed all the non-gas based cars because electrics, natural gas, and diesel cars are in a different realm when it comes to prices. Then, I abbreviated the drive system names (four wheel drive = FWD). Then, I cleaned the Market Category variable because there were many category tags for each car. Then, I created a variable named "YearsSince2001" that had values which are the difference between a car's manufacturing year and 2001. Some other minor data cleaning was done to remove incorrect values, and irrelevant variables were dropped. No issues were encountered during cleaning other than making decisions on what market category tags should identified as normal or luxury.

```
#Remove non-gas cars and standardize gas names
df <- subset(df, Engine.Fuel.Type %in% c('flex-fuel (premium unleaded recomme
nded/E85)', 'flex-fuel (premium unleaded required/E85)', 'flex-fuel (unleaded
/E85)', 'premium unleaded (recommended)', 'premium unleaded (required)', 'reg
ular unleaded'))
df$Engine.Fuel.Type <- gsub("flex-fuel.*", "Flex-Fuel", df$Engine.Fuel.Type)
df$Engine.Fuel.Type <- gsub("premium unleaded.*", "Premium Unleaded", df$Engi
ne.Fuel.Type)
df$Engine.Fuel.Type <- gsub("regular unleaded.*", "Regular Unleaded", df$Engi
ne.Fuel.Type)

#Rename Drive System Names
df$Driven_Wheels <- recode(df$Driven_Wheels, 'all wheel drive' = "AWD", 'four
wheel drive' = "4WD", 'front wheel drive' = "FWD", 'rear wheel drive' = "RWD"
)

#Remove unknown transmission types
df <- df [!df$Transmission.Type %in% c("UNKNOWN", "DIRECT_DRIVE"),]

#Multiple Tags for each car's Market.Category are assigned to Luxury or Norma
```

```
l
df$Market.Category <- gsub(".*Exotic.*","Luxury", df$Market.Category)
df$Market.Category <- gsub(".*Luxury.*","Luxury", df$Market.Category)
df$Market.Category <- gsub(".*High-Performance.*","Luxury", df$Market.Categor
y)
df$Market.Category <- gsub(".*Performance.*","Luxury", df$Market.Category)
df$Market.Category <- gsub(".*Flex Fuel.*","Normal", df$Market.Category)
df$Market.Category <- gsub(".*Hatchback.*","Normal", df$Market.Category)
df$Market.Category <- gsub(".*Factory Tuner.*","Normal", df$Market.Category)
df$Market.Category <- gsub(".*Crossover.*","Normal", df$Market.Category)
df$Market.Category <- gsub(".*Hybrid.*","Normal", df$Market.Category)
df$Market.Category <- gsub(".*Diesel.*","Normal", df$Market.Category)
df$Market.Category <- gsub(".*N/A.*","Normal", df$Market.Category)

#Changing Years to Years Since 2001
df <- df[df$Year>2000,]
df$YearsSince2001 <- df$Year-2001
df <- subset(df, select=-c(Year))

#Remove Cars with 3 doors
df <- df[df$Number.of.Doors %in% c(2,4),]

#Remove Cars based on price
df <- df[df$MSRP > 2000,] #Removing Lemons
df <- df[df$MSRP < 650000,] #Removing ultra exclusive cars, 3 cars

#Remove Outlier Highway MPG
df <- df[df$highway.MPG < 60,]
df <- df[df$city.mpg < 60,]
```

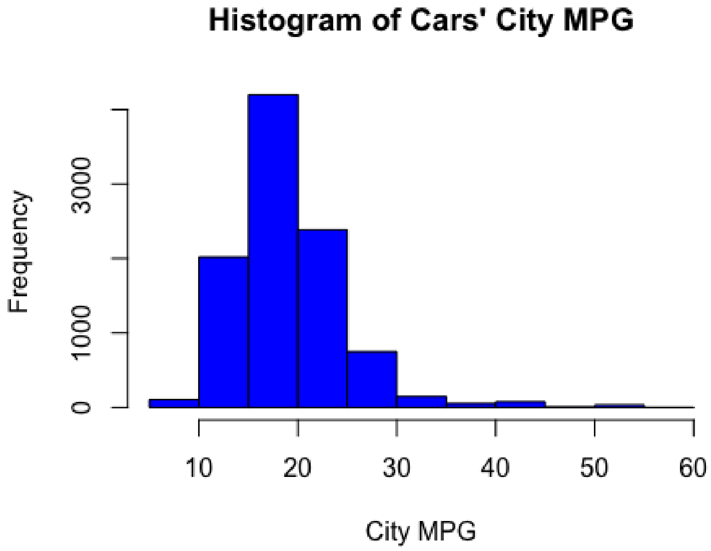
Variable Descriptions

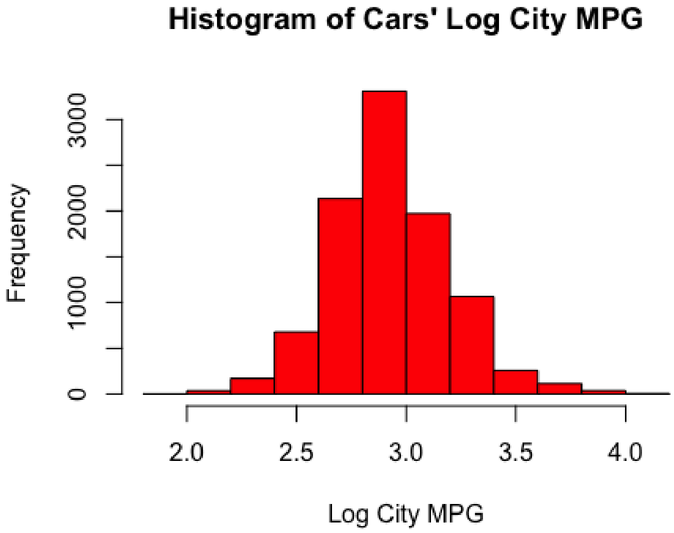
- Make = Brand of the Car (Categorical)
- Engine.Fuel.Type = Fuel type required to run a car's engine (Categorical)
- Engine.HP = Horsepower of car's engine (Continuous)
- Engine.Cylinders = Number of Cylinders in a car's engine (Ranges from V3 to V12) (Continuous)
- Transmission.Type = Whether a car is automatic, manual, or automated manual (Categorical)
- Driven_Wheels = Drive system (Four Wheel Drive = FWD, Rear Wheel Drive = RWD, All Wheel Drive = AWD, Back Wheel Drive = BWD) (Categorical)
- Number.of.Doors = Number of car doors (Categorical)
- Market.Category = Consumer market a car is aimed towards (Luxury or Normal) (Categorical)
- Vehicle.Size = Size of car (Compact, Midsize, Large) (Categorical)
- Vehicle.Style = Style of car (Convertible, Sedan, SUV, etc.) (Categorical)
- highway.MPG = Miles per gallon expected during highway use (Continuous)

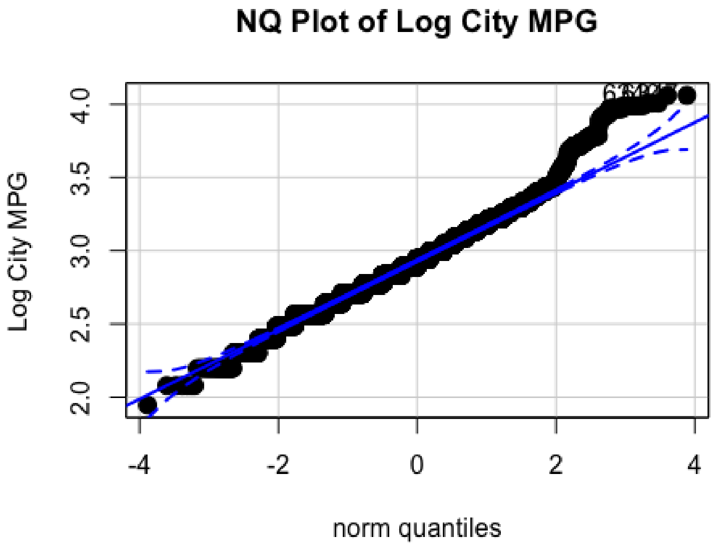
- city.mpg = Miles per gallon expected during city use (Continuous)
- MSRP = Present day Manufacturer's Suggested Retail Price of car in USD (Continuous)
- YearsSince2001 = Indicates how new a car is, Larger value means newer car (Oldest car in dataset is from 2001) (Continuous)

Data Transformations

First, some data transformations must be done. As you can see there is a heavy right skew on the city mpg data in the histogram below. The following histogram and NQ plot shows the log transformation of city mpg which is now more normally distributed. Log transformations are also performed on MSRP and highway.MPG.

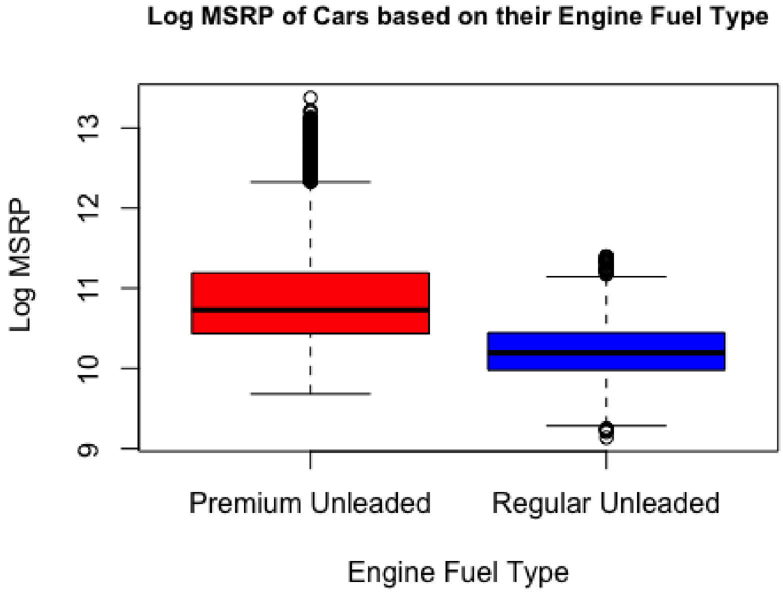


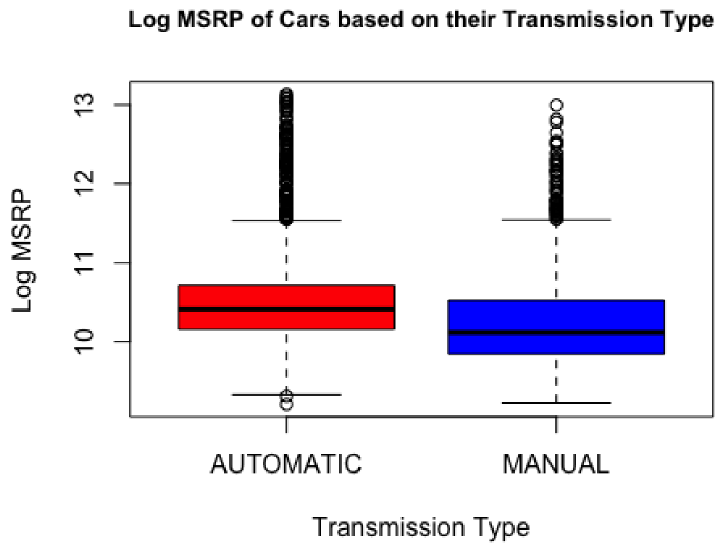




Boxplots

I wanted to see if there is a difference between the mean log MSRP for cars with Premium Unleaded and Regular Unleaded engine fuel types (Two of the most popular engine fuel types). I want to also look at the difference in the mean log MSRP based on cars' transmission types. First, I will use boxplots to visualize if there is a difference in log MSRP based on the aforementioned categorical variables.





There seems to be a relationship between log MSRP and a car's engine fuel type. From a pricing standpoint, Premium Unleaded is the most expensive while Regular Unleaded is the cheapest. It could be that cheaper cars only need cheaper fuel because the "average person" demographic is who the cheaper car is targeted to. There also seems to be a relationship between log MSRP and a car's transmission type. Automatic cars are known to be easier to drive, so it makes sense that there is a premium placed on that convenience.

Bootstrap

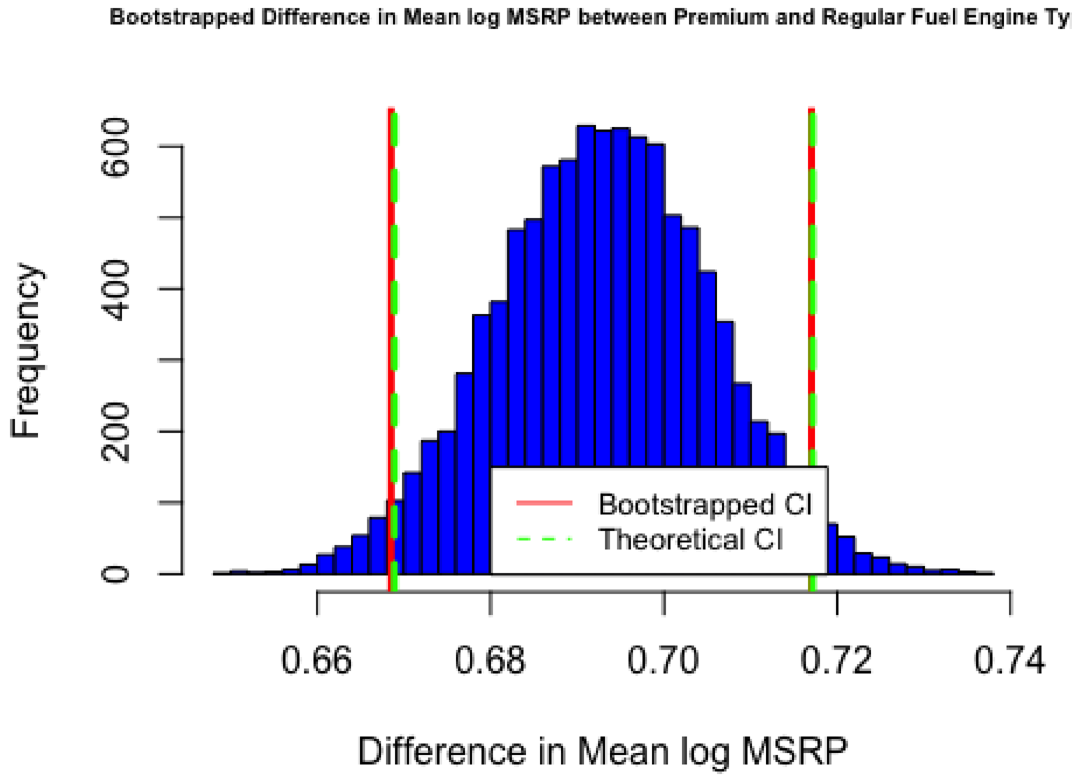
I'll now use a two-sample t-test to test the significance of the difference of mean log MSRP between Premium and Normal engine fuel type cars, and confirm my interpretation of the prior respective boxplot. The null hypothesis is that the difference in mean log MSRP from normal and premium engine fuel types is 0, and the alternative hypothesis is that the difference is not zero.

T-Test for Log MSRP by Fuel Engine Type

```
## 
##  Welch Two Sample t-test
## 
## data:  msrpPremReg$MSRP[msrpPremReg$Engine.Fuel.Type == "Premium Unleaded"
] and msrpPremReg$MSRP[msrpPremReg$Engine.Fuel.Type == "Regular Unleaded"]
## t = 56.313, df = 4739.7, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.6689051 0.7171593
## sample estimates:
## mean of x mean of y 
##  10.90980  10.21677
```

From looking at the difference in the means of log MSRP, the p-value was very small and is less than the alpha . The 95% confidence interval does not include zero. Thus, I reject the null hypothesis that the difference in mean log MSRP between premium and regular fuel engine types is 0 (There is a difference in means).

I will now create a bootstrap confidence interval for the difference in mean log MSRP between Premium Unleaded and Regular Unleaded engine fuel types (Using 10000 bootstrap samples).



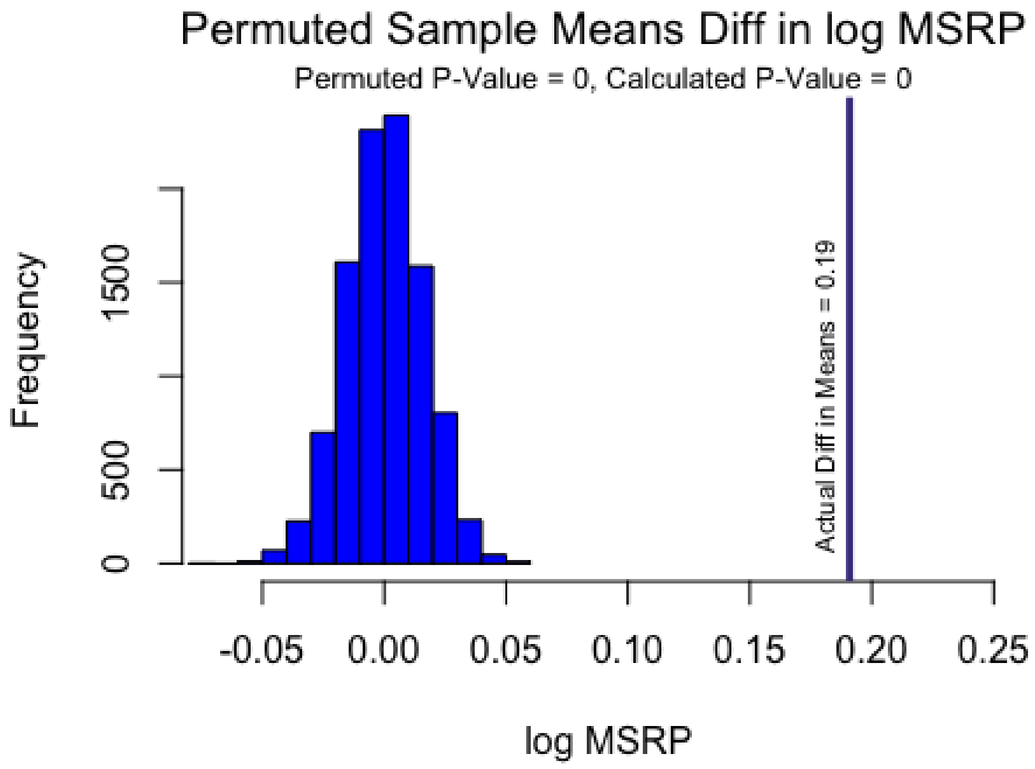
The bootstrapped confidence interval has roughly the same range as the theoretical confidence interval. We see that 0 is not included in the the bootstrapped 95% CI, which shows that there is a statistically significant difference in mean log MSRP by fuel engine type. The difference in the sample means log MSRP between Premium and Regular Fuel Engine type cars lies between 0.67 and 0.71 based on the bootstrapped interval.

Permutation

I will now use a permutation test to see how likely it is that the difference in log MSRP based on transmission type is due to random chance. First, I will use the t-test to see if there is a difference in mean log MSRP based on transmission type, which shows that there is a difference with the p-value being lower than the threshold alpha (0.05).

T-Test for Log MSRP by Transmission Type

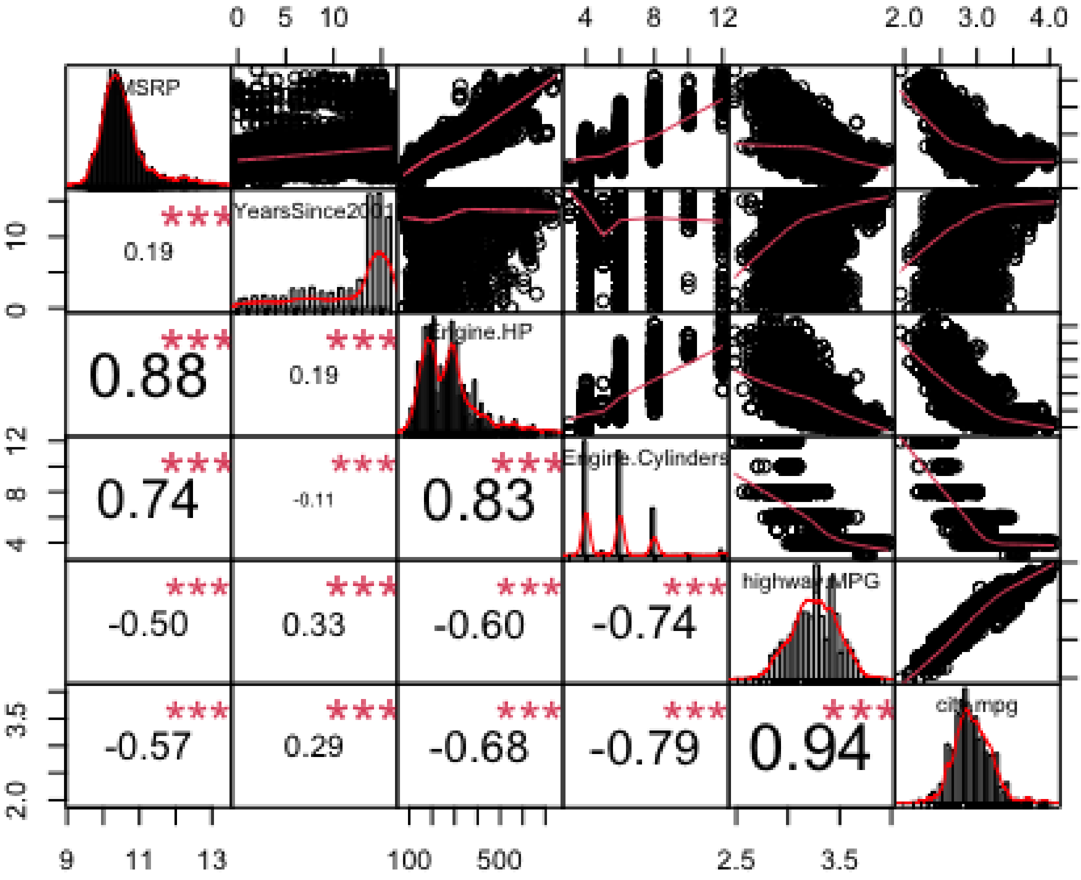
```
##
##  Welch Two Sample t-test
##
## data:  msrpPremReg$MSRP[msrpPremReg$Transmission.Type == "AUTOMATIC"] and
msrpPremReg$MSRP[msrpPremReg$Transmission.Type == "MANUAL"]
## t = 11.301, df = 2496.4, p-value < 2.2e-16
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  0.1577936 0.2240479
## sample estimates:
## mean of x mean of y
##  10.49317  10.30225
```

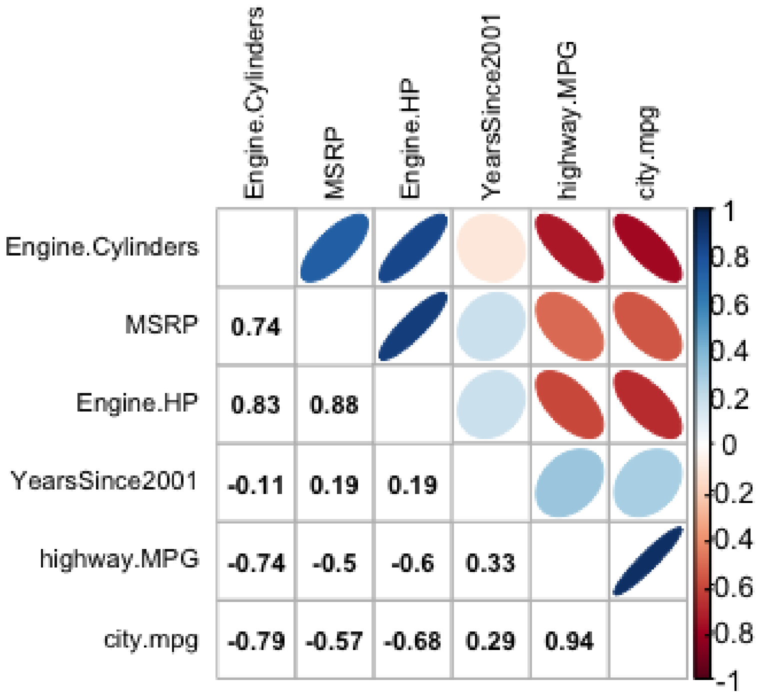


This histogram shows the results of the permutation test as well as the actual observed difference in mean log MSRP by transmission type. Almost none of the random samples produced a difference in mean log MSRP as large as the observed difference. This shows that there is a low chance that the observed difference happened by chance. This makes me conclude that there is a significant difference in mean log MSRP by transmission type.

Correlation

My intent is to create a multiple regression model, and I must see what correlations there are between log MSRP and the other continuous variables. Below are the correlation statistics between log MSRP and other variables through the matrix plot and corrplot functions.



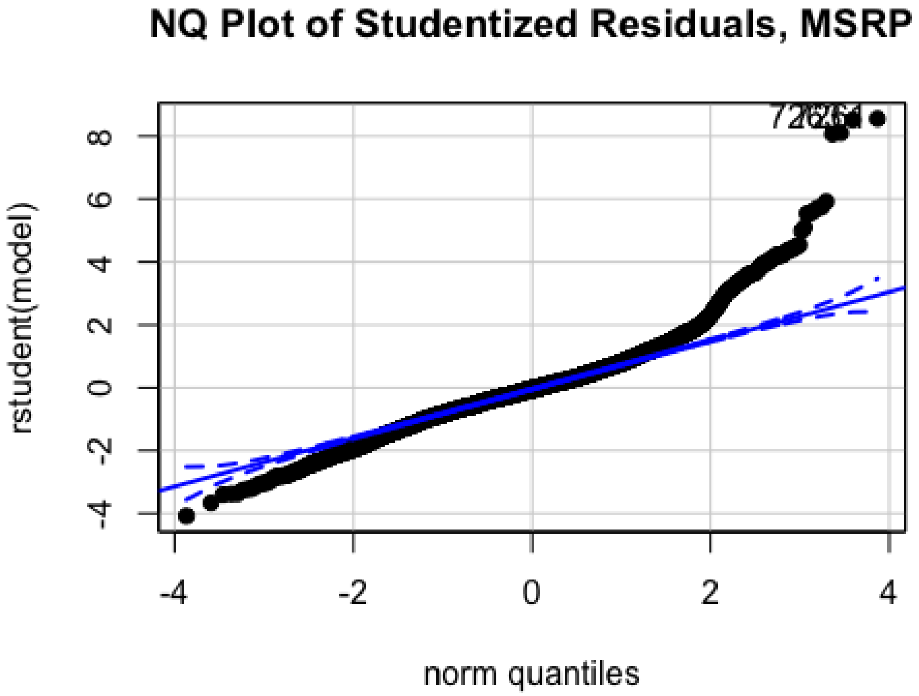


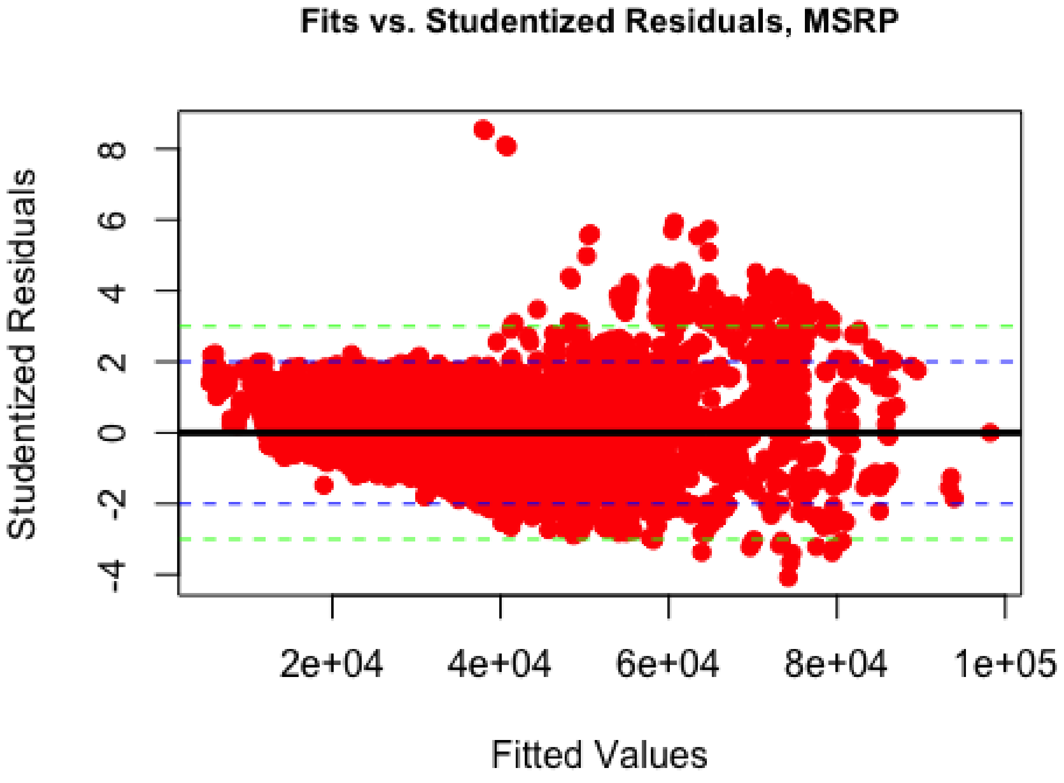
There seems to be a significant correlation between all of the continuous variables (High collinearity!). Log MSRP proves to show significant, strong, positive correlation with YearsSince2001, Engine.HP, and Engine.Cylinders, because newer and more complicated/"faster" engines are more likely to be expensive. There is a moderate negative correlation with highway mpg and city mpg probably because more expensive cars probably fall into the sports category range, and are not known for fuel efficiency.

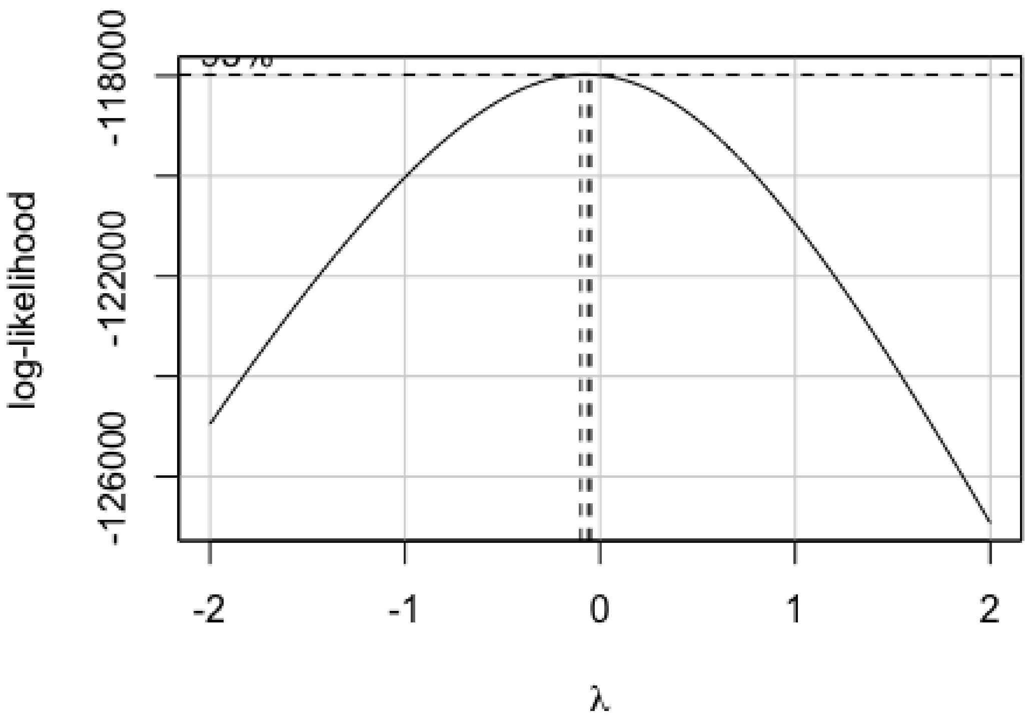
Multiple Regression

I'm interested in what factors can significantly predict the price of cars. Horsepower? Cylinders in Engine? Manufacturing date of car? I'm going to investigate through a multiple regression for predictors of car MSRP. I will be using backwards stepwise regression to determine my final model. I will be using all of the variables from my data set as predictors of price in my original model (Make, Engine.Fuel.Type, Engine.HP, Engine.Cylinders, Transmission.Type, Driven_Wheels, Number.of.Doors, Market.Category, Vehicle.Size, Vehicle.Style, highway.MPG, city.mpg, and YearsSince2001). I have further subset the data to include cars at the $100,000 or lower price point because beyond that, car prices become dependent on exclusivity factors that are not measured in this data set.

```
#Original model
dfTemp <- subset(df, df$dupMSRP <= 100000)
dfTemp <- na.omit(dfTemp[,c(15, 1:12, 14)])
mod <- lm(dupMSRP ~ ., data = dfTemp)
```


NQ Plot of Studentized Residuals, MSRP
rstudent(model)
-4
-2
0
2
4
6
8
-4
-2
0
2
4
norm quantiles


Fits vs. Studentized Residuals, MSRP
Studentized Residuals
-4
-2
0
2
4
6
8
2e+04
4e+04
6e+04
8e+04
1e+05
Fitted Values



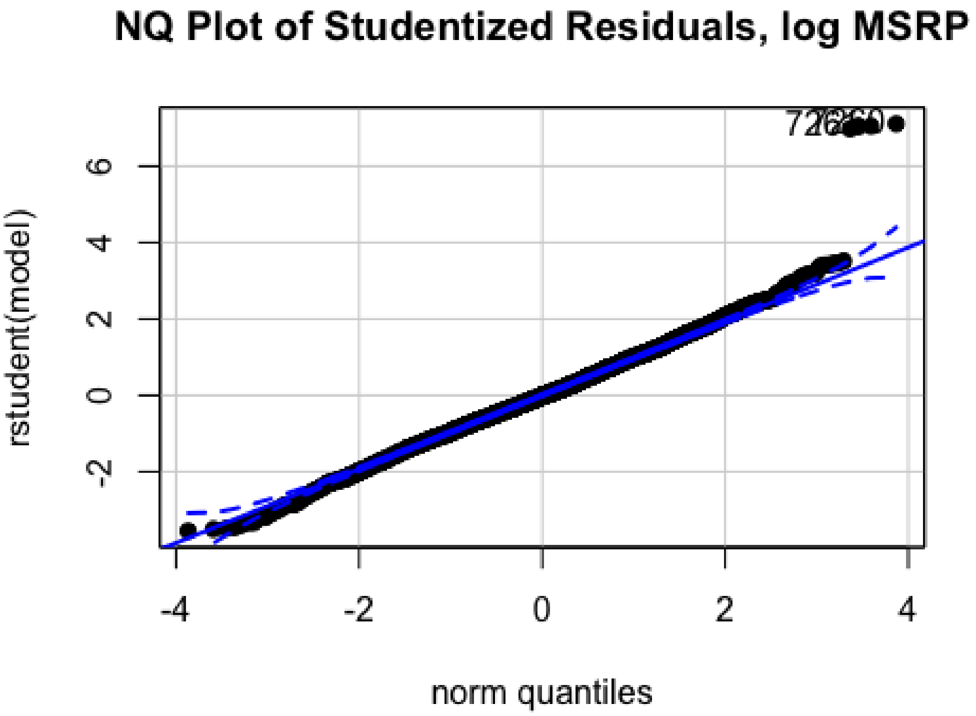
When using the normal MSRP values, there appears to be severe heteroskedasticity and a lot more outliers than I would have expected and a skewed distribution of the residuals. From the boxCox, it would be suggested that MSRP should be log transformed, which was already done earlier and will now be used in my final model. All predictors are significant based off the ANOVA table results of this model, which means I will not remove any predictors.

```
#Final Model
dfTemp <- subset(df, df$dupMSRP <= 100000)
dfTemp <- na.omit(dfTemp[,c(13, 1:12, 14)])

modBest <- lm(MSRP ~ ., data = dfTemp)

## Anova Table (Type III tests)
##
## Response: MSRP
##                     Sum Sq   Df    F value    Pr(>F)
## (Intercept)         337.50    1 18547.2503 < 2.2e-16 ***
## Make                 93.61   38   135.3792 < 2.2e-16 ***
## Engine.Fuel.Type      0.96    2    26.4600 3.484e-12 ***
## Engine.HP            53.31    1  2929.8831 < 2.2e-16 ***
## Engine.Cylinders      0.96    1    52.5975 4.430e-13 ***
## Transmission.Type     8.56    2   235.1951 < 2.2e-16 ***
## Driven_Wheels       6.67    3   122.2485 < 2.2e-16 ***
## Number.of.Doors     0.09    1     4.9878   0.02555 *  
## Market.Category     2.53    1   138.7974 < 2.2e-16 ***
## Vehicle.Size        6.55    2   180.0753 < 2.2e-16 ***
## Vehicle.Style      34.07   13   144.0148 < 2.2e-16 ***
## highway.MPG         1.66    1    91.4635 < 2.2e-16 ***
## city.mpg            3.55    1   194.9707 < 2.2e-16 ***
## YearsSince2001      6.00    1   329.4704 < 2.2e-16 ***
## Residuals         164.73 9053                         
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

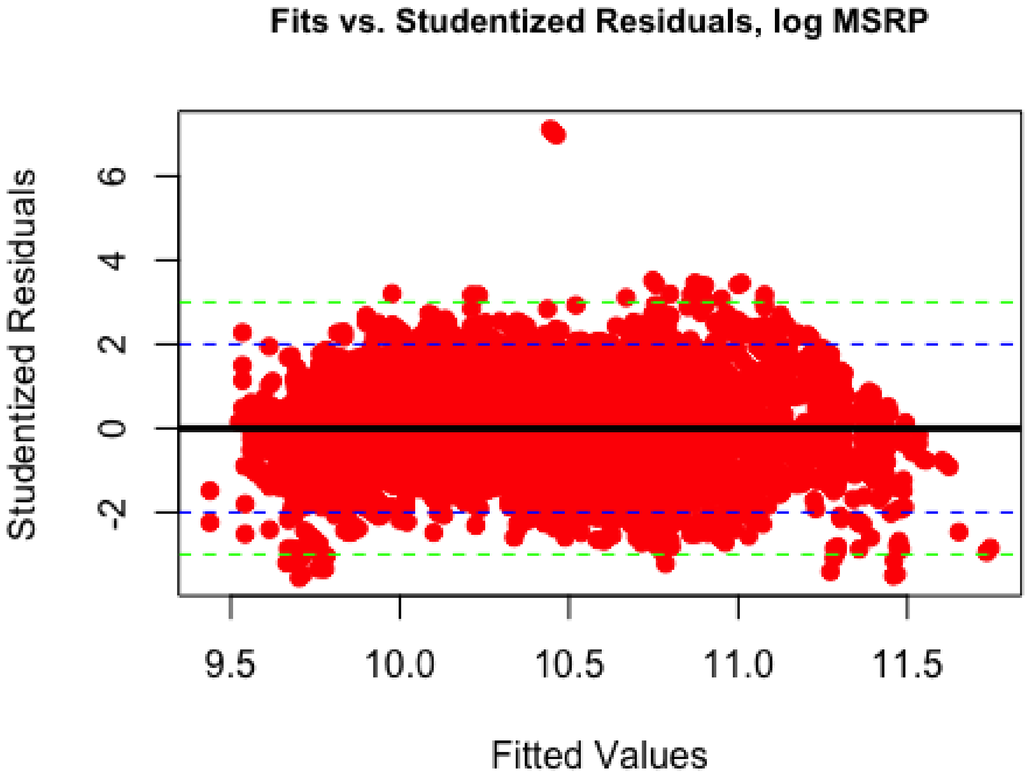
```
## 
## Call:
## lm(formula = MSRP ~ ., data = dfTemp)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -0.47827 -0.08757 -0.00305  0.08784  0.95163 
## 
## Coefficients:
##                                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)                      9.854e+00  7.236e-02 136.188  < 2e-16 **
*
## MakeAlfa Romeo                   4.438e-01  6.189e-02   7.171 8.05e-13 **
*
## MakeAston Martin                 5.510e-01  1.356e-01   4.064 4.86e-05 **
*
## MakeAudi                         9.415e-02  1.362e-02   6.911 5.14e-12 **
*
## MakeBMW                          9.630e-02  1.294e-02   7.439 1.11e-13 **
*
## MakeBuick                       -7.898e-02  1.483e-02  -5.327 1.02e-07 **
*
## MakeCadillac                     7.990e-02  1.248e-02   6.403 1.60e-10 **
*
## MakeChevrolet                   -2.196e-01  1.168e-02 -18.805  < 2e-16 **
*
## MakeChrysler                    -1.794e-01  1.525e-02 -11.766  < 2e-16 **
*
## MakeDodge                       -3.079e-01  1.269e-02 -24.267  < 2e-16 **
*
## MakeFIAT                        -2.190e-01  2.097e-02 -10.441  < 2e-16 **
*
## MakeFord                        -1.930e-01  1.221e-02 -15.809  < 2e-16 **
*
## MakeGenesis                     -1.808e-01  7.875e-02  -2.295 0.021741 * 
## MakeGMC                         -1.548e-01  1.311e-02 -11.811  < 2e-16 **
*
## MakeHonda                       -1.473e-01  1.260e-02 -11.692  < 2e-16 **
```

```
*
## MakeHUMMER                        -1.257e-01  3.507e-02  -3.584 0.000340 **
*
## MakeHyundai                       -2.295e-01  1.362e-02 -16.855  < 2e-16 **
*
## MakeInfiniti                      -1.198e-01  1.248e-02  -9.600  < 2e-16 **
*
## MakeKia                           -2.785e-01  1.415e-02 -19.683  < 2e-16 **
*
## MakeLand Rover                     5.824e-02  1.656e-02   3.517 0.000439 **
*
## MakeLexus                          3.661e-02  1.406e-02   2.603 0.009245 **
## MakeLincoln                       -1.106e-02  1.525e-02  -0.725 0.468483
## MakeLotus                          5.755e-01  2.756e-02  20.883  < 2e-16 **
*
## MakeMaserati                       4.203e-01  2.881e-02  14.588  < 2e-16 **
*
## MakeMazda                         -2.067e-01  1.301e-02 -15.890  < 2e-16 **
*
## MakeMercedes-Benz                  4.732e-02  1.412e-02   3.350 0.000811 **
*
## MakeMitsubishi                    -2.185e-01  1.481e-02 -14.759  < 2e-16 **
*
## MakeNissan                        -2.456e-01  1.245e-02 -19.727  < 2e-16 **
*
## MakeOldsmobile                    -7.525e-02  2.117e-02  -3.555 0.000380 **
*
## MakePlymouth                      -1.956e-01  9.611e-02  -2.035 0.041919 *
## MakePontiac                       -2.302e-01  1.555e-02 -14.803  < 2e-16 **
*
## MakePorsche                        3.569e-01  2.076e-02  17.190  < 2e-16 **
*
## MakeSaab                           6.119e-02  1.830e-02   3.344 0.000828 **
*
## MakeScion                         -2.743e-01  2.075e-02 -13.215  < 2e-16 **
*
## MakeSubaru                        -2.061e-01  1.405e-02 -14.671  < 2e-16 **
*
## MakeSuzuki                        -3.139e-01  1.367e-02 -22.955  < 2e-16 **
*
## MakeToyota                        -2.080e-01  1.235e-02 -16.842  < 2e-16 **
*
## MakeVolkswagen                    -9.730e-02  1.142e-02  -8.518  < 2e-16 **
*
## MakeVolvo                          6.349e-04  1.428e-02   0.044 0.964543
## Engine.Fuel.TypePremium Unleaded  4.330e-02  7.510e-03   5.765 8.42e-09 **
*
## Engine.Fuel.TypeRegular Unleaded  4.752e-03  5.998e-03   0.792 0.428224
## Engine.HP                          2.364e-03  4.367e-05  54.128  < 2e-16 **
*
```

```
## Engine.Cylinders                  1.798e-02  2.479e-03   7.252 4.43e-13 **
*
## Transmission.TypeAUTOMATIC        2.090e-02  8.508e-03   2.457 0.014048 *
## Transmission.TypeMANUAL          -7.680e-02  8.670e-03  -8.858  < 2e-16 **
*
## Driven_WheelsAWD                 -4.747e-02  6.690e-03  -7.095 1.39e-12 **
*
## Driven_WheelsFWD                 -1.248e-01  7.409e-03 -16.845  < 2e-16 **
*
## Driven_WheelsRWD                 -7.077e-02  5.906e-03 -11.981  < 2e-16 **
*
## Number.of.Doors                  -3.300e-02  1.478e-02  -2.233 0.025551 *
## Market.CategoryNormal            -6.987e-02  5.931e-03 -11.781  < 2e-16 **
*
## Vehicle.SizeLarge                 1.082e-01  5.834e-03  18.538  < 2e-16 **
*
## Vehicle.SizeMidsize               2.921e-02  4.215e-03   6.929 4.53e-12 **
*
## Vehicle.Style2dr SUV              6.741e-02  2.435e-02   2.768 0.005644 **
## Vehicle.Style4dr Hatchback        6.606e-02  3.110e-02   2.124 0.033719 *
## Vehicle.Style4dr SUV              1.741e-01  3.091e-02   5.633 1.82e-08 **
*
## Vehicle.StyleCargo Minivan        1.325e-01  3.700e-02   3.582 0.000343 **
*
## Vehicle.StyleConvertible          2.152e-01  1.035e-02  20.788  < 2e-16 **
*
## Vehicle.StyleConvertible SUV      1.098e-01  3.438e-02   3.193 0.001411 **
## Vehicle.StyleCoupe                4.065e-02  1.034e-02   3.933 8.46e-05 **
*
## Vehicle.StyleCrew Cab Pickup      9.233e-02  3.147e-02   2.934 0.003353 **
## Vehicle.StyleExtended Cab Pickup -4.101e-03  3.049e-02  -0.134 0.893016
## Vehicle.StylePassenger Minivan    2.319e-01  3.169e-02   7.319 2.72e-13 **
*
## Vehicle.StyleRegular Cab Pickup  -1.408e-01  1.435e-02  -9.811  < 2e-16 **
*
## Vehicle.StyleSedan                1.117e-01  3.077e-02   3.630 0.000284 **
*
## Vehicle.StyleWagon                1.118e-01  3.131e-02   3.571 0.000358 **
*
## highway.MPG                      -2.786e-01  2.913e-02  -9.564  < 2e-16 **
*
## city.mpg                          2.871e-01  2.056e-02  13.963  < 2e-16 **
*
## YearsSince2001                    1.021e-02  5.627e-04  18.151  < 2e-16 **
*
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## Residual standard error: 0.1349 on 9053 degrees of freedom
## Multiple R-squared:  0.8929, Adjusted R-squared:  0.8921
## F-statistic:  1126 on 67 and 9053 DF,  p-value: < 2.2e-16
```



NQ Plot of Studentized Residuals, log MSRP



Fits vs. Studentized Residuals, log MSRP

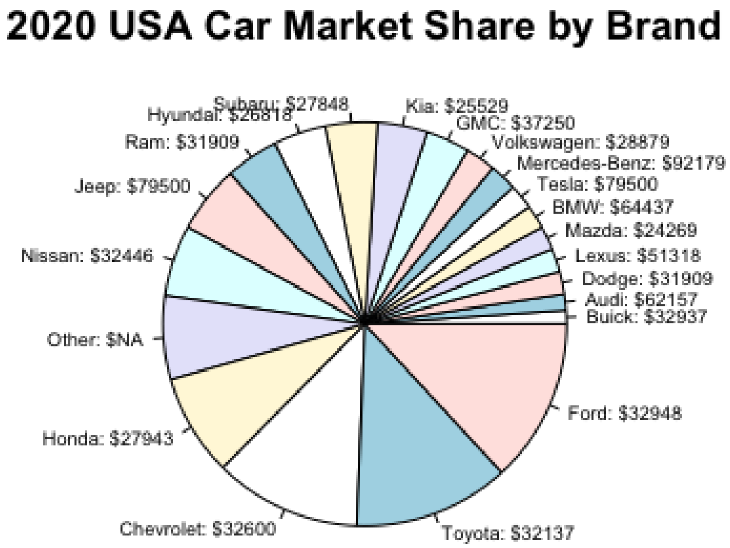
We can see that all of the coefficients in this model are significant after my use of backwards stepwise regression. I have an R-Squared of 0.89, due mostly to the fact I used several predictors. By looking at the Make coefficients, you can see that more affordable cars (Toyota, Honda) have negative coefficients while more luxury cars have positive coefficients (Land Rover, BMW), indicating that luxury type brands will cost more. Also, when looking at car size it looks like large cars have a smaller coefficeint than midsize cars, and this can perhaps be attributed to that large cars are meant for families and are supposed to be more affordable. Something interesting to note also is that when engine fuel type is regular, the predictor is not significant. Another interesting aspect is that city.MPG has a positive coefficient, but in the correlation plot, it had a negative correlation with log MSRP. This change is probably due to the fact that $100K+ cars were removed for creating this regression model. Horsepower and number of cylinders had positive coefficients as I had expected.

By looking at the normal quantile plot, we see that the residuals are much more normally distributed than before, except for a few major outliers which was expected due to some oddly configured cars in the data set. In the fit vs. residuals plot, there is little issue of heteroskedasticity and there appears to be constant variance of the residuals across all fitted values. However, there are a few outliers, with four residuals being larger than 4, but this was expected in a dataset of ~9,000 observations. Overall, this model seems to fit well for cars made after 2001 and which are under the $100K price point.

Data Scraping

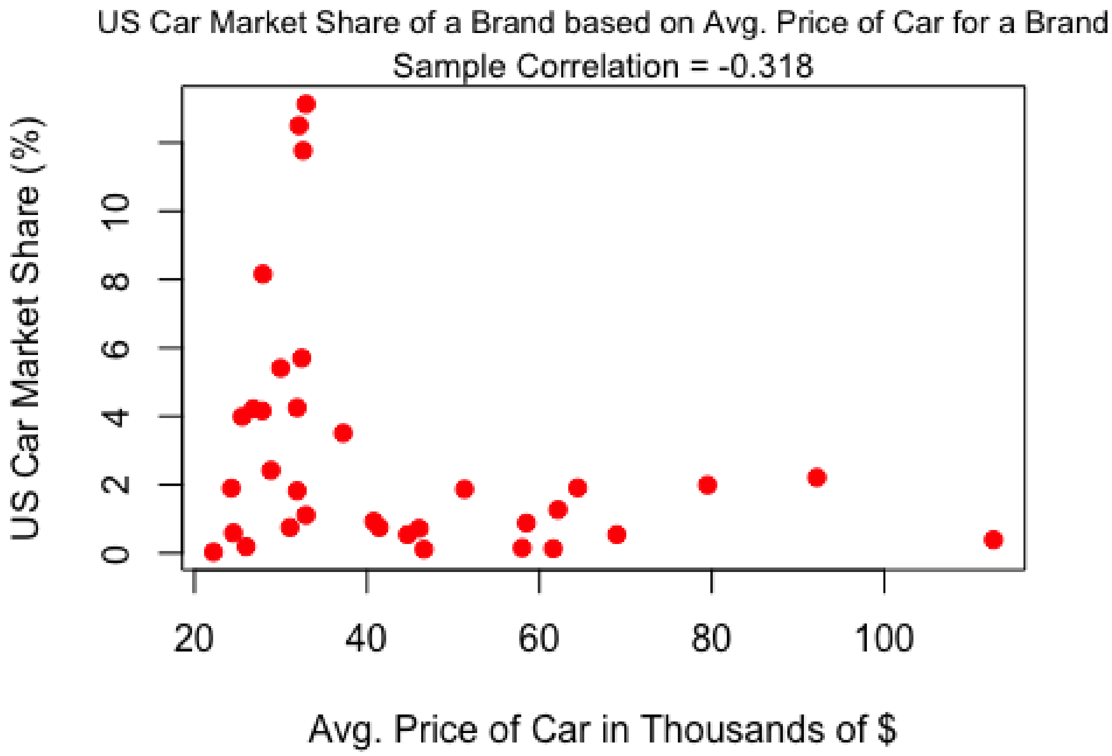
Taking a step back from cars and their characteristics, I want to look at brands and their market share of the car market. Is there a correlation between the average car price for a brand and their market share? I will scrape the 2020 car sales numbers for each brand which will give me an estimate of each brand's market share (Example: Cars sold by BMW divided by total number of cars sold in 2020). I will then add to that data each brand's average car price which is calculated from the data set I've been working with so far.

```
url <- "https://www.goodcarbadcar.net/2020-us-vehicle-sales-figures-by-brand/
"
webpage <- read_html(url)
data <- html_text(html_nodes(webpage, "td"))
data <- data[1:442]
brand <- data[seq(1, length(data), 13)]
data <- suppressWarnings(as.integer(gsub(",","", data)))
data <- na.omit(data)
frame <- matrix(data, 12)
carsSold <- colSums(frame)
Total2020CarsSold <- sum(carsSold)

cars <- data.frame(brand, carsSold)
cars$marketSharePct <- round((cars$carsSold/Total2020CarsSold)*100, 2)
cars <- cars[order(cars$marketSharePct),]
cars2 <- cars

#Make all < 1% market share brands into "Other"
cars$brand[cars$marketSharePct < 1] <- "Other"
sumOfOtherPct <- sum(cars$marketSharePct[cars$brand == "Other"])
sumOfOtherSold <- sum(cars$carsSold[cars$brand == "Other"])
other <- data.frame("Other", sumOfOtherSold, sumOfOtherPct)
names(other) <- c("brand", "carsSold", "marketSharePct")
cars <- cars[cars$brand!="Other",]
cars <- rbind(cars, other)
cars <- cars[order(cars$marketSharePct),]

cars$avgPrice <- NA
for (x in 1:length(brand)) {
  cars$avgPrice[cars$brand==brand[x]] <- round(sum(df$dupMSRP[df$Make==brand[
x]])/sum(df$Make==brand[x]),0)
  cars2$avgPrice[cars2$brand==brand[x]] <- round(sum(df$dupMSRP[df$Make==bran
d[x]])/sum(df$Make==brand[x]),0)
}
```



2020 USA Car Market Share by Brand

Avg. Car Cost for a brand is indicated in $



US Car Market Share of a Brand based on Avg. Price of Car for a Brand
Sample Correlation = -0.318

There is a negative correlation between a brand’s average price of car and their market share. This makes sense because, in general, brands like Toyota and Ford are catering to the general public and must be affordable, where as brands like BMW cater to a nicher demographic and make up in lost marketshare by selling more expensive cars. However, I believe this correlation value would higher/stronger if I had access to more car brands’ sales figures, particularly luxury brands, and, perhaps looking at the aggregate of 5 years of sales rather than a year of car sales which so happened to be during a pandemic year. I am highly skeptical of the correlation value I calculated.

Conclusions and Summary

This project has probably confirmed some of your preconceived notions of what affects car prices. It has been discovered that there is a statistically significant difference in cars’ log MSRP based on their fuel engine types and transmission type through the use of t-tests, bootstrapping, and permutation tests. We also discovered that car’s miles per gallon, brand, manufacturing date, and horsepower are some of the signifcant predictors of price. While the regression model I created seems best suited for sub $100k cars, it would be interesting to see what type of model we can fit for the super high end cars and what predictor variables would be signifcant in that model. Lastly, through the use of web scraped data, I was able to see that there is a negative correlation between a car brand’s market share and the average price of their car, but the specific correlation value should be viewd with skepticism.