RSM8512: Assignment 1

Understanding the Bias-Variance Trade-off

**Question 1: For each parts indicate whether we would generally expect the performance of a flexible statistical learning method to be better or worse than an inflexible method.**

1. **The sample size n is extremely large, and the number of predictors p is small.**

Flexible methods will perform **better** than inflexible methods here. The reason is that we have a large sample size (n), so we have lots of information to estimate f. Also, the number of predictors (p) here is small, so the curse of dimensionality is not a problem. Therefore, flexible (non-parametric) methods, which try to learn the shape of f directly from data, can do a great job here compared to inflexible methods because the state of being overwhelmed by too many predictors or features is avoided.

1. **The sample size is small, and the number of predictors p is extremely large.**

Flexible methods will perform **worse** than inflexible methods. The reason is that we have a small sample size (n), so the chance of overfitting the data while estimating f is high if we use flexible methods. Since the number of predictors (p) is extremely large, this inflates model complexity and variance, which can be controlled by using regularized versions of inflexible methods. Controlling the degrees of freedom is also important. Hence, after considering these trade-offs, inflexible methods will perform better than flexible methods in this case.

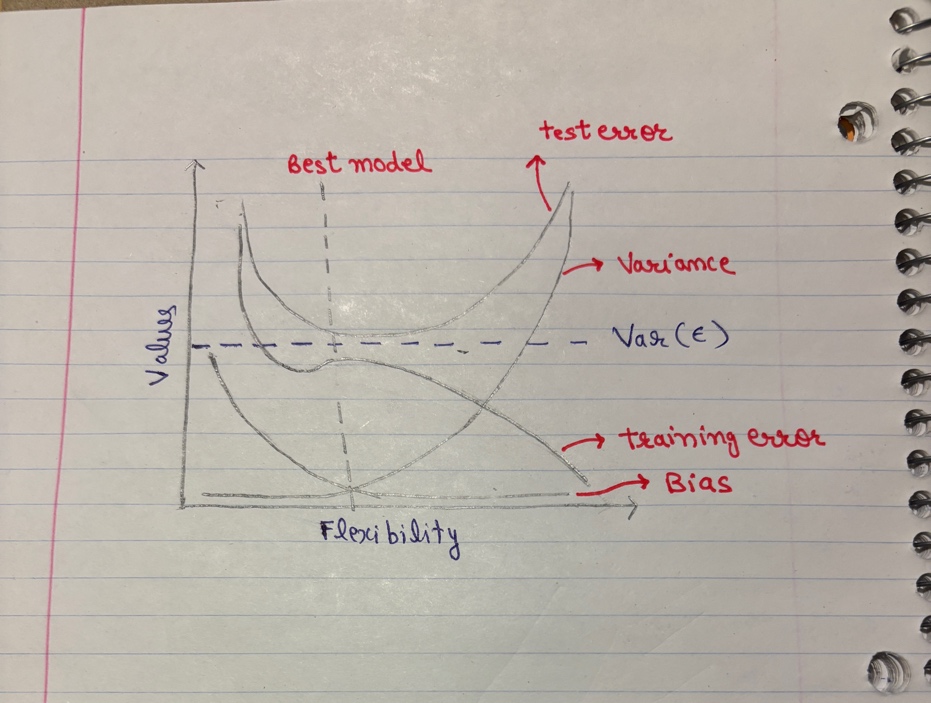
1. **The relationship between the predictors and response is highly non-linear.**

Flexible methods will perform **better** here compared to inflexible methods. The reason is that flexible methods estimate f directly from the data rather than assuming a simple form (like linear regression). As a result, they generally have lower bias than inflexible methods in highly non-linear settings. Since our goal is to achieve a model with both low bias and low variance, flexible methods are more suitable in this case.

1. **The variance of error terms is extremely high.**

This is the irreducible error and when its variance is high, it means the data has a lot of noise. We accept this as the upper bound on how well any statistical method can perform, or the lowest possible MSE we can achieve. Hence, no method (flexible or inflexible) can break through that level. Now, below this upper bound, flexible methods will perform **worse** than inflexible methods. The reason is that flexible methods will try to capture a lot of the noise, which will make the model perform poorly on the test dataset. Therefore, in high noise settings, inflexible methods are generally the safer choice.

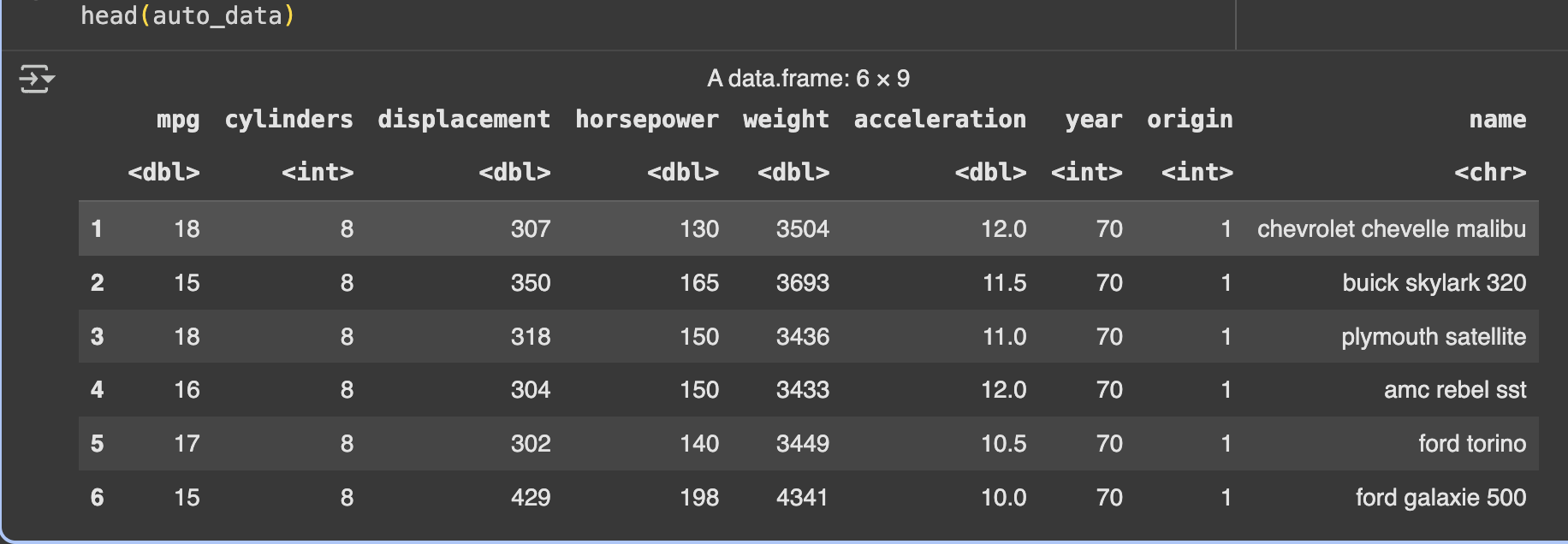
**Question 3: Provide a sketch of typical (squared) bias, variance, training error, test error, and Bayes (or irreducible) error curves, on a single plot, as we go from less flexible statistical learning methods towards more flexible approaches. The x-axis should represent the amount of flexibility in the method, and the y-axis should represent the values for each curve. There should be five curves. Make sure to label each one.**

****

1. **Bias:** As the flexibility of model increases, it tries to learn from data and relies on strong assumptions to estimate f instead of pre-assuming its shape. This reduces the systematic error (bias), since the model is better able to capture the true relationship between predictors and response.  
     
   **Variance:** The more flexible a method is, the greater its tendency to fit the training data. While this reduces bias, it also means the model is cramming the data instead of learning the true pattern. Therefore, it generalizes poorly to test data and shows higher variance because it also tries to capture noise. That’s what we called bias and variance trade off.  
     
   **Train Error:** As explained in the book, “it’s a fundamental property of statistical learning that hold regardless of the particular data set at hand and regardless of statistical method being used”. As we increase the flexibility of model, it’s training data MSE will decrease.

**Irreducible Error:** We accept the irreducible error as the upper bound of how any model can perform on a given dataset or in case of regression analysis it’s the lowest MSE which a model can achieve. This is a random error independent of X with a mean 0. Hence it stays constant because it is inherent in data and we cannot do anything about this.  
  
**Test Error:** The more flexible model is, the higher chances of it capturing noise in data and hence overfitting. Now, when we overfit the training data, the test MSE becomes very large because the patterns which training set followed (noise) doesn’t exist in training data.

**Question 9: This exercise involves Auto dataset. Make sure the missing values have been removed from the data.**

****

1. **Which predictors are qualitative and which are quantitative?**

|  |  |
| --- | --- |
| **Quantitative** | mpg, cylinders, displacement, horsepower, weight, acceleration, year |
| **Qualitative** | origin, name |

***Notes:****Origin is numerically encoded but categorical in nature, so it is included under qualitative variables  
In this assignment, mpg is assumed to be target variable*

1. **What is the range of each quantitative predictor?**

|  |  |  |
| --- | --- | --- |
| **Quantitative Predictor** | **Min Value** | **Max Value** |
| mpg | 9 | 46.6 |
| cylinders | 3 | 8 |
| displacement | 68 | 455 |
| horsepower | 46 | 230 |
| weight | 1613 | 5140 |
| acceleration | 8 | 24.8 |
| year | 70 | 82 |

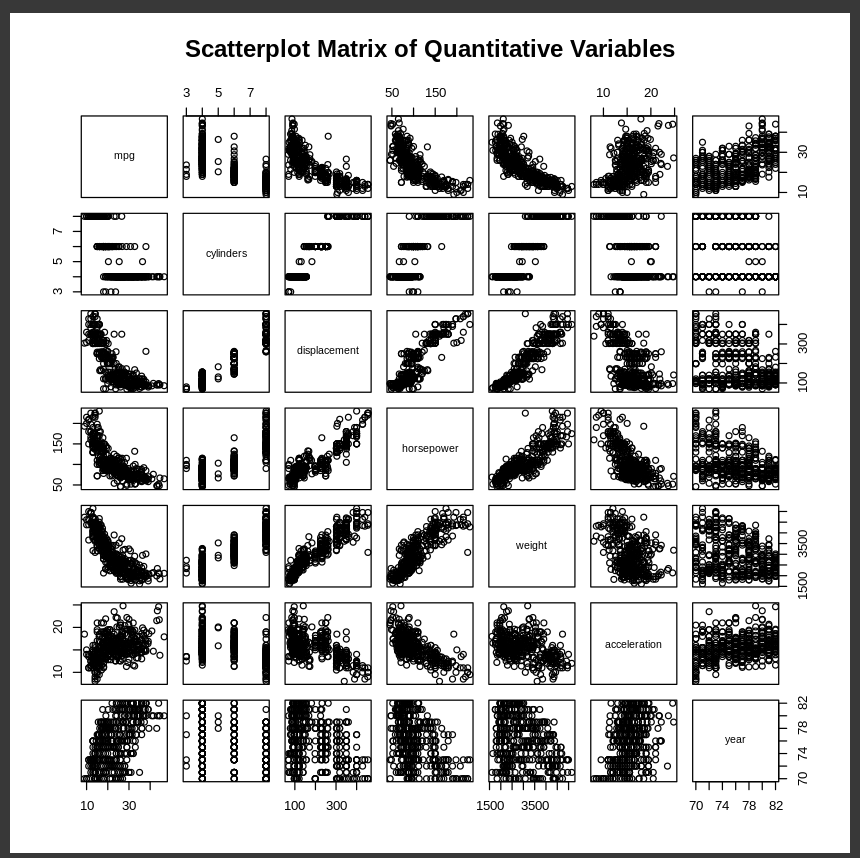
1. **What is the mean and standard deviation of each quantitative predictor?**

|  |  |  |
| --- | --- | --- |
| **Quantitative Predictor** | **Mean** | **Standard Deviation** |
| mpg | 23.466 | 7.805 |
| cylinders | 5.472 | 1.706 |
| displacement | 194.412 | 104.644 |
| horsepower | 104.469 | 38.491 |
| weight | 2977.584 | 849.403 |
| acceleration | 15.541 | 2.759 |
| year | 75.980 | 3.684 |

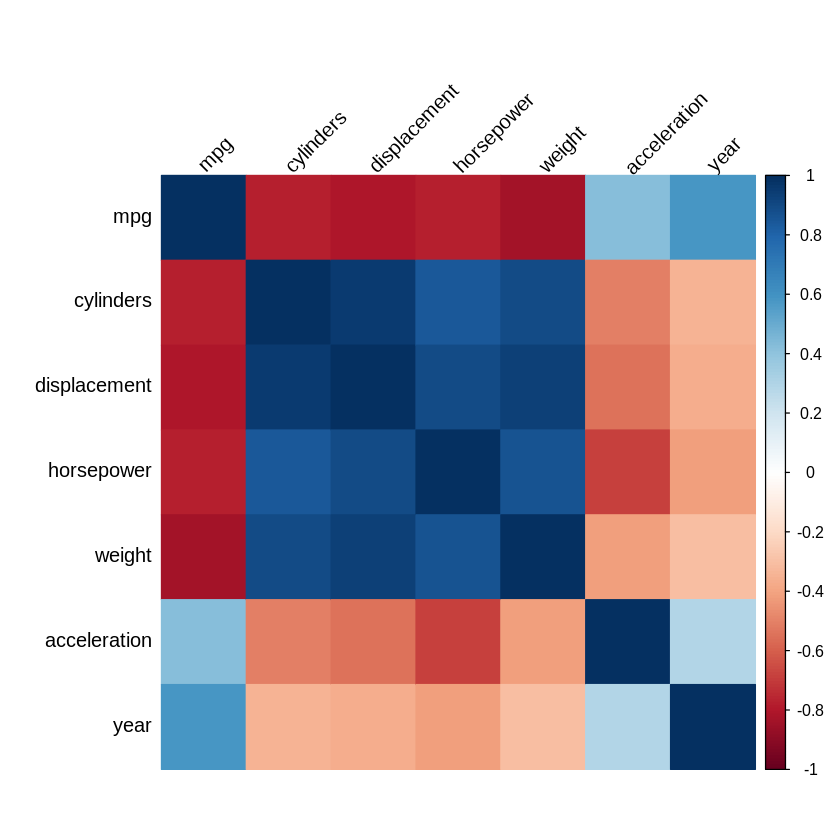
1. **Now remove the 10th through 85th observations. What is the range, mean, and standard deviation of each predictor in the subset of the data that remains?**

|  |  |  |  |
| --- | --- | --- | --- |
| **Quantitative Predictors** | **Mean** | **Standard Deviation** | **Range (Min – Max)** |
| mpg | 24.368 | 7.881 | 11 – 46.6 |
| cylinders | 5.382 | 1.658 | 3 – 8 |
| displacement | 187.754 | 99.939 | 68 – 455 |
| horsepower | 100.956 | 35.896 | 46 – 230 |
| weight | 2939.644 | 812.650 | 1649 – 4997 |
| acceleration | 15.718 | 2.694 | 8.5 – 24.8 |
| year | 77.132 | 3.110 | 70 – 82 |

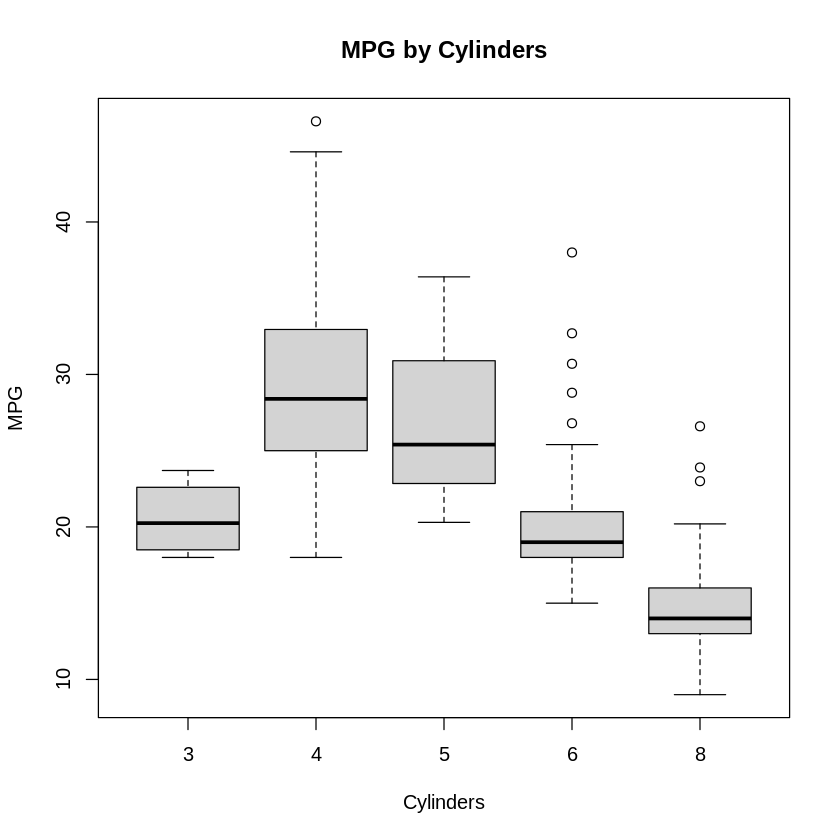
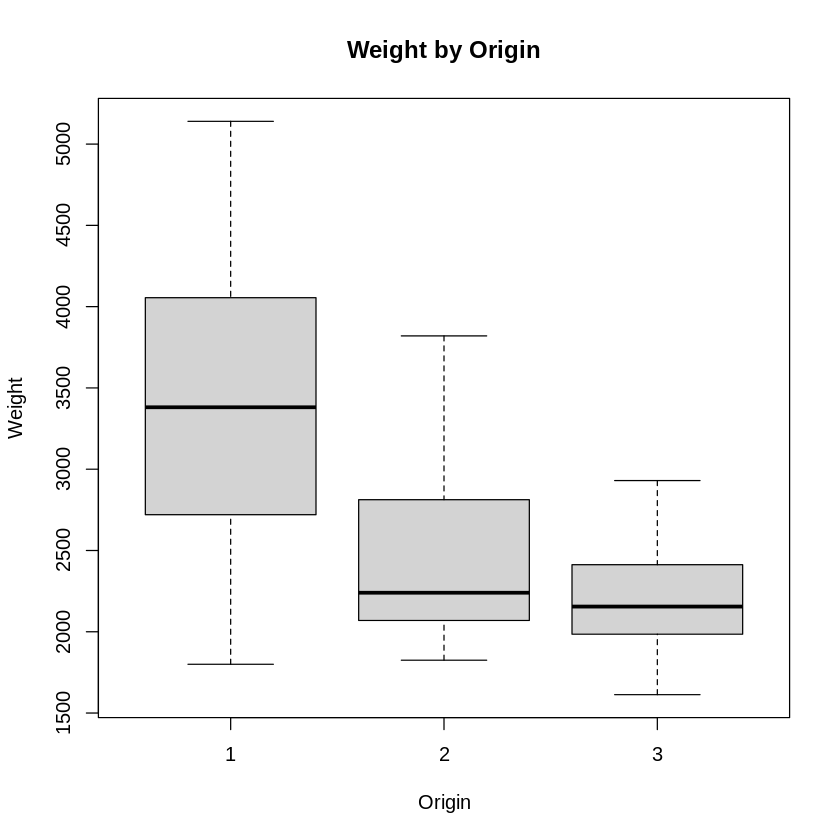
1. **Using the full data set, investigate the predictors graphically, using scatterplots or other tools of your choice. Create some plots highlighting the relationships among the predictors. Comment on your findings**



*Figure 1: Bird’s Eye View*

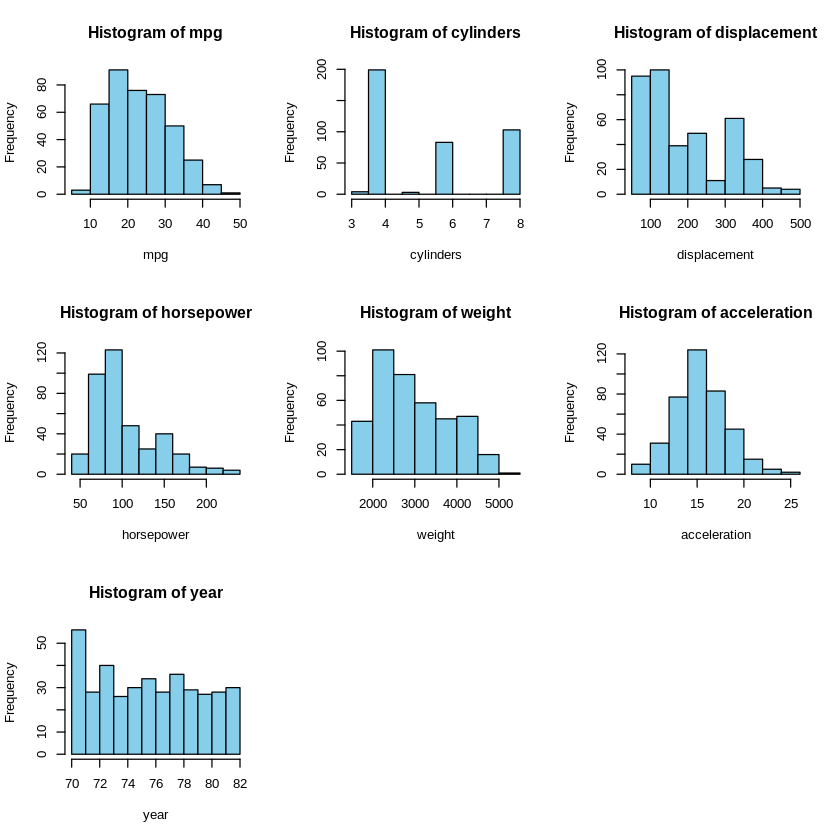


*Figure 2: Correlation Heatmap*

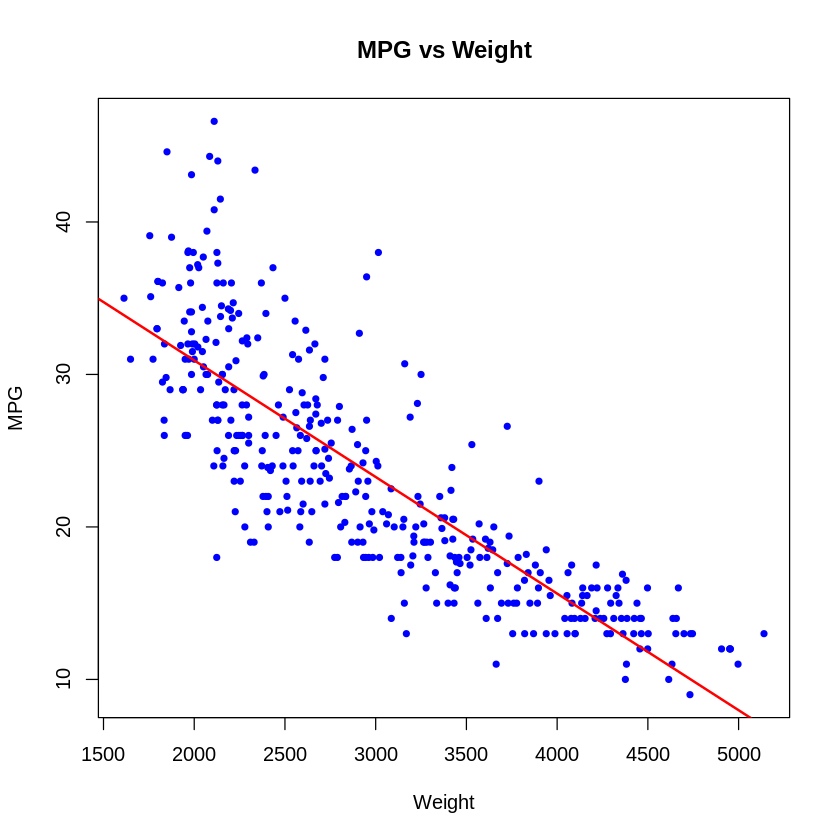
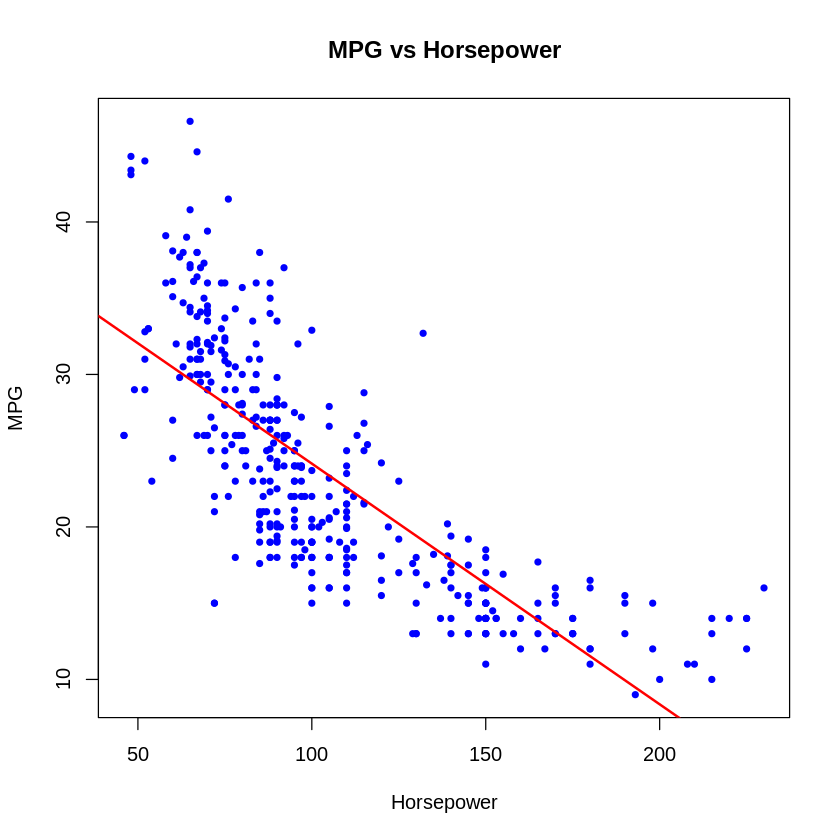


*Figure 4: Boxplot of Weight by Origin of Car*

*Figure 3: Boxplot of MPG by Number of Cylinders*

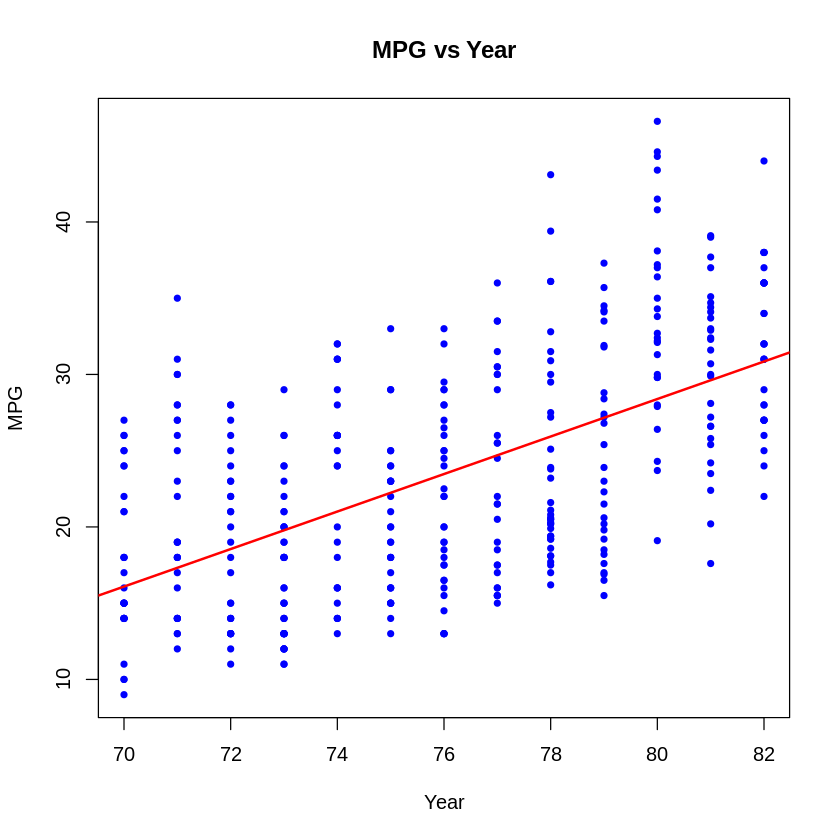
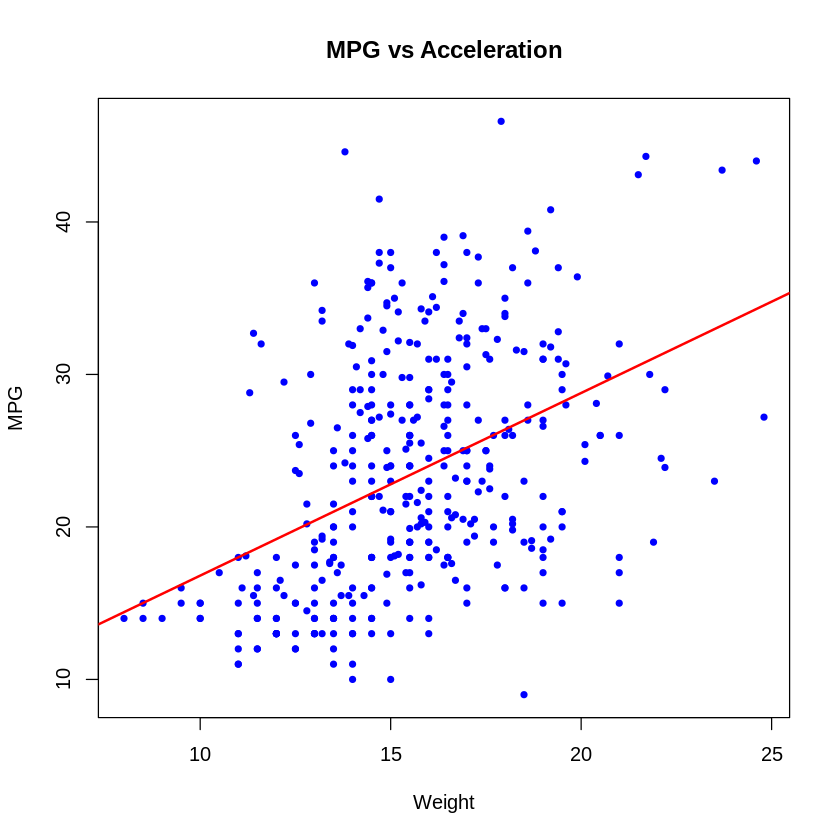


*Figure 5: Histograms of every quantitative variable*

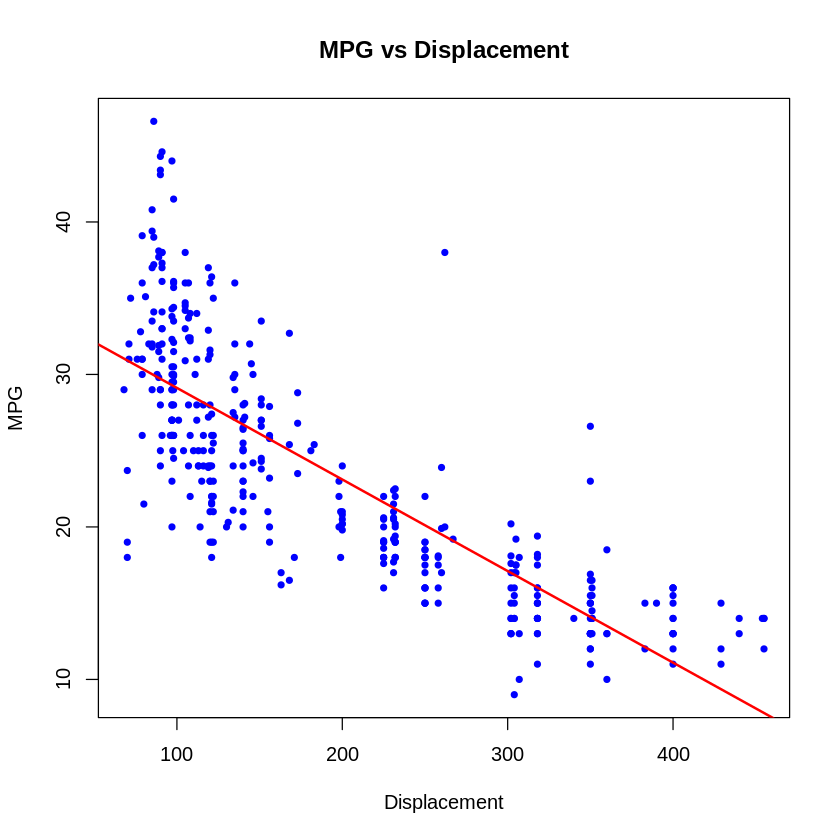
*Figure 7: Scatterplot of MPG vs Weight*

*Figure 6: Scatterplot of MPG vs Horsepower*



*Figure 8: Scatterplot of MPG vs Acceleration*

*Figure 7: Scatterplot of MPG vs Year*



*Figure 10: Scatterplot of Weight vs Year*

*Figure 9: Scatterplot of MPG vs Displacement*

**Observations & Findings**

1. **Correlation Heatmap**Weight, Horsepower, Displacement, Cylinders show negative correlation with MPG.  
   Acceleration and Year show positive correlation with MPG.
2. **Histograms  
   MPG**: Most cars fall between 15–30 mpg. Few cars have very high mpg (>35).  
   **Cylinders:** Cars mostly have 4, 6, or 8 cylinders, with 4 being most common.  
   **Displacement**: Many small engines, but also some very large ones → right-skewed.  
   **Horsepower**: Peaks at 70–100 hp, but a few cars go well above 150 hp.  
   **Weight**: Most cars weigh 2,000–4,000 lbs; heavier cars are less frequent.  
   **Acceleration**: Cantered around 15–20 sec; fairly evenly spread.  
   **Year**: Uniform spread from 1970–1982, as expected (one value per year).
3. **Box Plots  
   Figure 3:** Cars with more cylinders (6, 8) have lower mpg. Cars with 4 cylinders have much higher mpg. **Figure 4**: Cars with origin = 1 are heaviest
4. **Scatter Plots  
   Horsepower vs MPG**: Cars with higher horsepower generally have worse mpg.  
   **Weight vs MPG**: Clear negative slope, heavier cars have lower mpg.  
   **Displacement vs MPG**: Bigger engines (displacement) give lower mpg.  
   **Year vs MPG**: Strong positive trend, newer cars (late 70s/early 80s) were more fuel-efficient.  
   **Year vs Weight**: Car weight decreases over time, supporting the above.  
   **Acceleration vs MPG**: No strong pattern, weak positive correlation.
5. **Suppose that we wish to predict gas mileage (mpg) on the basis of the other variables. Do your plots suggest that any of the other variables might be useful in predicting mpg? Justify your answer.**Yes, the plots suggest that several predictors are likely to be useful in predicting mpg. **Weight, horsepower, and displacement** show strong negative associations with mpg. Heavier cars with larger engines and more horsepower tend to have lower fuel efficiency. These variables are therefore strong candidates for predicting mpg. **Cylinders** also provide predictive power: cars with more cylinders (6, 8) consistently achieve lower mpg compared to 4-cylinder cars, highlighting a categorical effect.

**R Codes (Text)**# Install and load required packages

install.packages("corrplot")

library(ggplot2)

library(corrplot)

# Load data

auto\_data <- read.table("/content/Auto.data", header = TRUE, na.strings = "?")

head(auto\_data)

# Explore unique values and missing values

unique(auto\_data$year)

unique(auto\_data$origin)

sum(is.na(auto\_data))

# Remove missing values

auto\_data\_cleaned <- na.omit(auto\_data)

auto\_data\_cleaned[84, ]

sum(is.na(auto\_data\_cleaned))

# Check range for quantitative predictors

sapply(auto\_data\_cleaned[c("mpg","cylinders","displacement","horsepower",

"weight","acceleration","year")], range)

quant\_variables <- auto\_data\_cleaned[c("mpg","cylinders","displacement",

"horsepower","weight","acceleration","year")]

summary\_table <- data.frame(

Mean = sapply(quant\_variables, mean),

Standard\_Deviation = sapply(quant\_variables, sd)

)

summary\_table <- round(summary\_table, 3)

summary\_table

auto\_data\_trimmed <- auto\_data\_cleaned[-(10:84), ]

quant\_variables\_trimmed <- auto\_data\_trimmed[c("mpg","cylinders","displacement",

"horsepower","weight","acceleration","year")]

summary\_table <- data.frame(

Mean = round(sapply(quant\_variables\_trimmed, mean), 3),

SD = round(sapply(quant\_variables\_trimmed, sd), 3),

Min = round(sapply(quant\_variables\_trimmed, min), 3),

Max = round(sapply(quant\_variables\_trimmed, max), 3)

)

summary\_table$Range <- paste0(summary\_table$Min, " – ", summary\_table$Max)

summary\_table <- summary\_table[, c("Mean", "SD", "Range")]

summary\_table

quant\_vars <- auto\_data\_cleaned[, c("mpg","cylinders","displacement",

"horsepower","weight","acceleration","year")]

pairs(quant\_vars, main = "Scatterplot Matrix of Quantitative Variables")

cor\_matrix <- cor(quant\_vars)

corrplot(cor\_matrix, method = "color", type = "full", tl.col = "black", tl.srt = 45)

boxplot(mpg ~ cylinders, data = auto\_data\_cleaned,

xlab = "Cylinders", ylab = "MPG", main = "MPG by Cylinders")

boxplot(weight ~ origin, data = auto\_data\_cleaned,

xlab = "Origin", ylab = "Weight", main = "Weight by Origin")

quant\_vars <- c("mpg","cylinders","displacement","horsepower","weight","acceleration","year")

par(mfrow = c(3, 3)) # 3x3 layout

for (var in quant\_vars) {

hist(auto\_data\_cleaned[[var]],

main = paste("Histogram of", var),

xlab = var,

col = "skyblue")

}

predictors <- c("weight","horsepower","acceleration","year","displacement")

for (var in predictors) {

plot(auto\_data\_cleaned[[var]], auto\_data\_cleaned$mpg,

main = paste("MPG vs", var),

xlab = var,

ylab = "MPG",

col = "blue", pch = 20)

abline(lm(auto\_data\_cleaned$mpg ~ auto\_data\_cleaned[[var]]),

col = "red", lwd = 2)

}

# Separate plot for Year vs Weight

plot(auto\_data\_cleaned$year, auto\_data\_cleaned$weight,

main = "Year vs Weight",

xlab = "Year", ylab = "Weight",

col = "blue", pch = 20)

abline(lm(weight ~ year, data = auto\_data\_cleaned),

col = "red", lwd = 2)