EDA and Data Cleaning on the `mtcars` Dataset*

Masud Rahman

2024-08-03

Table of contents

List of Figures

*https://masud90.github.io/, https://x.com/masudtweets/

1 Introduction

The `mtcars` dataset is a well-known dataset in R that contains data extracted from the 1974 Motor Trend US magazine. The dataset comprises various automobile design and performance aspects for 32 cars, including miles per gallon, number of cylinders, horsepower, weight, and more. In this report, we will perform data cleaning and exploratory data analysis (EDA) to uncover the underlying patterns in the data.

1.1 Load the mtcars dataset and Inspect the first few rows of the dataset

```
data("mtcars")
head(mtcars)
```

```
                   mpg cyl disp  hp drat    wt  qsec vs am gear carb
Mazda RX4         21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
Mazda RX4 Wag     21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
Datsun 710        22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
Hornet 4 Drive    21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
Hornet Sportabout 18.7   8  360 175 3.15 3.440 17.02  0  0    3    2
Valiant           18.1   6  225 105 2.76 3.460 20.22  1  0    3    1
```

1.2 Summary statistics

```
summary(mtcars)
```

```
      mpg             cyl             disp             hp       
 Min.   :10.40   Min.   :4.000   Min.   : 71.1   Min.   : 52.0  
 1st Qu.:15.43   1st Qu.:4.000   1st Qu.:120.8   1st Qu.: 96.5  
 Median :19.20   Median :6.000   Median :196.3   Median :123.0  
 Mean   :20.09   Mean   :6.188   Mean   :230.7   Mean   :146.7  
 3rd Qu.:22.80   3rd Qu.:8.000   3rd Qu.:326.0   3rd Qu.:180.0  
 Max.   :33.90   Max.   :8.000   Max.   :472.0   Max.   :335.0  
      drat             wt             qsec             vs        
 Min.   :2.760   Min.   :1.513   Min.   :14.50   Min.   :0.0000  
 1st Qu.:3.080   1st Qu.:2.581   1st Qu.:16.89   1st Qu.:0.0000  
 Median :3.695   Median :3.325   Median :17.71   Median :0.0000  
 Mean   :3.597   Mean   :3.217   Mean   :17.85   Mean   :0.4375  
 3rd Qu.:3.920   3rd Qu.:3.610   3rd Qu.:18.90   3rd Qu.:1.0000  
 Max.   :4.930   Max.   :5.424   Max.   :22.90   Max.   :1.0000  
       am              gear            carb      
 Min.   :0.0000   Min.   :3.000   Min.   :1.000  
```

```
 1st Qu.:0.0000   1st Qu.:3.000   1st Qu.:2.000
 Median :0.0000   Median :4.000   Median :2.000
 Mean   :0.4062   Mean   :3.688   Mean   :2.812
 3rd Qu.:1.0000   3rd Qu.:4.000   3rd Qu.:4.000
 Max.   :1.0000   Max.   :5.000   Max.   :8.000
```

2 Pre-processing

2.1 Check for missing values in the dataset

```
missing_values <- sum(is.na(mtcars))
if (missing_values == 0) {
  print("There are no missing values in the dataset.")
} else {
  print(paste("There are", missing_values, "missing values in the dataset."))
}
```

```
[1] "There are no missing values in the dataset."
```

2.2 Inspect the data types of all columns

```
str(mtcars)
```

```
'data.frame':   32 obs. of  11 variables:
 $ mpg : num  21 21 22.8 21.4 18.7 18.1 14.3 24.4 22.8 19.2 ...
 $ cyl : num  6 6 4 6 8 6 8 4 4 6 ...
 $ disp: num  160 160 108 258 360 ...
 $ hp  : num  110 110 93 110 175 105 245 62 95 123 ...
 $ drat: num  3.9 3.9 3.85 3.08 3.15 2.76 3.21 3.69 3.92 3.92 ...
 $ wt  : num  2.62 2.88 2.32 3.21 3.44 ...
 $ qsec: num  16.5 17 18.6 19.4 17 ...
 $ vs  : num  0 0 1 1 0 1 0 1 1 1 ...
 $ am  : num  1 1 1 0 0 0 0 0 0 0 ...
 $ gear: num  4 4 4 3 3 3 3 4 4 4 ...
 $ carb: num  4 4 1 1 2 1 4 2 2 4 ...
```

2.3 Convert numerical features to factors for categorical analysis

```r
mtcars$cyl <- as.factor(mtcars$cyl)
mtcars$vs <- as.factor(mtcars$vs)
mtcars$am <- as.factor(mtcars$am)
mtcars$gear <- as.factor(mtcars$gear)
mtcars$carb <- as.factor(mtcars$carb)
```

2.4 Check the structure again to confirm data type changes

```r
str(mtcars)
```

```
'data.frame':   32 obs. of  11 variables:
 $ mpg : num  21 21 22.8 21.4 18.7 18.1 14.3 24.4 22.8 19.2 ...
 $ cyl : Factor w/ 3 levels "4","6","8": 2 2 1 2 3 2 3 1 1 2 ...
 $ disp: num  160 160 108 258 360 ...
 $ hp  : num  110 110 93 110 175 105 245 62 95 123 ...
 $ drat: num  3.9 3.9 3.85 3.08 3.15 2.76 3.21 3.69 3.92 3.92 ...
 $ wt  : num  2.62 2.88 2.32 3.21 3.44 ...
 $ qsec: num  16.5 17 18.6 19.4 17 ...
 $ vs  : Factor w/ 2 levels "0","1": 1 1 2 2 1 2 1 2 2 2 ...
 $ am  : Factor w/ 2 levels "0","1": 2 2 2 1 1 1 1 1 1 1 ...
 $ gear: Factor w/ 3 levels "3","4","5": 2 2 2 1 1 1 1 2 2 2 ...
 $ carb: Factor w/ 6 levels "1","2","3","4",..: 4 4 1 1 2 1 4 2 2 4 ...
```

3 Exploratory Data Analysis

3.1 Histogram: Miles per Gallon

```r
ggplot(mtcars, aes(x = mpg)) +
  geom_histogram(binwidth = 2, fill = "skyblue", color = "black") +
  labs(x = "Miles Per Gallon", y = "Frequency")
```

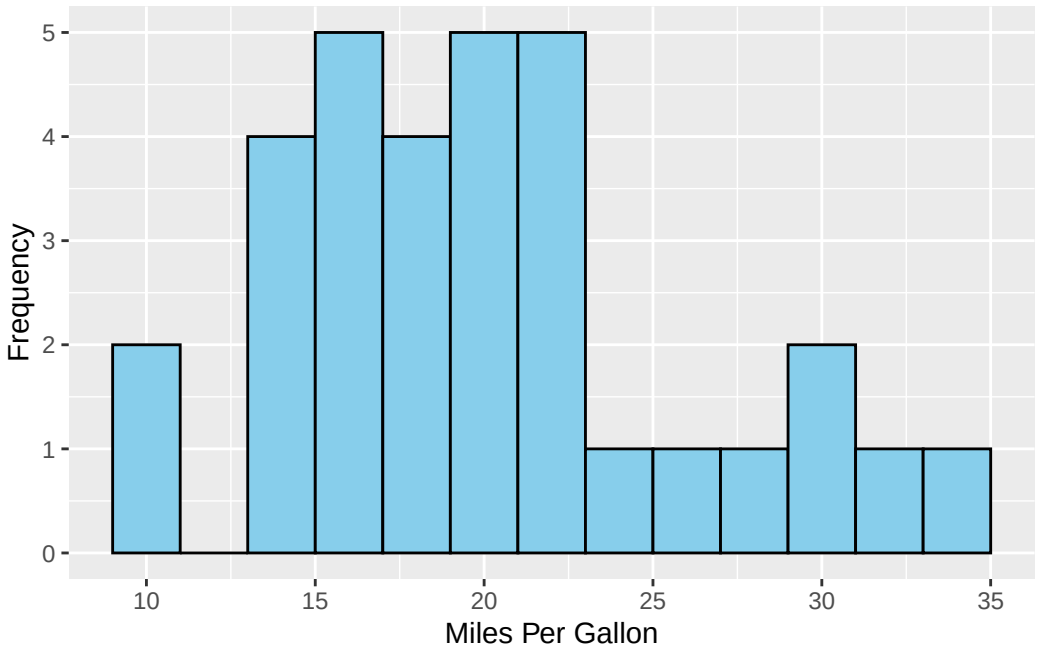

Figure 1: Histogram showing the distribution of miles per gallon (mpg) across the cars.

3.2 Boxplot of MPG by Number of Cylinders

```
ggplot(mtcars, aes(x = cyl, y = mpg, fill = cyl)) +
  geom_boxplot() +
  scale_fill_brewer(palette = "Set3") +
  labs(x = "Number of Cylinders", y = "Miles Per Gallon")
```

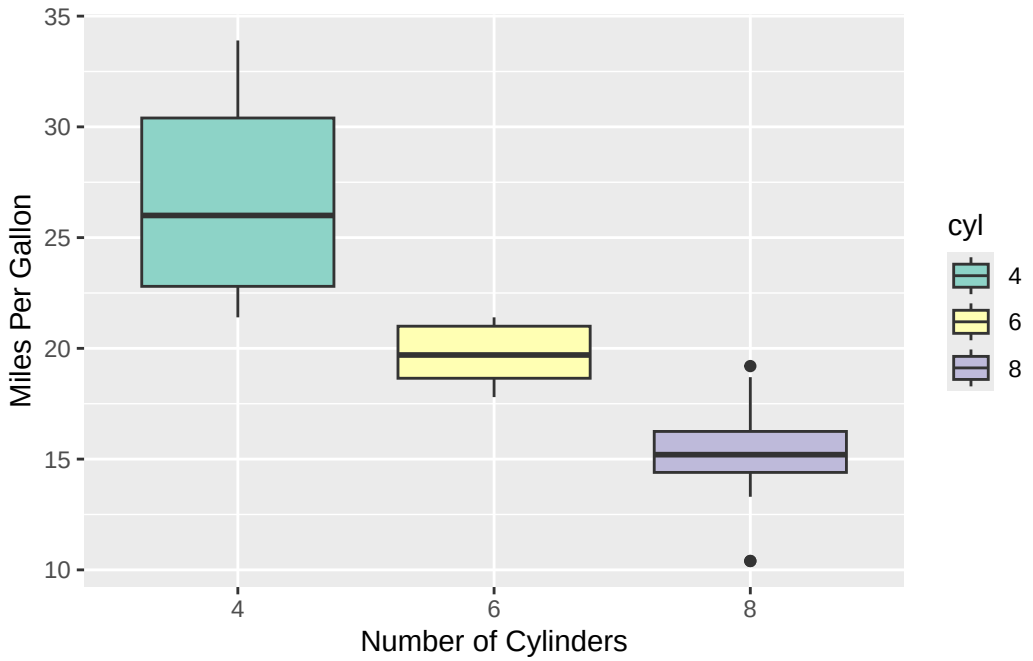

Figure 2: Boxplot depicting the variation in miles per gallon (mpg) and cylinders.

3.3 Scatter Plot of Horsepower vs. Miles Per Gallon

```
ggplot(mtcars, aes(x = hp, y = mpg)) +
  geom_point() +
  geom_smooth(method = "lm", color = "darkblue", se = FALSE) +
  labs(x = "Horsepower", y = "Miles Per Gallon")
```

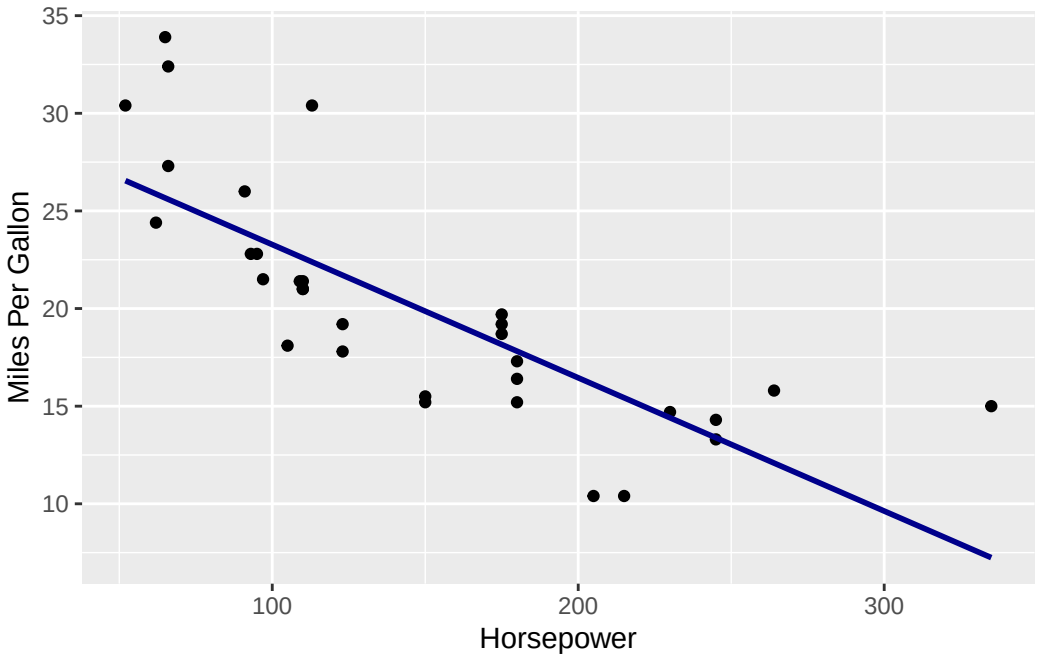

Figure 3: Scatter plot showing the relationship between horsepower and miles per gallon (mpg).

3.4 Pairwise Plot of All Variables

```
ggpairs(mtcars, columns = 1:7, ggplot2::aes(color = cyl))
```

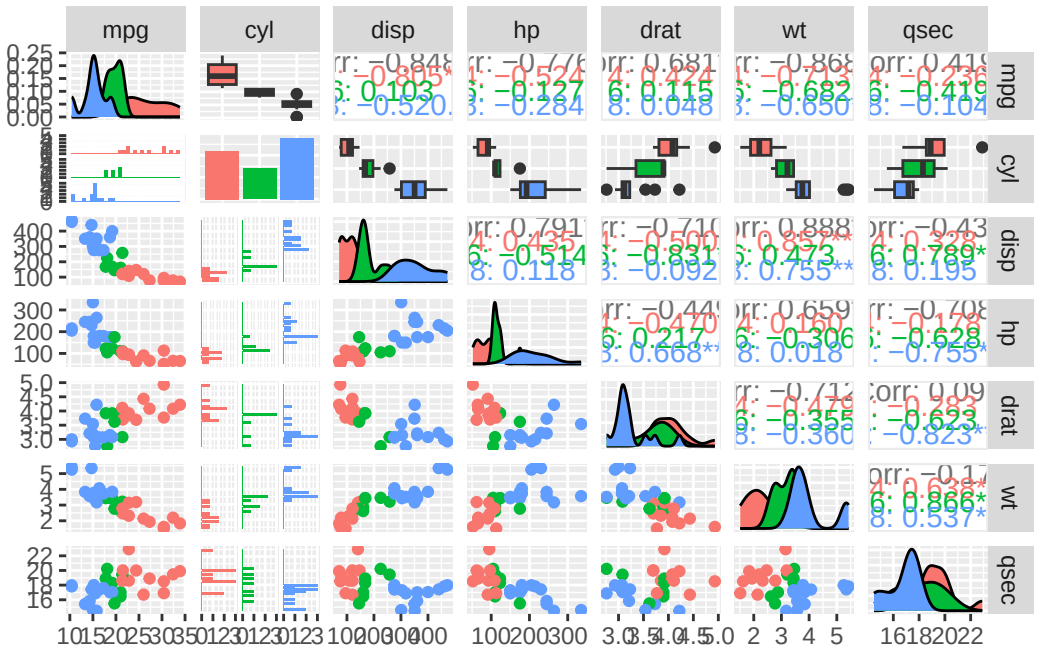

Figure 4: Pairwise plot illustrating the relationships between all numeric variables.

3.5 Correlation Heatmap

```
# Select numeric columns
mtcars_numeric <- mtcars %>%
  select_if(is.numeric)

# Calculate correlation matrix
cor_matrix <- round(cor(mtcars_numeric), 2)

# Reshape the correlation matrix for plotting
cor_melted <- melt(cor_matrix)

# Plot the correlation heatmap
ggplot(cor_melted, aes(Var1, Var2, fill = value)) +
  geom_tile() +
  scale_fill_gradient2(low = "red", high = "blue", mid = "white",
                       midpoint = 0, limit = c(-1, 1), space = "Lab",
                       name = "Pearson\nCorrelation") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, vjust = 1,
                                   size = 12, hjust = 1)) +
  coord_fixed()
```

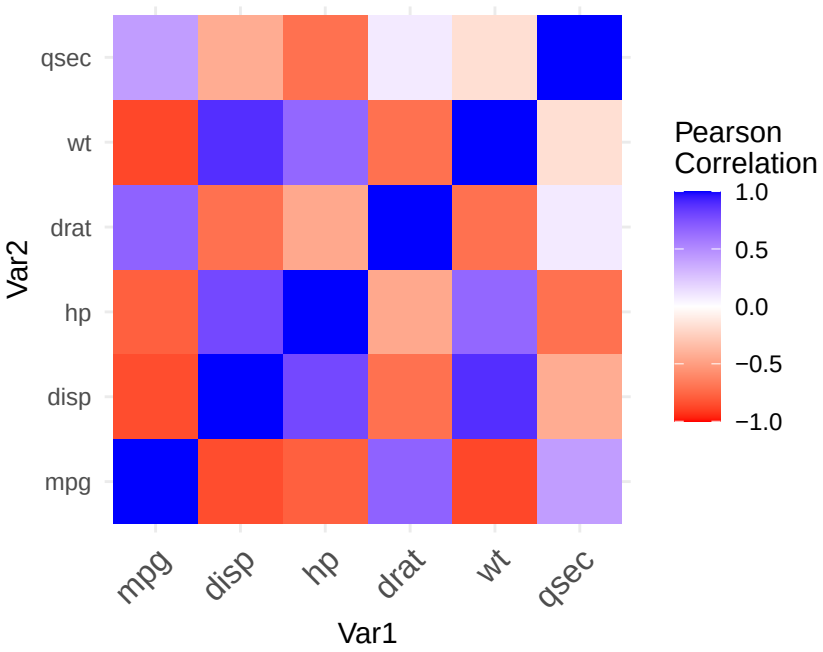

Figure 5: Heatmap showing the Pearson correlation coefficients.