Modelling Used Car Prices in R

Alejandro Vazquez

Load Libraries and Data

```
# load libraries
library(ggplot2)
library(GGally)
library(tidyr)
library(dplyr)
library(fixest)
library(car)
library(caret)

# clear environment
rm(list = ls())

# set working directory
setwd("/Users/alejandrovazquez/Desktop/econ121/car-sales-data")

# load data
df <- read.csv("Ad_table.csv")
```

Cleaning and Transformation

```
# verify data types are what they should be
sapply(df, class)
```

```
##        Maker     Genmodel  Genmodel_ID       Adv_ID     Adv_year    Adv_month
##  "character"  "character"  "character"  "character"    "integer"    "integer"
##        Color     Reg_year     Bodytype Runned_Miles   Engin_size      Gearbox
##  "character"    "integer"  "character"  "character"  "character"  "character"
##    Fuel_type        Price     Seat_num     Door_num
##  "character"  "character"    "integer"    "integer"
```

```
# change 'Runned_Miles' to integer type
df <- df %>% mutate(Runned_Miles = na_if(Runned_Miles, ""))
df$Runned_Miles <- as.integer(df$Runned_Miles)
```

```
## Warning: NAs introduced by coercion
```

```
# change 'Price' to integer type
df <- df %>% mutate(Price = na_if(Price, "Uknown"))
df$Price <- as.integer(df$Price)

# change blank values to NA
df <- df %>% mutate(Bodytype = na_if(Bodytype, "")) # Bodytype
df <- df %>% mutate(Color = na_if(Color, "")) # Color
df <- df %>% mutate(Engin_size = na_if(Engin_size, "")) # Engin_size
df <- df %>% mutate(Gearbox = na_if(Gearbox, "")) # Gearbox
df <- df %>% mutate(Fuel_type = na_if(Fuel_type, "")) # Fuel_type

# Remove 'L' from the end of the Engin_size values and convert to numeric
df$Engin_size <- gsub("L", "", df$Engin_size)
df$Engin_size <- as.numeric(df$Engin_size)

# view a summary of the data to see if any other adjustments are needed
summary(df)
```

```
##     Maker              Genmodel         Genmodel_ID           Adv_ID
##  Length:268255      Length:268255      Length:268255      Length:268255
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##     Adv_year      Adv_month         Color              Reg_year
##  Min.   :2012   Min.   : 1.000   Length:268255      Min.   :1900
##  1st Qu.:2018   1st Qu.: 4.000   Class :character   1st Qu.:2010
##  Median :2018   Median : 5.000   Mode  :character   Median :2014
##  Mean   :2018   Mean   : 5.626                      Mean   :2013
##  3rd Qu.:2018   3rd Qu.: 7.000                      3rd Qu.:2016
##  Max.   :2021   Max.   :33.000                      Max.   :2019
##                                                     NA's   :7
##    Bodytype          Runned_Miles       Engin_size         Gearbox
##  Length:268255      Min.   :      0   Min.   :   0.100   Length:268255
##  Class :character   1st Qu.:  14160   1st Qu.:   1.400   Class :character
```

```
##  Mode  :character   Median :  39296   Median :   1.800   Mode  :character
##                     Mean   :  48170   Mean   :   1.964
##                     3rd Qu.:  75000   3rd Qu.:   2.000
##                     Max.   :6363342   Max.   :3500.000
##                     NA's   :1313      NA's   :2064
##   Fuel_type            Price            Seat_num         Door_num
##  Length:268255      Min.   :    100   Min.   : 1.000   Min.   :0.000
##  Class :character   1st Qu.:   4990   1st Qu.: 5.000   1st Qu.:4.000
##  Mode  :character   Median :   9299   Median : 5.000   Median :5.000
##                     Mean   :  14756   Mean   : 4.904   Mean   :4.372
##                     3rd Qu.:  17150   3rd Qu.: 5.000   3rd Qu.:5.000
##                     Max.   :9999999   Max.   :17.000   Max.   :7.000
##                     NA's   :1145      NA's   :6474     NA's   :4553
```

There may be some mis-entered data in the month column as evidenced by the max value being 33. The number of null values shouldn't be an issue considering the size of the dataset.

```
# Lets take a look at the month outlier
month <- df %>% arrange(desc(Adv_month))
head(month['Adv_month'], 10)
```

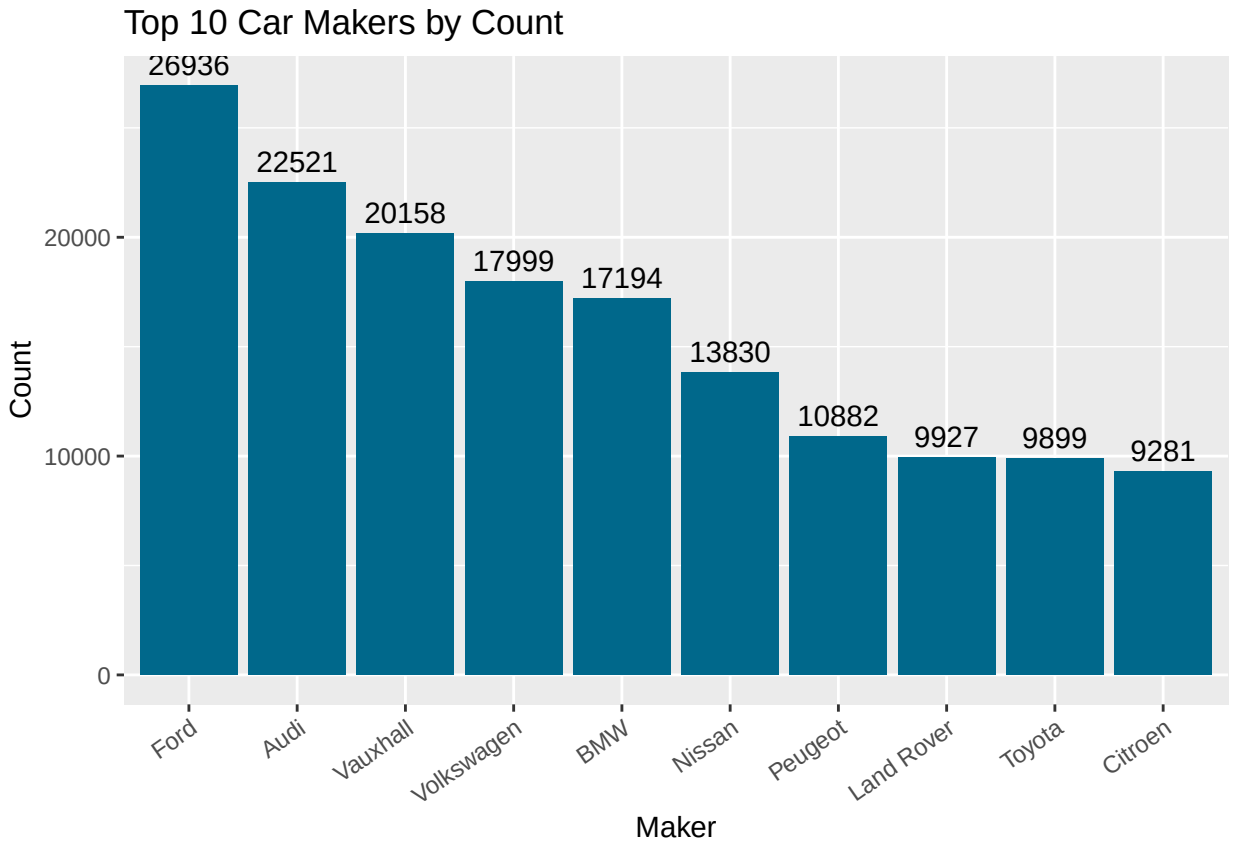
```
##    Adv_month
## 1         33
## 2         17
## 3         13
## 4         12
## 5         12
## 6         12
## 7         12
## 8         12
## 9         12
## 10        12
```

It appears that there are 3 observations where the month is above 12. I will remove them from the dataset since it is just 3 observations and won't have a big impact on the analysis.

```
# Remove month outliers
df <- df %>% filter(Adv_month <= 12)
```

Exploratory Analysis: Maker

```
# view the top 10 manufacturers represented in this dataset
make_counts <- df %>%
  group_by(Maker) %>%
  summarize(count = n()) %>%
  arrange(desc(count))

top_10 <- head(make_counts, 10) # top 10
ggplot(top_10, aes(x = reorder(Maker, -count), y = count)) +
  geom_bar(stat = "identity", fill = "deepskyblue4") +
  geom_text(aes(label = count), vjust = -0.5, position = position_dodge(width = 0.9)) +
  theme(axis.text.x = element_text(angle = 35, hjust = 1)) +
  xlab("Maker") +
  ylab("Count") +
  ggtitle("Top 10 Car Makers by Count")
```




```
summary(make_counts$count)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       1       3     296    3048    3422   26936
```

It seems that 25% of the makes in our dataset have less than 3 vehicles. Because of this we may need to remove these makes before creating dummies for 'Maker' to prevent overfitting and reduce model complexity.

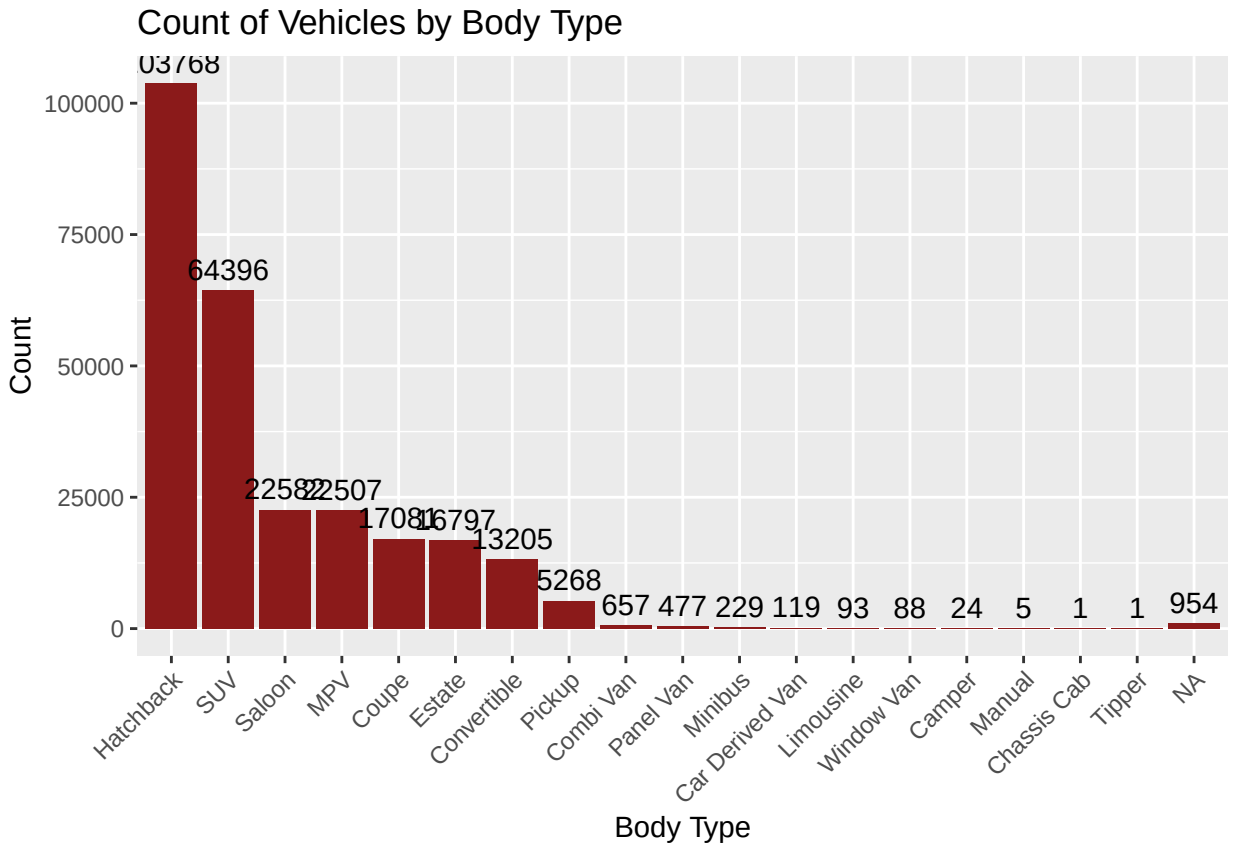
```
# Lets set a threshold at 50 vehicles.
df_fil <- df %>% filter(Maker %in% make_counts$Maker[make_counts$count >= 50])
```

I believe this is reasonable because removes makes infrequently represented in the data while maintaining a

majority of the data. Plus, this threshold ensures most high-end low-volume manufacturers like McLaren and Aston Martin remain in the dataset.

Exploratory Analysis: Body Type

```
# create a bar chart to show the body types in our dataset
bodytype_counts <- df %>%
  group_by(Bodytype) %>%
  summarise(count = n()) %>%
  arrange(desc(count))

ggplot(bodytype_counts, aes(x = reorder(Bodytype, -count), y = count)) +
  geom_bar(stat = "identity", fill = "firebrick4") +
  geom_text(aes(label = count), vjust = -0.5, position = position_dodge(width = 0.9)) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  xlab("Body Type") +
  ylab("Count") +
  ggtitle("Count of Vehicles by Body Type")
```



We will need to remove some of the body types with low counts if we wish to create a dummy variable for 'Bodytype'.

```
summary(bodytype_counts$count)
```

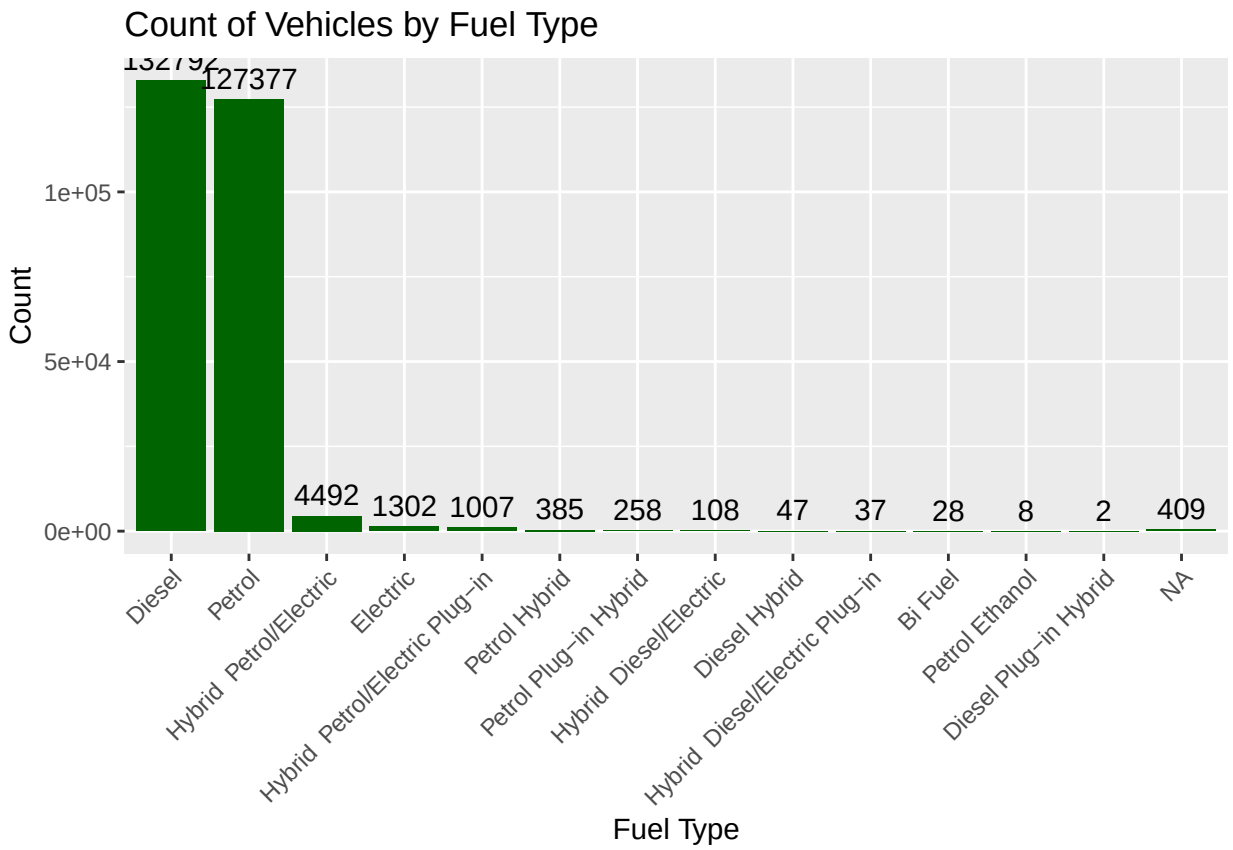
```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##      1.0     90.5    657.0  14118.5  16939.0 103768.0
```

```
# We will set a threshold at 700.
df_fil <- df_fil %>% filter(Bodytype %in% bodytype_counts$Bodytype[bodytype_counts$count >= 700])
```

This removes all the specialty body types while maintaining all the standard body types, which includes the majority of the data.

Exploratory Analysis: Fuel Type

```
# create a bar chart to show the fuel types in our dataset
fueltype_counts <- df %>%
  group_by(Fuel_type) %>%
  summarise(count = n()) %>%
  arrange(desc(count))

ggplot(fueltype_counts, aes(x = reorder(Fuel_type, -count), y = count)) +
  geom_bar(stat = "identity", fill = "darkgreen") +
  geom_text(aes(label = count), vjust = -0.5, position = position_dodge(width = 0.9)) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  xlab("Fuel Type") +
  ylab("Count") +
  ggtitle("Count of Vehicles by Fuel Type")
```



It seems the vast majority of vehicles are Diesel or Petrol (~ 97%), with a small proportion being hybrid (~ 1.7%) or electric (~ 0.5%). This is also a concern if we wish to create dummy variables for fuel type.

```
summary(fueltype_counts$count)
```

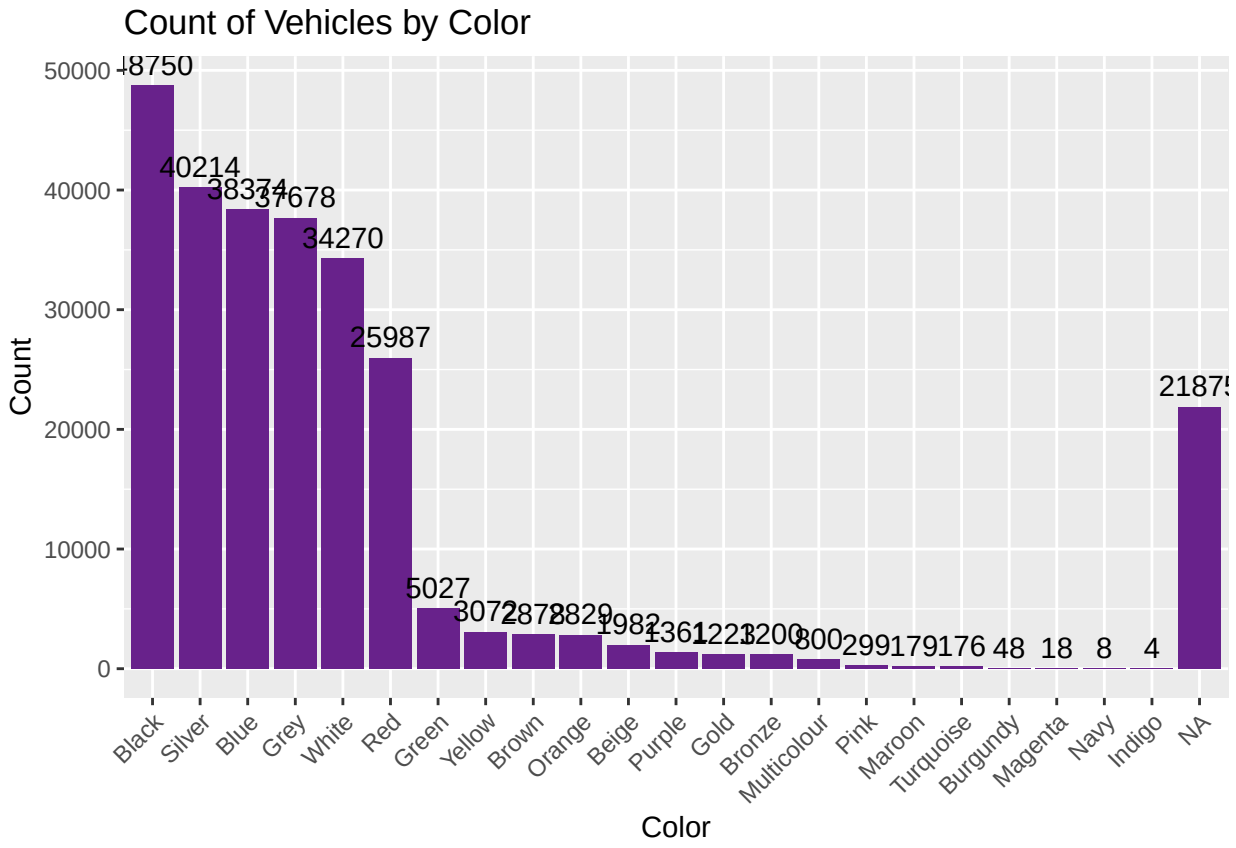
```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##      2.0     39.5    321.5  19160.9   1228.2 132792.0
```

```
# We will set a threshold at 300.
df_fil <- df_fil %>% filter(Fuel_type %in% fueltype_counts$Fuel_type[fueltype_counts$count >= 300])
```

This removes the fuel types that are not common while maintaining the most common fuel types which represent the majority of the data.

Exploratory Analysis: Color

```
# create a bar chart to show the colors in our dataset
color_counts <- df %>%
  group_by(Color) %>%
  summarise(count = n()) %>%
  arrange(desc(count))

ggplot(color_counts, aes(x = reorder(Color, -count), y = count)) +
  geom_bar(stat = "identity", fill = "darkorchid4") +
  geom_text(aes(label = count), vjust = -0.5, position = position_dodge(width = 0.9)) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  xlab("Color") +
  ylab("Count") +
  ggtitle("Count of Vehicles by Color")
```



Most of the vehicles in our dataset are black, silver, blue, grey, white, and red. If we wish to create dummy variables for color we may have to remove the other colors, but there is a large amount of NAs which may prevent us from doing so.

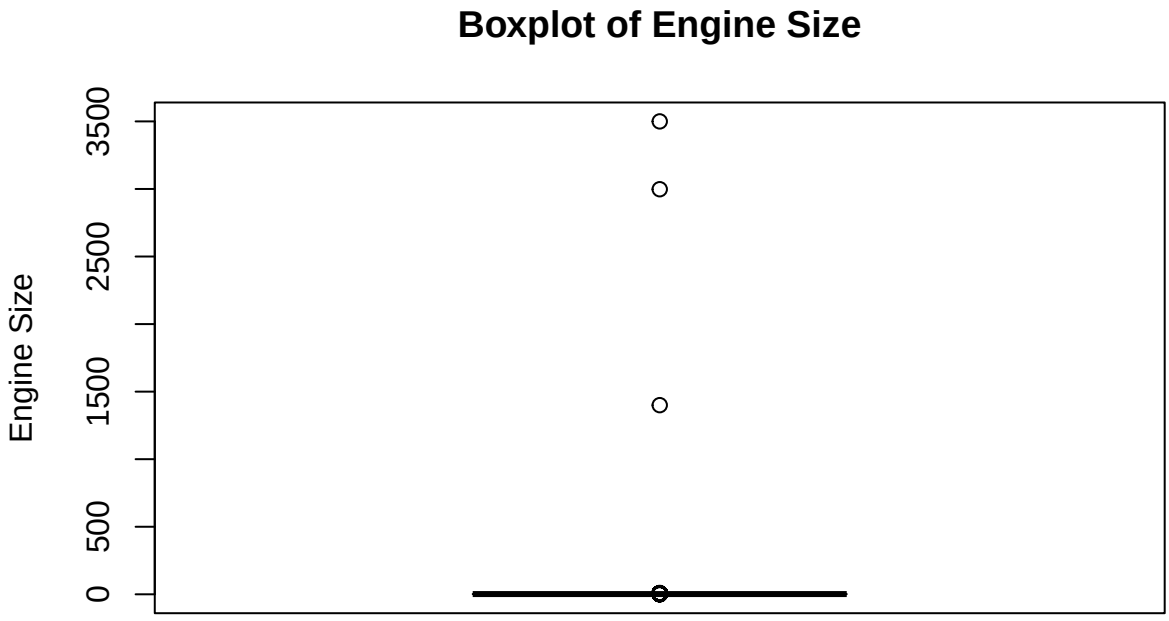
```
summary(color_counts$count)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       4     239    1982   11663   23931   48750
```

For now I won't filter out any colors because I do not think color has that large of an impact on sale price. Although I may revisit this later.

Exploratory Analysis: Engine Size

```
# plot a box plot for engine size to see outliers
boxplot(df_fil$Engin_size, main = "Boxplot of Engine Size", ylab = "Engine Size")
```



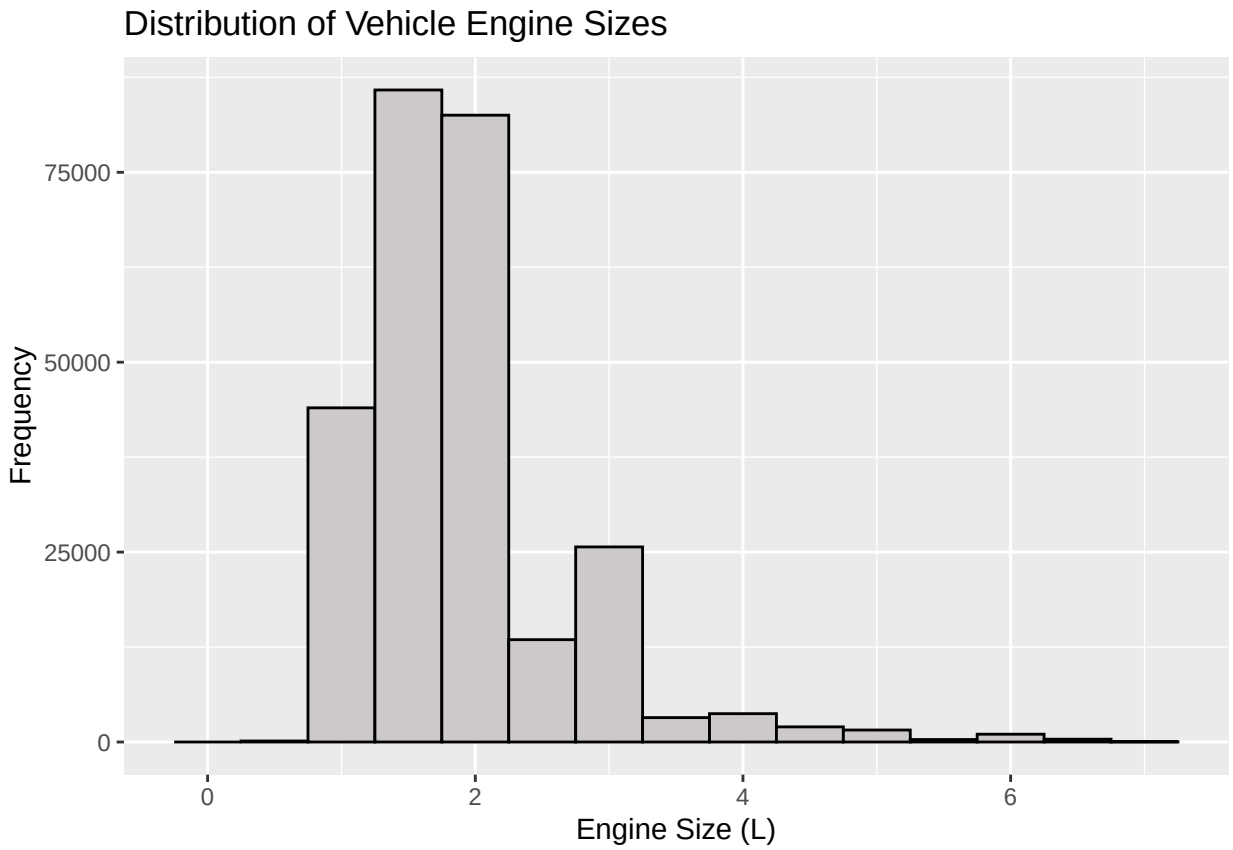
Looks like there are some significant outliers, one vehicle even has 3000 Liters! Lets remove them since it is only a few.

```
# identify outliers by engine size
IQR_values <- IQR(df$Engin_size, na.rm = TRUE)
Q1 <- quantile(df$Engin_size, 0.25, na.rm = TRUE)
Q3 <- quantile(df$Engin_size, 0.75, na.rm = TRUE)

lower_bound <- Q1 - 1.5 * IQR_values # 0.5
upper_bound <- Q3 + 1.5 * IQR_values # 2.9

outliers <- subset(df, Engin_size < lower_bound | Engin_size > upper_bound)
nrow(outliers)
```

```
## [1] 37314
```

Because there are 37,314 outliers, I think we should only remove the extreme outliers so as not to lost too much data.

```
# Remove outliers with extremely large engine size (5 observations)
df_fil <- subset(df_fil, Engin_size <= 8)

# create a histogram to show the engine sizes in our dataset
ggplot(df_fil, aes(x = Engin_size)) +
  geom_histogram(binwidth = 0.5, fill = "snow3", color = "black") +
  xlab("Engine Size (L)") +
  ylab("Frequency") +
  ggtitle("Distribution of Vehicle Engine Sizes")
```




```
# Log transformation of Engine Size to mitigate effect of outliers
df_fil$log_Engin_size <- log(df_fil$Engin_size + 1)
```

Exploratory Analysis: Gearbox

```
# The PDF will show the code AND output here.
# The PDF will show the code AND output here.
# The PDF will show the code AND output here.
# The PDF will show the code AND output here.
# The PDF will show the code AND output here.
# The PDF will show the code AND output here.
# The PDF will show the code AND output here.
```

Template

```
# The PDF will show the code you write here but not the output.
# The PDF will show the code AND output here.
```