Assignment2 Memo

Dr. Niladri Chakraborty

2024-03-25

Q1. Suppose a manufacturing plant produces 3 different products, and it has limited resources available for production. It has three different production units. Let X_1, X_2, X_3 be the number of machines to run for each product that are produced. The weekly cost of running these machines (in R10000) in three different units can be framed as a system of linear equations:

$$2X_1 - X_2 + 3X_3 = 4$$

$$3X_1 + 2X_2 - 5X_3 = 5$$

$$X_1 + 4X_2 + 2X_3 = 3$$

#Solve these equations with the help of R functions to find the number of machines to run for each product.

```
A <- matrix(c(2, -1, 3, 3, 2, -5, 1, 4, 2), nrow = 3, byrow = TRUE)
B <- c(4, 5, 3)
X <- solve(A, B)
X

## [1] 1.8314607 0.2022472 0.1797753
```

After solving the system of linear equations, we find that the number of machines to run for each product are approximately

Product 1: 1.83

Product 2: 0.20

Product 3: 0.18

Q2. Draw random samples of size 1000, 100 000, 50 00 000, respectively, from uniform(0,1) distribution. Save them into three different variables, say, x1, x2, x3. You do not need to print the random samples. Then calculate the mean of x1, x2, x3. Which mean is the closest to 0.5. Explain your findings.

```r
# Set seed for reproducibility
set.seed(123)

# Draw random samples from a uniform(0,1) distribution
x1 <- runif(1000)
x2 <- runif(100000)
x3 <- runif(50000000)

# Calculate the mean of each sample
mean_x1 <- mean(x1)
mean_x2 <- mean(x2)
mean_x3 <- mean(x3)

# Print the means
print(mean_x1)

## [1] 0.4972778

print(mean_x2)

## [1] 0.4993165

print(mean_x3)

## [1] 0.4999909

# Determine which mean is closest to 0.5
means <- c(mean_x1, mean_x2, mean_x3)
names <- c("x1", "x2", "x3")
names[which.min(abs(means - 0.5))]

## [1] "x3"
```

as the sample size increases, the sample mean should get closer to the expected value (mean) of the population. In this case, the population mean of a uniform(0,1) distribution is 0.5. Therefore, we expect larger sample sizes to produce means closer to 0.5, as observed.

Q3. Get a dataset from a package in R. Analyse the data and prepare a report.

(The data analysis report provided here serves merely as a template and should not be taken as a strict guideline. Feel free to create your own report based on this example or your unique approach to statistical data analysis.)

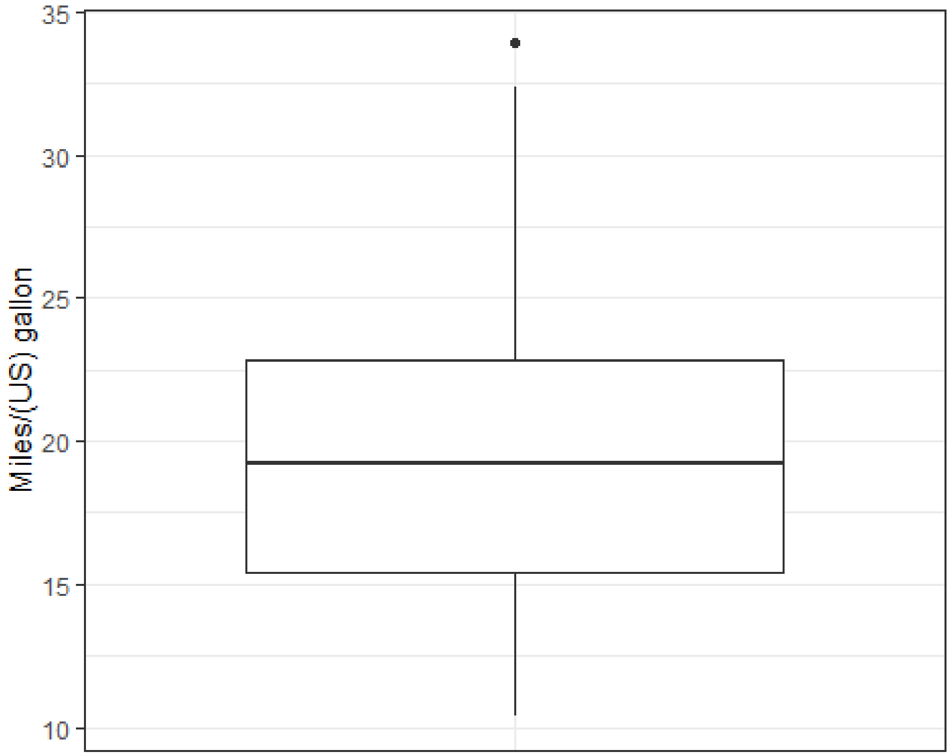
Let's use the mtcars dataset from R's built-in datasets, a commonly used dataset. The mtcars dataset comprises fuel consumption and ten aspects of automobile design and performance for 32 automobiles.

1. Preliminary Data Check and Cleaning

```
library(ggplot2)
library(GGally)

## Warning: package 'GGally' was built under R version 4.3.1

## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2

# Load the dataset
data("mtcars")

# Check for missing values
any(is.na(mtcars))

## [1] FALSE
```

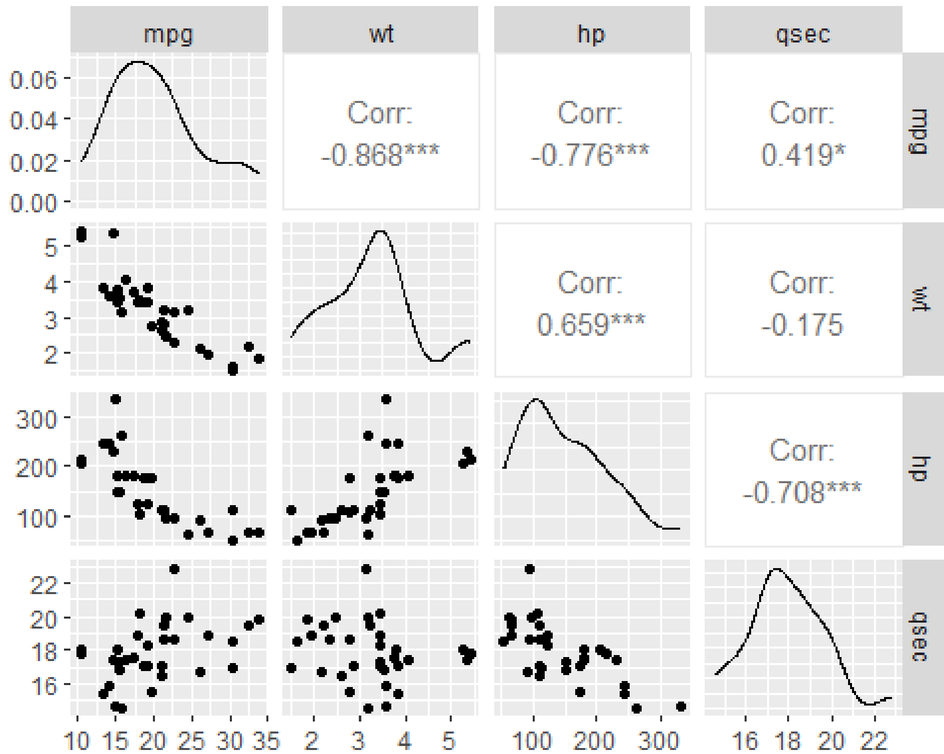
As we find, there is no missing value in the dataset. Now, we can use ggplot2 to create boxplots for the mpg (miles per gallon) variable as an example to check for outliers:

```
ggplot(mtcars, aes(x = factor(0), y = mpg)) +
  geom_boxplot() +
  labs(x = "", y = "Miles/(US) gallon") +
  theme_bw() +
  theme(axis.title.x=element_blank(),
        axis.ticks.x=element_blank(),
        axis.text.x=element_blank())
```


35
30
25
20
15
10
Miles/(US) gallon

Analyzing correlations:

```
# Correlation matrix plot for selected variables
ggpairs(mtcars[, c("mpg", "wt", "hp", "qsec")])
```

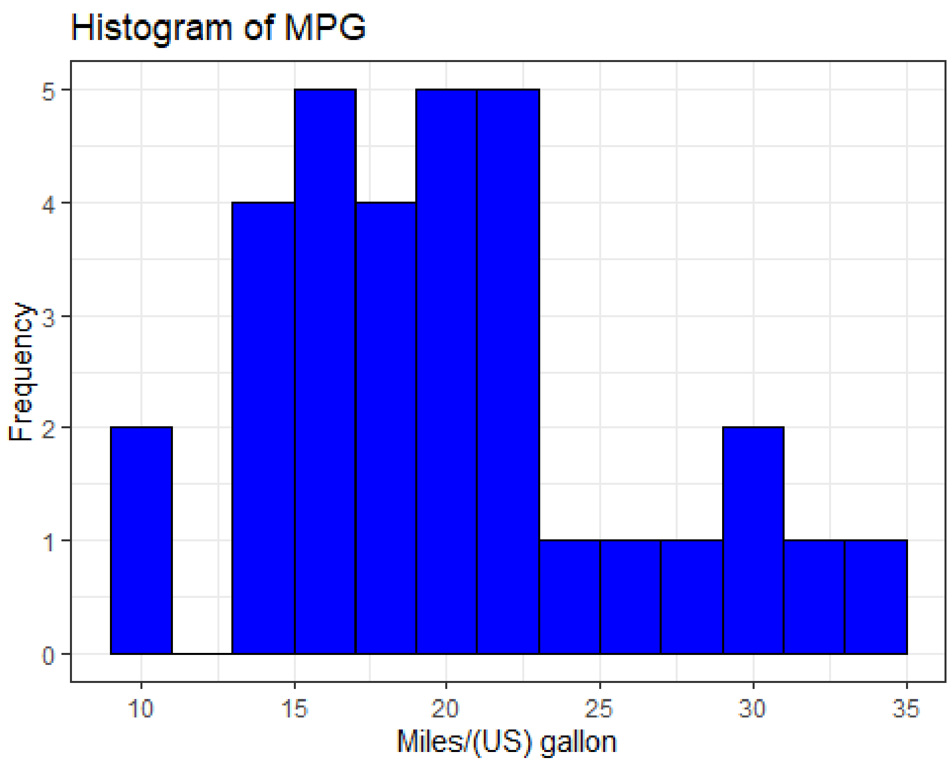


The correlation plot shows that there are significant correlations among all the variables considered.

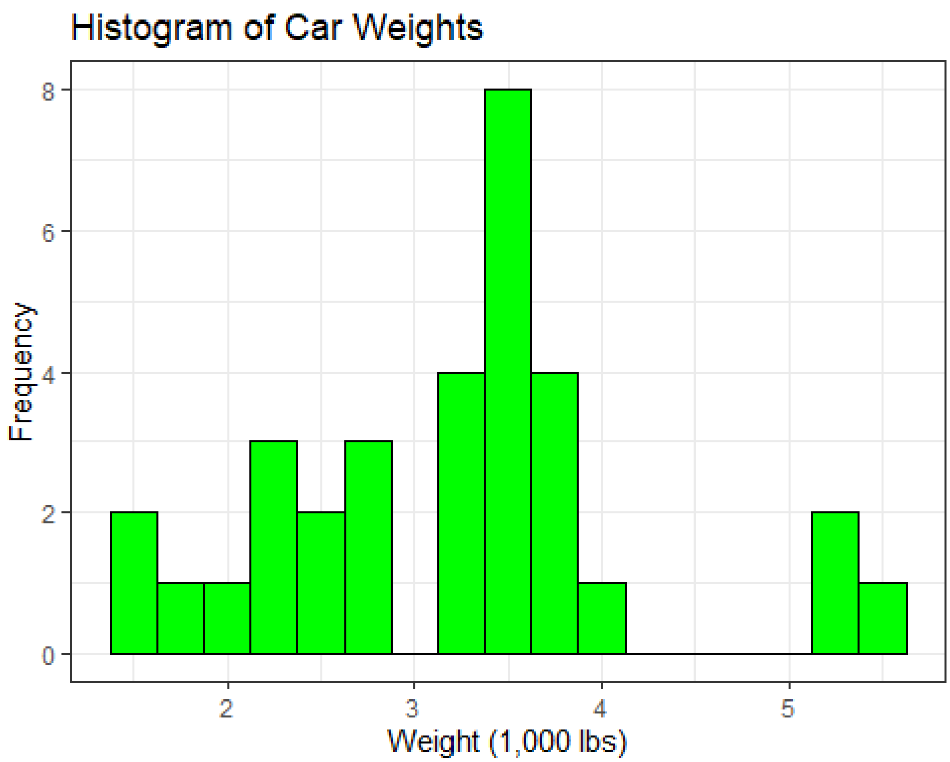
Creating Histograms:

Histograms can give insights into the distribution of variables.

```
# Histogram for mpg
ggplot(mtcars, aes(x = mpg)) +
  geom_histogram(binwidth = 2, fill = "blue", color = "black") +
  labs(title = "Histogram of MPG", x = "Miles/(US) gallon", y = "Frequency")
+
  theme_bw()
```




```r
# Histogram for wt
ggplot(mtcars, aes(x = wt)) +
  geom_histogram(binwidth = 0.25, fill = "green", color = "black") +
  labs(title = "Histogram of Car Weights", x = "Weight (1,000 lbs)", y =
"Frequency") +
  theme_bw()
```



MPG (Miles Per Gallon): The histogram of mpg shows a distribution of fuel efficiencies across the dataset. As the distribution appears right-skewed/positively skewed, it may indicate that most cars fall within a particular range of fuel efficiency, with fewer cars achieving very high mpg.

WT (Weight): The histogram for car weights (wt) shows how the weights of cars are distributed. An approximately symmetric distribution suggests that most cars in the dataset are around 3500 pound (lbs), highlighting the concentration of car designs around this specific weight configuration.