Analysis

Justin Chan and Isaac Plotkin

3/3/2022

Narrative

Research Question

Background

One of the hardest things for investments is to decide whether the amount to pay for that investment will have a great return. Often times, people struggle choosing what is the best car for them and if they are paying a fair price for that type of car. However, it can be hard to estimate a fair price for a car due to a number of various features such as car brand, car model, engine type, horsepower, car size and many other features that can be hard to compare car prices depending on different features.

Motivation

In this project, we want to make it easier for others to be able to actually determine if they are receiving fair price for the car they are purchasing. Many salesmen try to upsell a car to make more commissions on a car where some may be overpaying for a car when comparing to its market price. To be able to determine people are getting a fair price for a car, we hope to create a linear regression model that can accurate predict car price based on these various car features.

Hypothesis

When making the linear regression model, we hypothesize that car brand (carName), type of car body (carBody), and fuel type (fueltype) will be the largest factors when determining a car's price. Furthermore, we believe that a linear regression model will be the ideal model to use to predict car prices.

Dataset

Obervations

The obervations of the dataset are pretty straight forward where each obervation or row is a car with the columns being various features of the car. The dataset includes 26 columns where one column is an obervation index and another column is car price which is the variable we are trying to predict so we have 24 input or car features for 205 observations/cars that we can use our linear regression model.

General Description of Variables

The following is the data dictionary of our dataset that gives a clear description of each variable and the type of variable (categorical, continuous/numeric) for each of our possible inputs to use in the linear model from our dataset.

{r car variables, echo=T} # print(car_data_dictionary, n = 25)

Data Collection

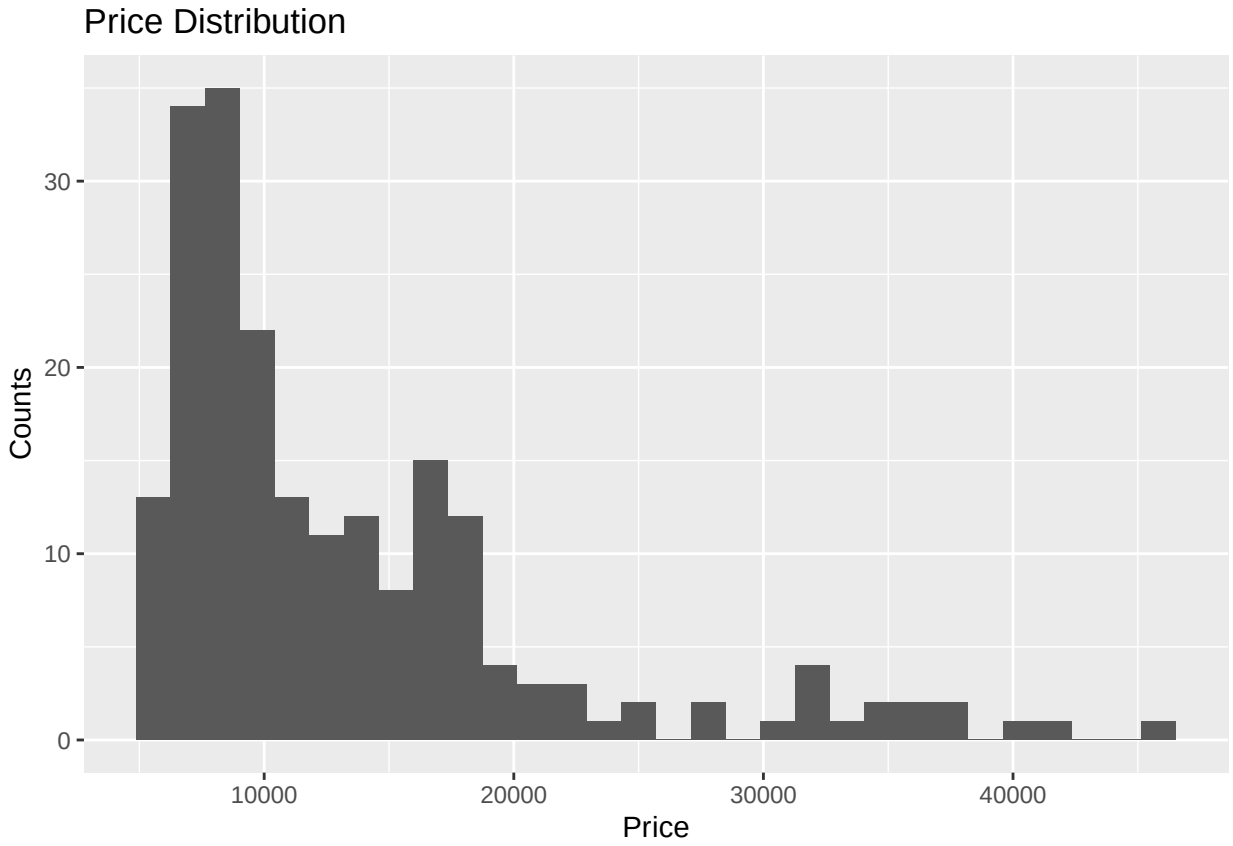
The data was originally collected from various market surveys of different types of cars across the United States market around 1987 to learn how to price cars in China depending on the American market. There is an assumption that the cars in the data set have been randomly chosen from the set of cars in the various market surveys. Link to the dataset: https://www.kaggle.com/hellbuoy/car-price-prediction

Analysis

EDA

```
ggplot(data = car_data, aes(x = price)) +
  geom_histogram() +
  labs(x = "Price",
       y = "Counts",
       title = "Price Distribution")
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
car_data %>%
  summarise(min = min(price),
            q1 = quantile(price, probs = c(0.25)),
            median = median(price),
            q3 = quantile(price, probs = c(0.75)),
            max = max(price),
            iqr = IQR(price),
            mean = mean(price),
            std_dev = sd(price)
            )
```

```
## # A tibble: 1 x 8
##     min    q1 median    q3   max   iqr   mean std_dev
##   <dbl> <dbl>  <dbl> <dbl> <dbl> <dbl>  <dbl>   <dbl>
## 1  5118  7788  10295 16503 45400  8715 13277.   7989.
```

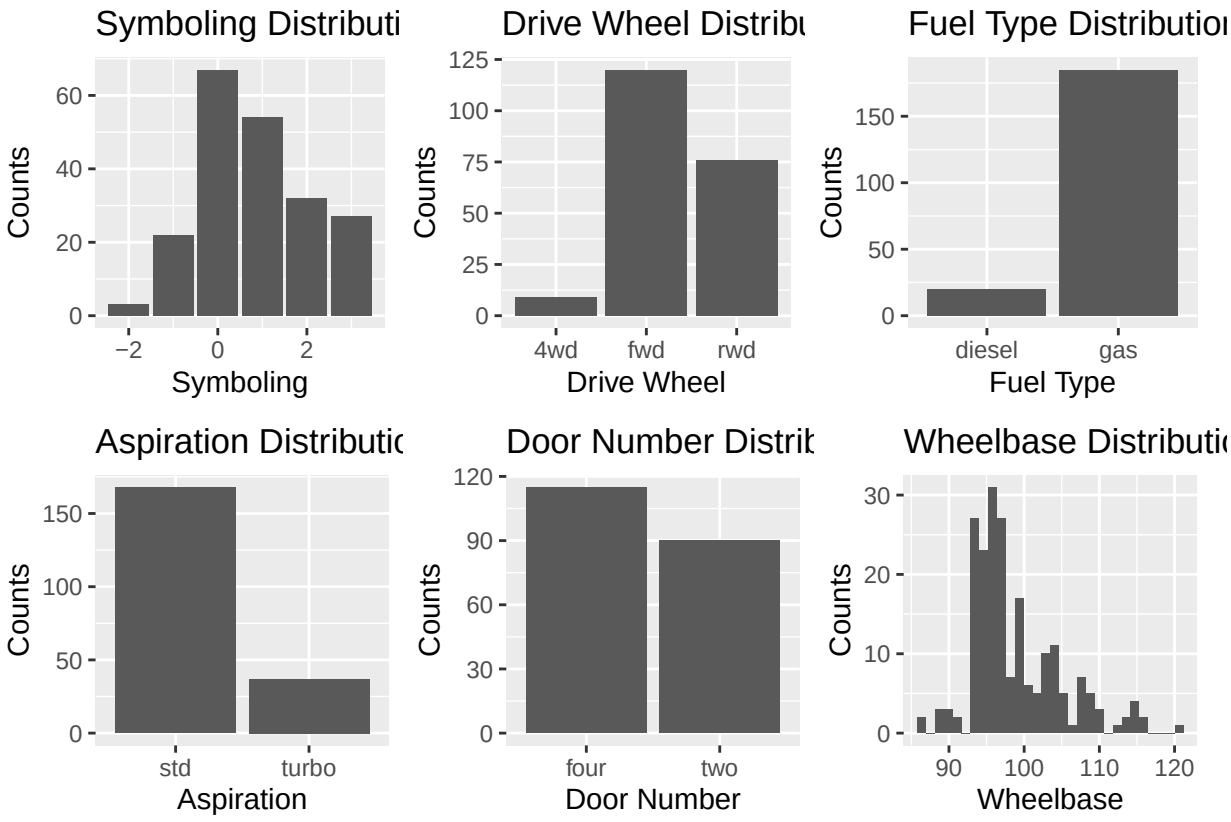
```
p1 <- ggplot(data = car_data, aes(x = symboling)) +
  geom_bar() +
  labs(x = "Symboling",
       y = "Counts",
       title = "Symboling Distribution")

p2 <- ggplot(data = car_data, aes(x = drivewheel)) +
  geom_bar() +
  labs(x = "Drive Wheel",
       y = "Counts",
```

```
       title = "Drive Wheel Distribution")

p3 <- ggplot(data = car_data, aes(x = fueltype)) +
  geom_bar() +
  labs(x = "Fuel Type",
       y = "Counts",
       title = "Fuel Type Distribution")

p4 <- ggplot(data = car_data, aes(x = aspiration)) +
  geom_bar() +
  labs(x = "Aspiration",
       y = "Counts",
       title = "Aspiration Distribution")

p5 <- ggplot(data = car_data, aes(x = doornumber)) +
  geom_bar() +
  labs(x = "Door Number",
       y = "Counts",
       title = "Door Number Distribution")

p6 <- ggplot(data = car_data, aes(x = carbody)) +
  geom_bar() +
  labs(x = "Car Body",
       y = "Counts",
       title = "Car Body Distribution")

p7 <- ggplot(data = car_data, aes(x = CarName)) +
  geom_bar() +
  labs(x = "Car Name",
       y = "Counts",
       title = "Car Name Distribution")

p8 <- ggplot(data = car_data, aes(x = wheelbase)) +
  geom_histogram() +
  labs(x = "Wheelbase",
       y = "Counts",
       title = "Wheelbase Distribution")

p9 <- ggplot(data = car_data, aes(x = carlength)) +
  geom_histogram() +
  labs(x = "Car Length",
       y = "Counts",
       title = "Car Length Distribution")

p10 <- ggplot(data = car_data, aes(x = carwidth)) +
  geom_histogram() +
  labs(x = "Car Width",
       y = "Counts",
       title = "Car Width Distribution")

p11 <- ggplot(data = car_data, aes(x = carheight)) +
  geom_histogram() +
  labs(x = "Car Height",
```

```
       y = "Counts",
       title = "Car Height Distribution")

p12 <- ggplot(data = car_data, aes(x = curbweight)) +
  geom_histogram() +
  labs(x = "Curb Weight",
       y = "Counts",
       title = "Curb Weight Distribution")

p13 <- ggplot(data = car_data, aes(x = enginetype)) +
  geom_bar() +
  labs(x = "Engine Type",
       y = "Counts",
       title = "Engine Type Distribution")

p14 <- ggplot(data = car_data, aes(x = cylindernumber)) +
  geom_bar() +
  labs(x = "Cylinder Number",
       y = "Counts",
       title = "Cylinder Number Distribution")

p15 <- ggplot(data = car_data, aes(x = enginesize)) +
  geom_histogram() +
  labs(x = "Engine Size",
       y = "Counts",
       title = "Engine Size Distribution")

p16 <- ggplot(data = car_data, aes(x = fuelsystem)) +
  geom_bar() +
  labs(x = "Fuel System",
       y = "Counts",
       title = "Fuel System Distribution")

p17 <- ggplot(data = car_data, aes(x = boreratio)) +
  geom_histogram() +
  labs(x = "Bore Ratio",
       y = "Counts",
       title = "Bore Ratio Distribution")

p18 <- ggplot(data = car_data, aes(x = stroke)) +
  geom_histogram() +
  labs(x = "Stroke",
       y = "Counts",
       title = "Stroke Distribution")

p19 <- ggplot(data = car_data, aes(x = compressionratio)) +
  geom_histogram() +
  labs(x = "Compression Ratio",
       y = "Counts",
       title = "Compression Distribution")

p20 <- ggplot(data = car_data, aes(x = horsepower)) +
  geom_histogram() +
```

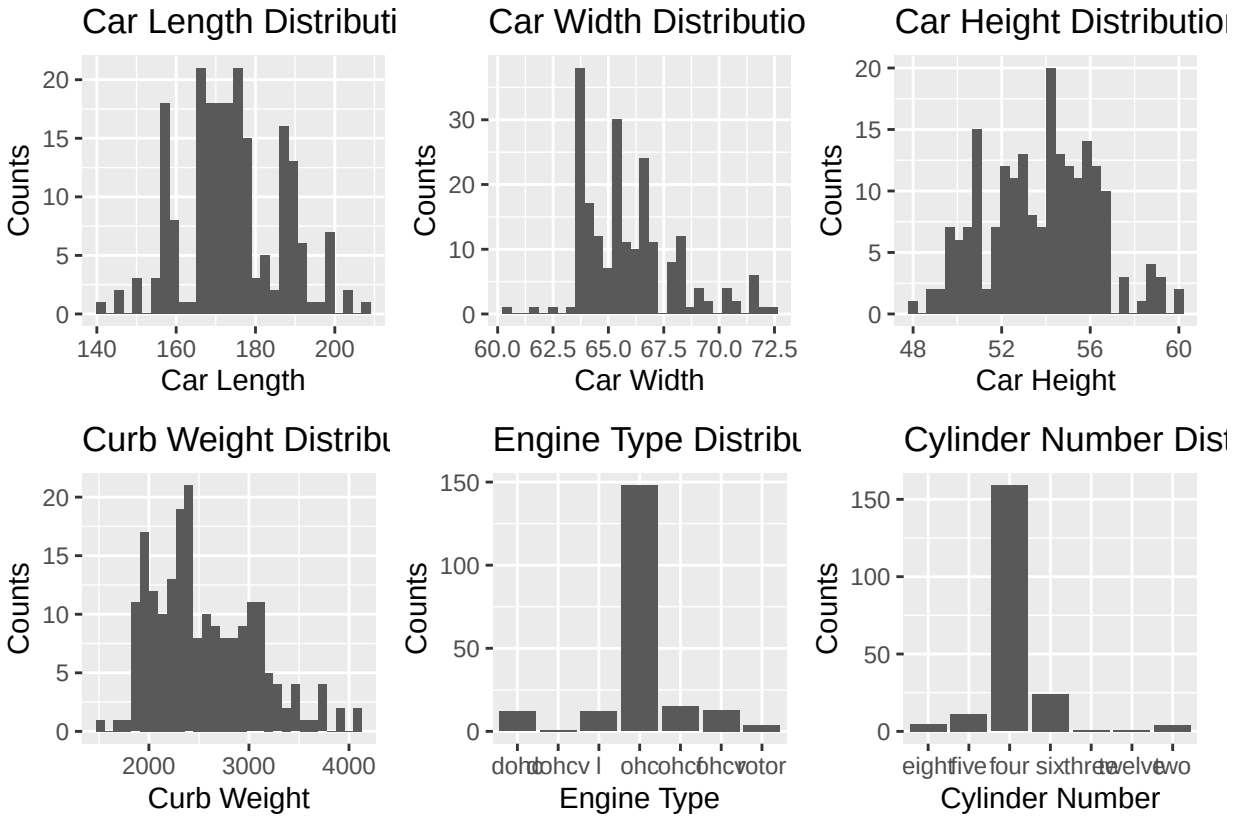
```
  labs(x = "Horsepower",
       y = "Counts",
       title = "Horsepower Distribution")

p21 <- ggplot(data = car_data, aes(x = peakrpm)) +
  geom_histogram() +
  labs(x = "Peak RPM",
       y = "Counts",
       title = "Peak RPM Distribution")

p22 <- ggplot(data = car_data, aes(x = citympg)) +
  geom_histogram() +
  labs(x = "City MPG",
       y = "Counts",
       title = "City MPG Distribution")

p23 <- ggplot(data = car_data, aes(x = highwaympg)) +
  geom_histogram() +
  labs(x = "Highway MPG",
       y = "Counts",
       title = "Highway MPG Distribution")

(p1+p2+p3)/(p4+p5+p8)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



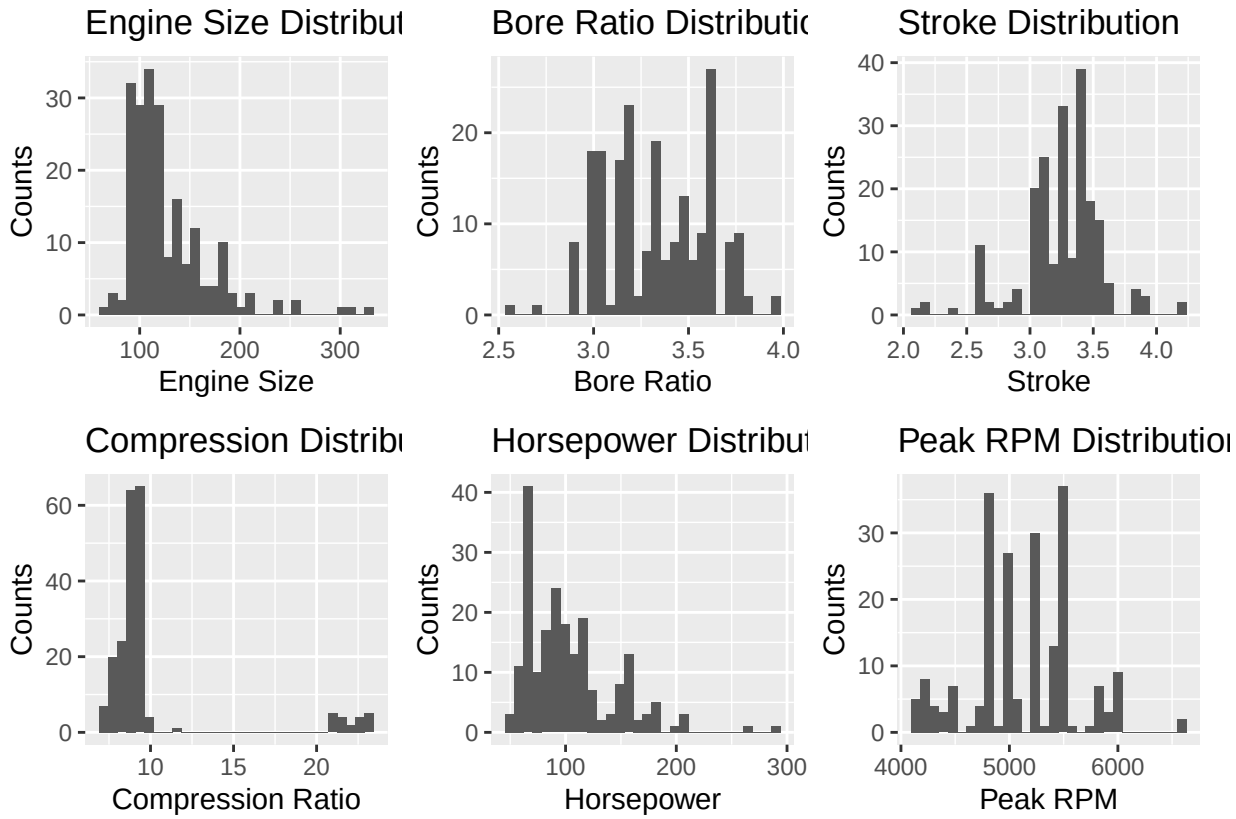
```
(p9+p10+p11)/(p12+p13+p14)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



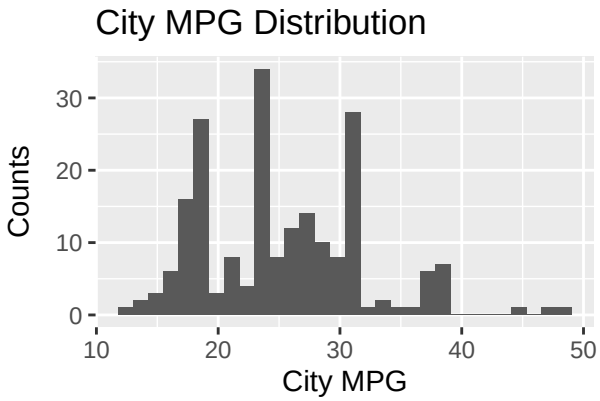

```
(p15+p17+p18)/(p19+p20+p21)
```

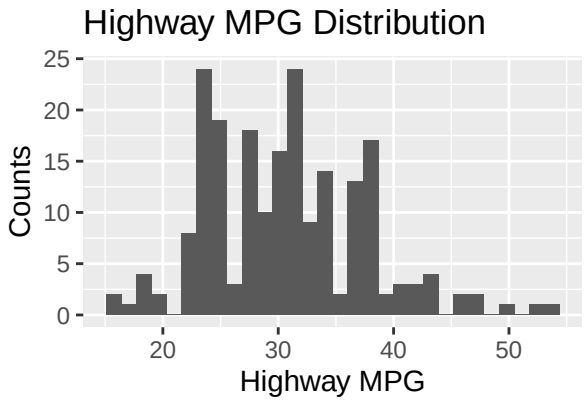
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

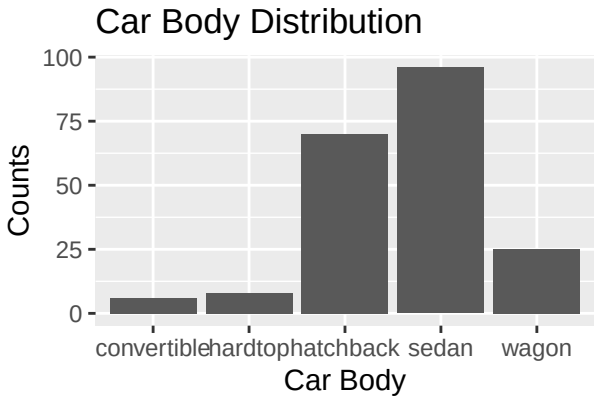


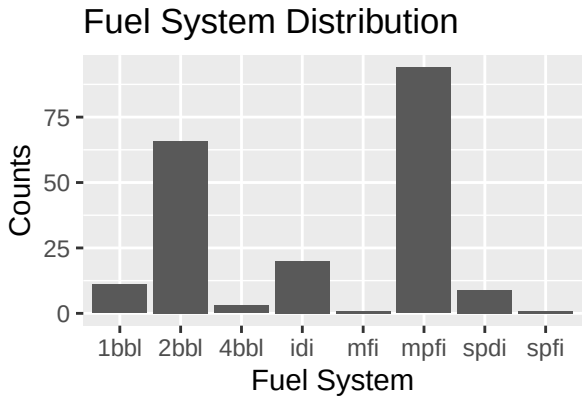

```
(p22+p23)/(p6 + p16)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

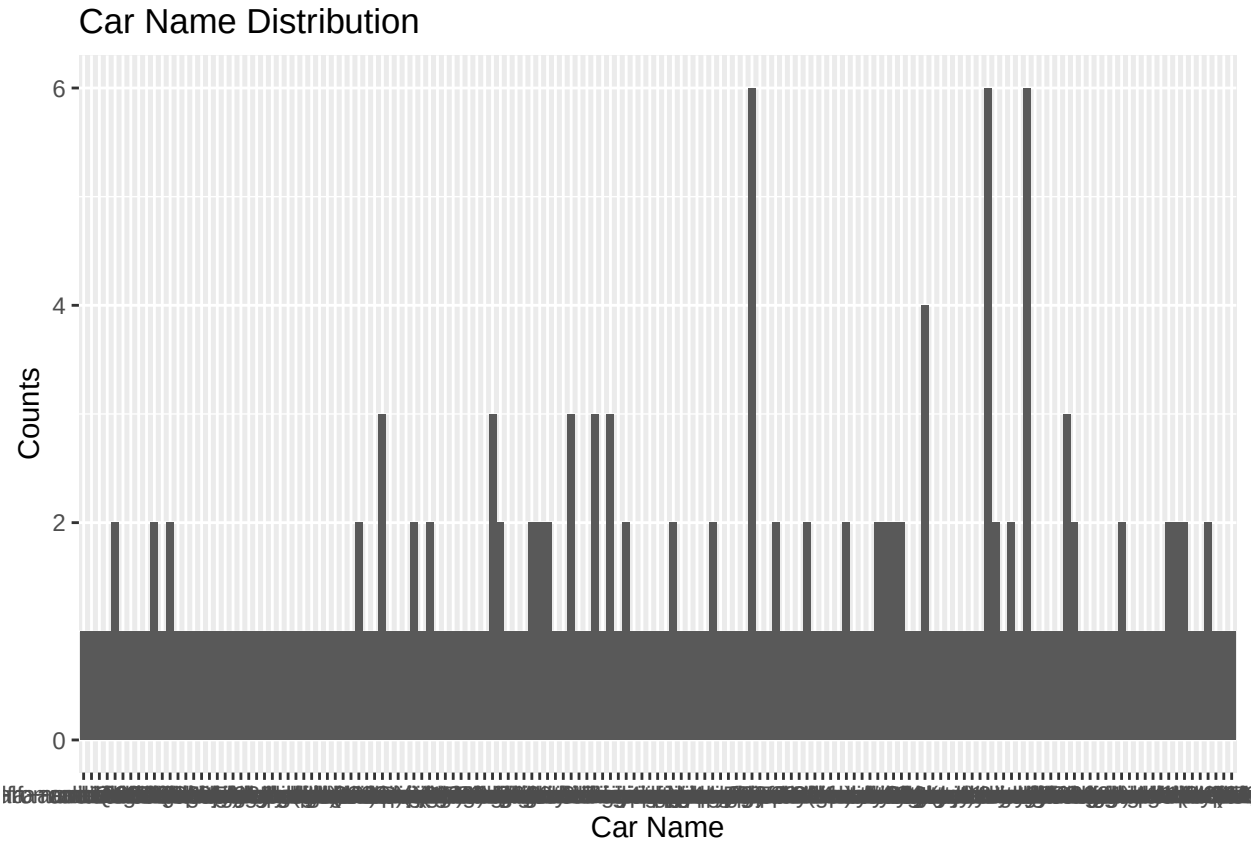

City MPG Distribution
Counts
City MPG
0
10
20
30
10
20
30
40
50


Highway MPG Distribution
Counts
Highway MPG
0
5
10
15
20
25
20
30
40
50


Car Body Distribution
Counts
Car Body
0
25
50
75
100
convertible
hardtop
hatchback
sedan
wagon


Fuel System Distribution
Counts
Fuel System
0
25
50
75
1bbl
2bbl
4bbl
idi
mfi
mpfi
spdi
spfi

```
p7
```


Car Name Distribution
Counts
6
4
2
0
Car Name