Homework 3

Han Nguyen - TXN200004

09/30/2025

Problem 1

```r
# Read the data
mobile_data <- read.csv("train.csv")
```

a)

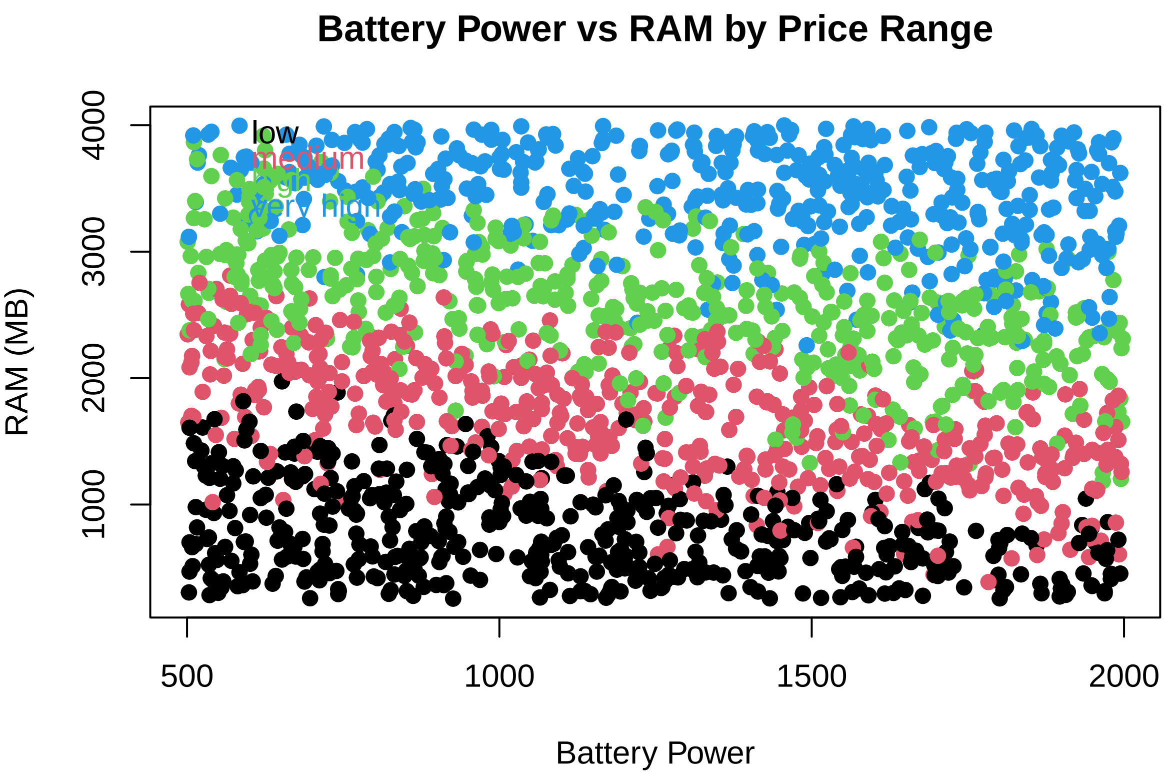
```r
# Convert price_range to a factor with proper labels
mobile_data$price_range <- factor(mobile_data$price_range,
                                   levels = c(0, 1, 2, 3),
                                   labels = c("low", "medium", "high", "very high"),
                                   ordered = TRUE)
head(mobile_data$price_range)
```

```
## [1] medium high   high   high   medium medium
## Levels: low < medium < high < very high
```

The variable price_range has been converted to a factor with levels: “low”, “medium”, “high”, and “very high”.

b)

```r
# Create scatter plot with colors based on price range
plot(mobile_data$battery_power, mobile_data$ram,
     col = as.numeric(mobile_data$price_range),
     pch = 19,
     xlab = "Battery Power",
     ylab = "RAM (MB)",
     main = "Battery Power vs RAM by Price Range")
text(x = par("usr")[1] + 0.1 * diff(par("usr")[1:2]),
     y = par("usr")[4] - 0.05 * diff(par("usr")[3:4]) * (1:4),
     labels = levels(mobile_data$price_range),
     col = 1:4,
     adj = 0)
```



c)

```r
# Calculate Pearson correlation between ram and battery_power
cor_overall <- cor(mobile_data$ram, mobile_data$battery_power)
```

The Pearson correlation between RAM and battery power is r $= -7 \times 10^{-4}$.

d)

```r
# Create four separate datasets by price range
priceLow <- subset(mobile_data, price_range == "low")
priceMedium <- subset(mobile_data, price_range == "medium")
priceHigh <- subset(mobile_data, price_range == "high")
priceVeryhigh <- subset(mobile_data, price_range == "very high")
```

Four separate datasets have been created: priceLow, priceMedium, priceHigh, and priceVeryhigh.

e)

```r
# Calculate correlations for each price range
cor_low <- cor(priceLow$ram, priceLow$battery_power)
cor_medium <- cor(priceMedium$ram, priceMedium$battery_power)
cor_high <- cor(priceHigh$ram, priceHigh$battery_power)
cor_veryhigh <- cor(priceVeryhigh$ram, priceVeryhigh$battery_power)
```

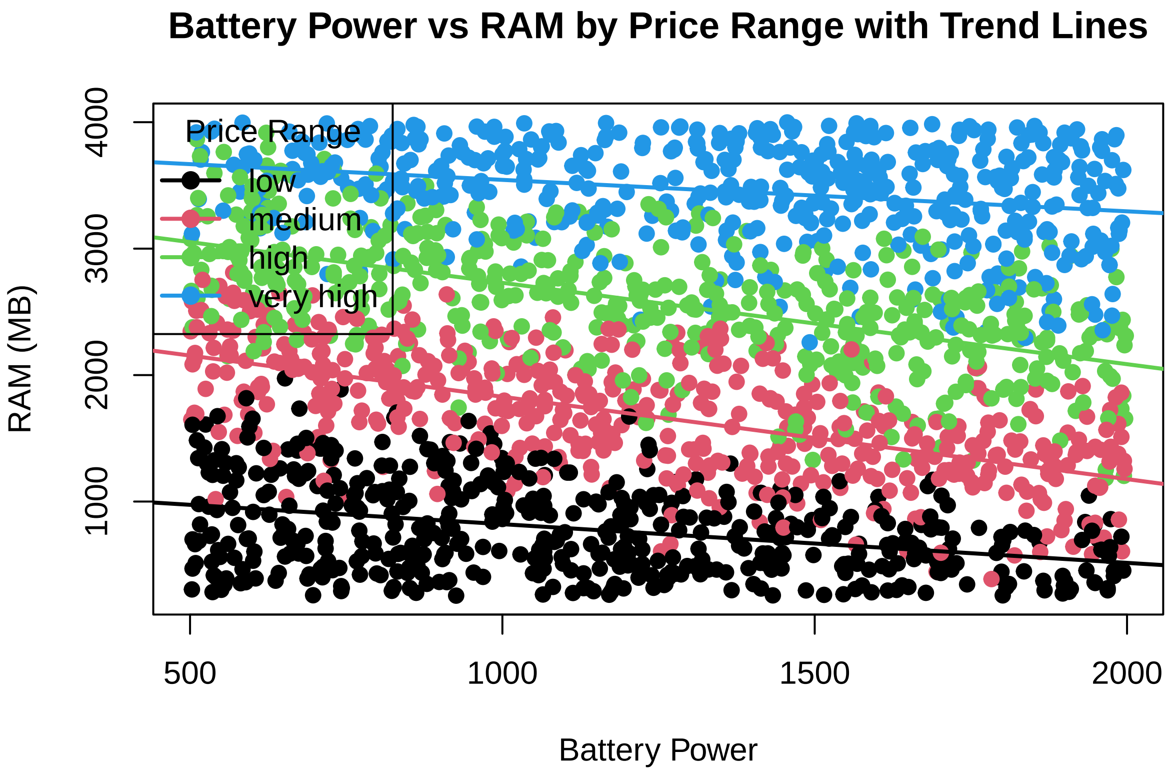
Correlations by price range:

- Low: r = -0.3466
- Medium: r = -0.6134
- High: r = -0.5874
- Very High: r = -0.2628

Explanation: The correlations within each price range are much weaker (near zero) compared to the overall correlation from part (c). This is an example of Simpson's Paradox. When we look at all data together, we see a positive correlation because higher-priced phones tend to have both more RAM and larger batteries. However, within each price category, there is little to no relationship between RAM and battery power, as manufacturers balance various features independently. The strong overall correlation was actually driven by the confounding variable (price range) rather than a true relationship between RAM and battery power.

f)

```r
# Recreate scatter plot with trend lines for each price range
plot(mobile_data$battery_power, mobile_data$ram,
     col = as.numeric(mobile_data$price_range),
     pch = 19,
     xlab = "Battery Power",
     ylab = "RAM (MB)",
     main = "Battery Power vs RAM by Price Range with Trend Lines")

# Add trend lines for each price range
abline(lm(ram ~ battery_power, data = priceLow), col = 1, lwd = 2)
abline(lm(ram ~ battery_power, data = priceMedium), col = 2, lwd = 2)
abline(lm(ram ~ battery_power, data = priceHigh), col = 3, lwd = 2)
abline(lm(ram ~ battery_power, data = priceVeryhigh), col = 4, lwd = 2)

legend("topleft", legend = levels(mobile_data$price_range),
       col = 1:4, pch = 19, lwd = 2, title = "Price Range")
```



g)

```r
# Calculate average and median clock speed for phones with 4, 6, and 8 cores
cores_4 <- subset(mobile_data, n_cores == 4)
cores_6 <- subset(mobile_data, n_cores == 6)
cores_8 <- subset(mobile_data, n_cores == 8)

avg_4 <- round(mean(cores_4$clock_speed), 2)
med_4 <- round(median(cores_4$clock_speed), 2)

avg_6 <- round(mean(cores_6$clock_speed), 2)
med_6 <- round(median(cores_6$clock_speed), 2)

avg_8 <- round(mean(cores_8$clock_speed), 2)
med_8 <- round(median(cores_8$clock_speed), 2)
```

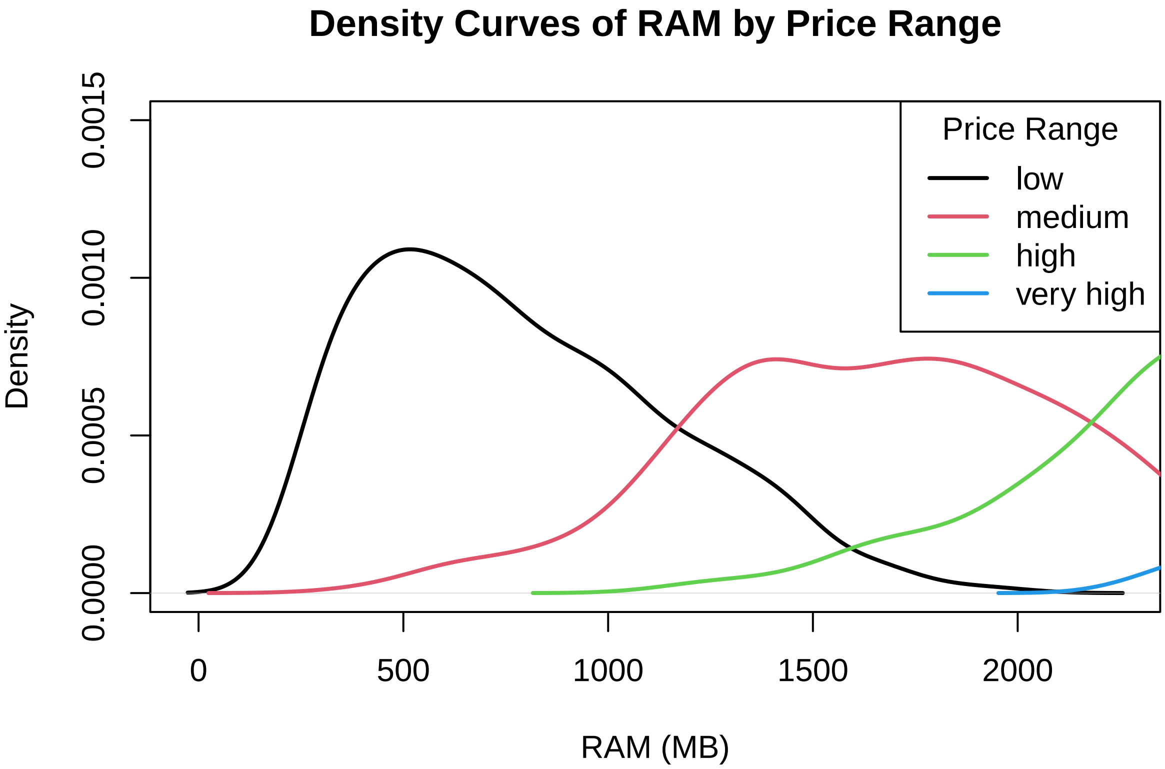
Clock Speed Statistics:

- 4 cores: Average = 1.55, Median = 1.5
- 6 cores: Average = 1.53, Median = 1.5
- 8 cores: Average = 1.51, Median = 1.4

Explanation: The average and median clock speeds remain essentially the same across different core counts because clock speed and number of cores appear to be independent features in this dataset. This suggests that the data was likely generated with these variables distributed independently, or that manufacturers don't systematically pair higher core counts with different clock speeds in this sample.

h)

```
# Create density curves for RAM by price range
plot(density(priceLow$ram), col = 1, lwd = 2,
     main = "Density Curves of RAM by Price Range",
     xlab = "RAM (MB)", ylim = c(0, 0.0015))
lines(density(priceMedium$ram), col = 2, lwd = 2)
lines(density(priceHigh$ram), col = 3, lwd = 2)
lines(density(priceVeryhigh$ram), col = 4, lwd = 2)
legend("topright", legend = levels(mobile_data$price_range),
       col = 1:4, lwd = 2, title = "Price Range")
```

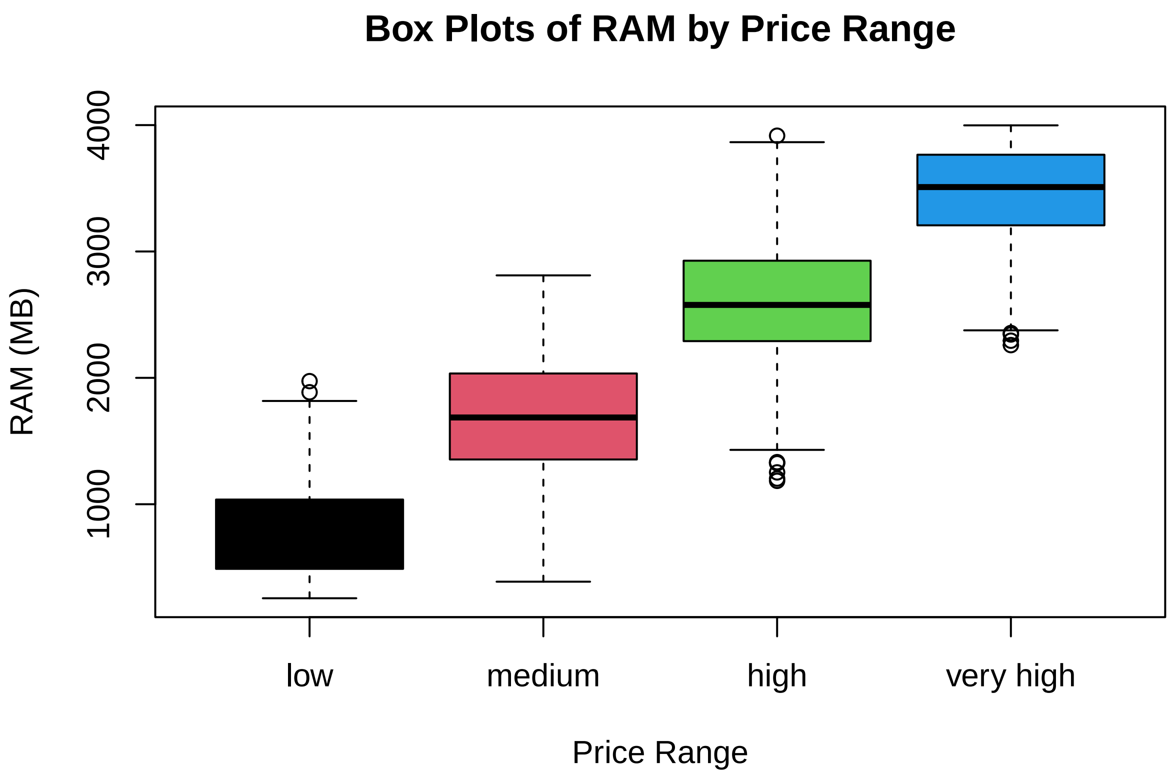


Shape descriptions:

- Low: Roughly uniform or slightly right-skewed distribution centered at lower RAM values
- Medium: More concentrated distribution in the mid-range RAM values
- High: Distribution shifts toward higher RAM values
- Very High: Distribution concentrated at the highest RAM values, left-skewed

i)

```
# Create box plots for RAM by price range
boxplot(ram ~ price_range, data = mobile_data,
        col = 1:4,
        xlab = "Price Range",
        ylab = "RAM (MB)",
        main = "Box Plots of RAM by Price Range")
```



Shape descriptions:

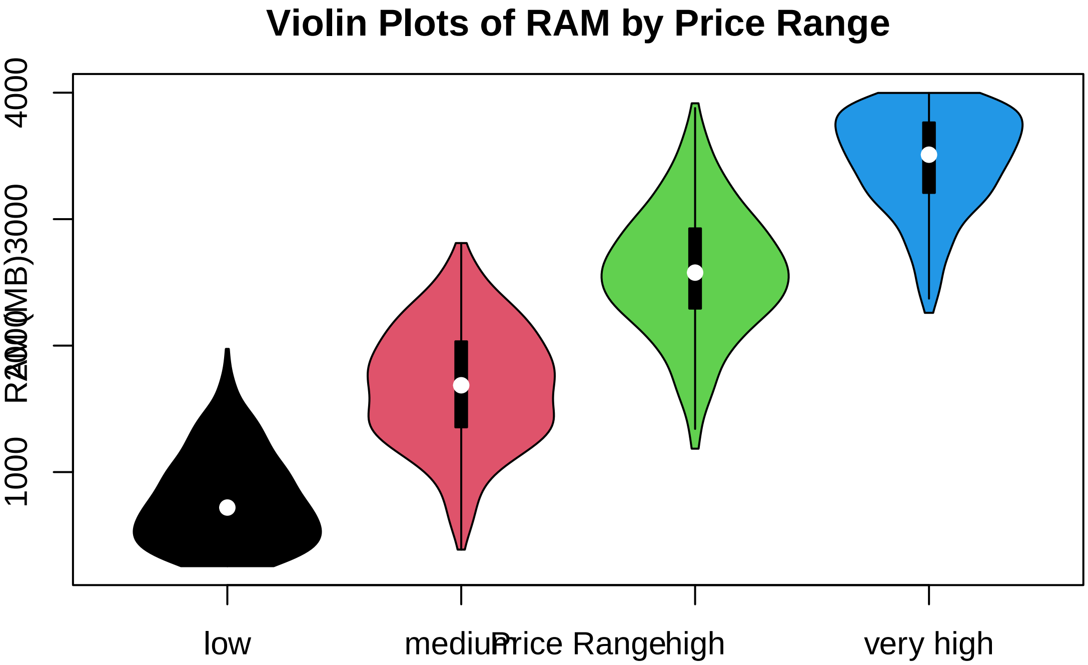
- Low: Median around 1000-1500 MB, relatively symmetric with some outliers
- Medium: Median around 1500-2000 MB, relatively symmetric
- High: Median around 2500-3000 MB, relatively symmetric
- Very High: Median around 3000-3500 MB, relatively symmetric
- All price ranges show approximately symmetric distributions with increasing median RAM as price increases

j)

```
# Create violin plots using base R vioplot package
library(vioplot)
```

```
## Loading required package: sm

## Package 'sm', version 2.2-6.0: type help(sm) for summary information

## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```r
vioplot(priceLow$ram, priceMedium$ram, priceHigh$ram, priceVeryhigh$ram,
        names = levels(mobile_data$price_range),
        col = 1:4,
        xlab = "Price Range",
        ylab = "RAM (MB)",
        main = "Violin Plots of RAM by Price Range")
```



Shape descriptions:

The violin plots confirm the findings from the density curves and box plots:

- Low: Distribution is fairly uniform/flat across lower RAM values
- Medium: More concentrated, roughly uniform distribution in mid-range
- High: Similar uniform pattern but shifted to higher values
- Very High: Concentrated at highest RAM values with more density at the upper end
- Clear separation and upward shift in RAM distributions as price range increases

````
# Problem 2

‘‘‘ r
# Install and load the package
# install.packages("ggplot2")
library(ggplot2)

# Load the mpg dataset
data("mpg")
````

a)

```r
# Turn cyl to an ordered factor variable
mpg$cyl <- factor(mpg$cyl, levels = c("4", "5", "6", "8"), ordered = TRUE)
```

The variable cyl has been converted to an ordered factor with levels “4”, “5”, “6”, and “8”.

b)

```r
# Extract first 4 characters and convert trans to factor with "auto" and "manu"
mpg$trans <- substr(mpg$trans, 1, 4)
mpg$trans <- factor(mpg$trans, levels = c("auto", "manu"))
```

The variable trans has been converted to a factor variable with levels “auto” and “manu”.

c)

```r
# Turn drv to an ordered factor variable
mpg$drv <- factor(mpg$drv, levels = c("f", "r", "4"), ordered = TRUE)
```

The variable drv has been converted to an ordered factor with levels “f”, “r”, and “4”.

d)

```r
# Turn fl to a factor variable with "gasoline", "diesel", and "other"
mpg$fl <- ifelse(mpg$fl == "d", "diesel",
                 ifelse(mpg$fl %in% c("p", "r"), "gasoline", "other"))
mpg$fl <- factor(mpg$fl, levels = c("diesel", "gasoline", "other"))
```

The variable fl has been converted to a factor variable with levels “diesel”, “gasoline”, and “other”.

e)

```r
# Turn class to an ordered factor variable
mpg$class <- factor(mpg$class,
                    levels = c("2seater", "subcompact", "compact", "midsize",
                               "suv", "minivan", "pickup"),
                    ordered = TRUE)
```

The variable class has been converted to an ordered factor with the specified levels.

f)

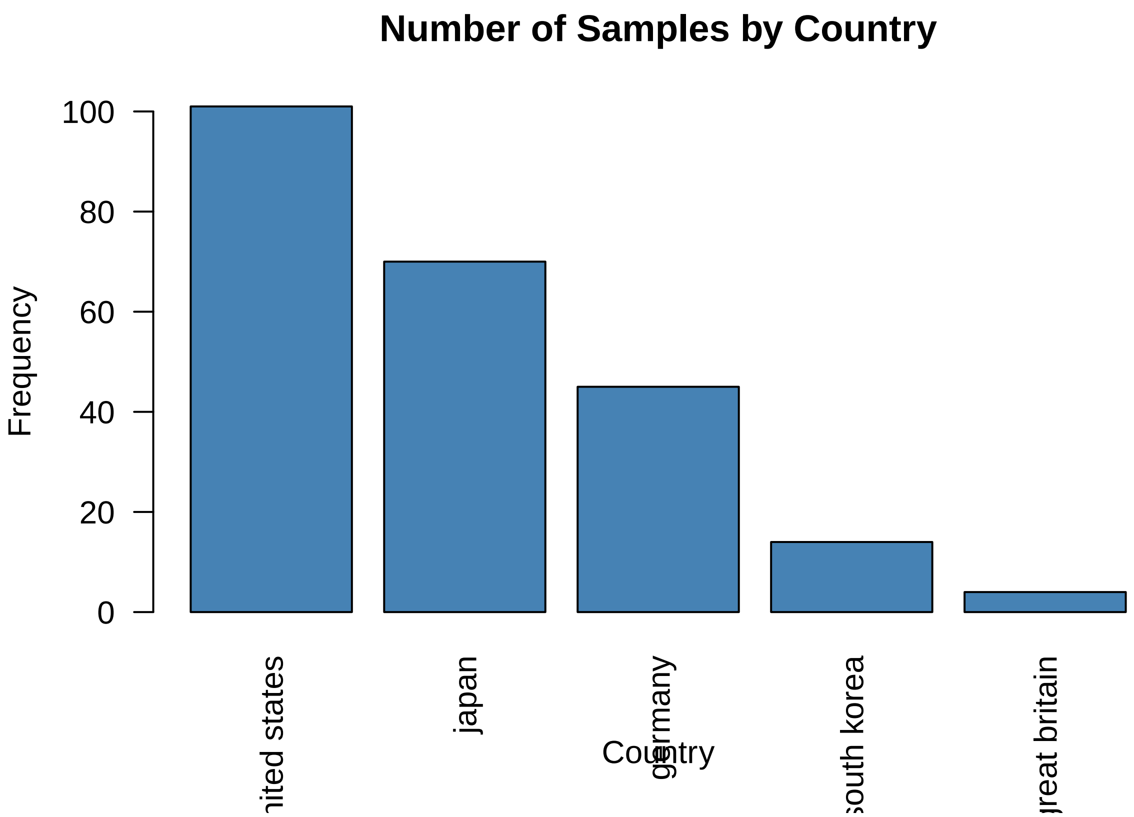
```
# Create country variable based on manufacturer
mpg$country <- ifelse(mpg$manufacturer %in% c("chevrolet", "dodge", "ford",
                                              "jeep", "lincoln", "mercury",
                                              "pontiac"), "united states",
                      ifelse(mpg$manufacturer %in% c("honda", "nissan",
                                                     "subaru", "toyota"), "japan",
                             ifelse(mpg$manufacturer %in% c("audi",
                                                            "volkswagen"), "germany",
                                    ifelse(mpg$manufacturer == "hyundai", "south korea",
                                           ifelse(mpg$manufacturer == "land rover",
                                                  "great britain", NA)))))

# Check the structure
str(mpg)
```

```
## tibble [234 x 12] (S3: tbl_df/tbl/data.frame)
##  $ manufacturer: chr [1:234] "audi" "audi" "audi" "audi" ...
##  $ model       : chr [1:234] "a4" "a4" "a4" "a4" ...
##  $ displ       : num [1:234] 1.8 1.8 2 2 2.8 2.8 3.1 1.8 1.8 2 ...
##  $ year        : int [1:234] 1999 1999 2008 2008 1999 1999 2008 1999 1999 2008 ...
##  $ cyl         : Ord.factor w/ 4 levels "4"<"5"<"6"<"8": 1 1 1 1 3 3 3 1 1 1 ...
##  $ trans       : Factor w/ 2 levels "auto","manu": 1 2 2 1 1 2 1 2 1 2 ...
##  $ drv         : Ord.factor w/ 3 levels "f"<"r"<"4": 1 1 1 1 1 1 1 3 3 3 ...
##  $ cty         : int [1:234] 18 21 20 21 16 18 18 18 16 20 ...
##  $ hwy         : int [1:234] 29 29 31 30 26 26 27 26 25 28 ...
##  $ fl          : Factor w/ 3 levels "diesel","gasoline",..: 2 2 2 2 2 2 2 2 2 2 ...
##  $ class       : Ord.factor w/ 7 levels "2seater"<"subcompact"<..: 3 3 3 3 3 3 3 3 3 3 ...
##  $ country     : chr [1:234] "germany" "germany" "germany" "germany" ...
```

The country variable has been created to indicate the manufacturer base location.

g)

```
# Create a table of country frequencies
country_counts <- sort(table(mpg$country), decreasing = TRUE)

# Draw bar plot with countries in decreasing order
barplot(country_counts,
        main = "Number of Samples by Country",
        xlab = "Country",
        ylab = "Frequency",
        col = "steelblue",
        las = 2)
```



The country with the most samples is united states with 101 samples. The country with the least samples is great britain with 4 sample(s).

h)

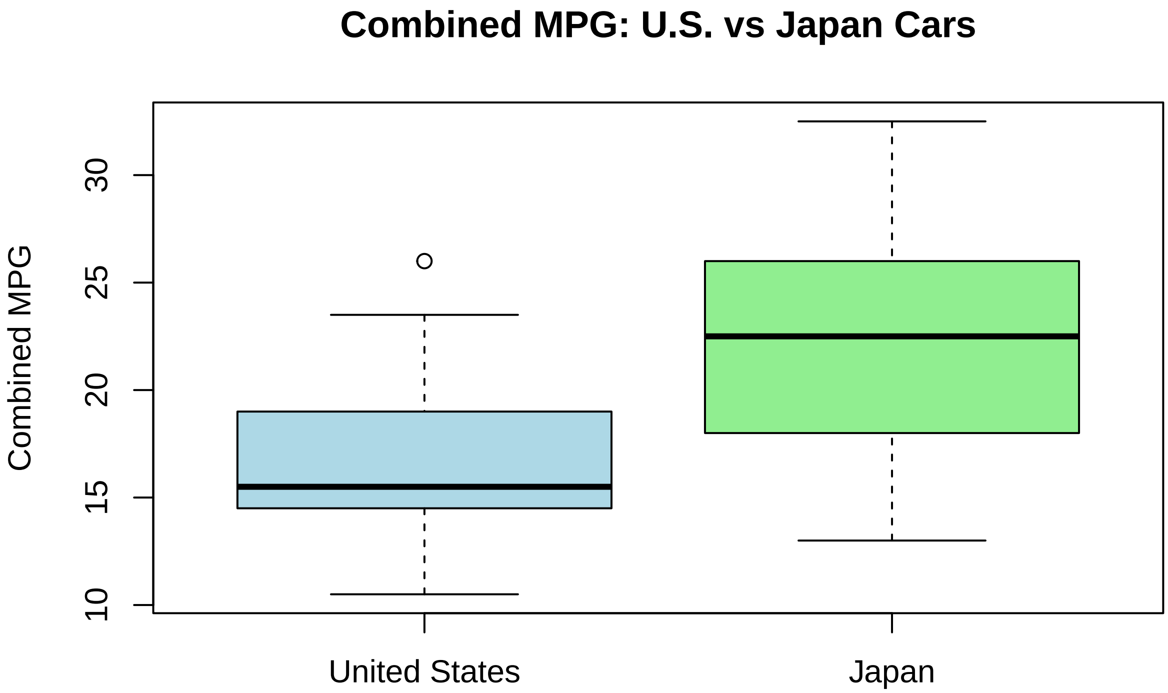
```
# Subset data for U.S. cars
us_cars <- subset(mpg, country == "united states")

# Find mode for each variable using table()
displ_mode <- as.numeric(names(sort(table(us_cars$displ), decreasing = TRUE)[1]))
cyl_mode <- names(sort(table(us_cars$cyl), decreasing = TRUE)[1])
trans_mode <- names(sort(table(us_cars$trans), decreasing = TRUE)[1])
drv_mode <- names(sort(table(us_cars$drv), decreasing = TRUE)[1])
fl_mode <- names(sort(table(us_cars$fl), decreasing = TRUE)[1])
class_mode <- names(sort(table(us_cars$class), decreasing = TRUE)[1])
```

A typical U.S. car has the following characteristics:

- Engine displacement: 4.7 liters
- Number of cylinders: 8
- Transmission type: auto
- Drive type: 4
- Fuel type: gasoline
- Class: suv

i)

```r
# Create combined mpg variable
mpg$combined_mpg <- (mpg$cty + mpg$hwy) / 2

# Subset U.S. and Japan cars
us_cars <- subset(mpg, country == "united states")
japan_cars <- subset(mpg, country == "japan")

# Create boxplot
boxplot(us_cars$combined_mpg, japan_cars$combined_mpg,
        names = c("United States", "Japan"),
        main = "Combined MPG: U.S. vs Japan Cars",
        ylab = "Combined MPG",
        col = c("lightblue", "lightgreen"))
```



```r
# Calculate statistics
us_mean <- round(mean(us_cars$combined_mpg), 2)
us_median <- round(median(us_cars$combined_mpg), 2)
us_sd <- round(sd(us_cars$combined_mpg), 2)
us_iqr <- round(IQR(us_cars$combined_mpg), 2)

japan_mean <- round(mean(japan_cars$combined_mpg), 2)
japan_median <- round(median(japan_cars$combined_mpg), 2)
japan_sd <- round(sd(japan_cars$combined_mpg), 2)
japan_iqr <- round(IQR(japan_cars$combined_mpg), 2)
```

Summary statistics for combined MPG:

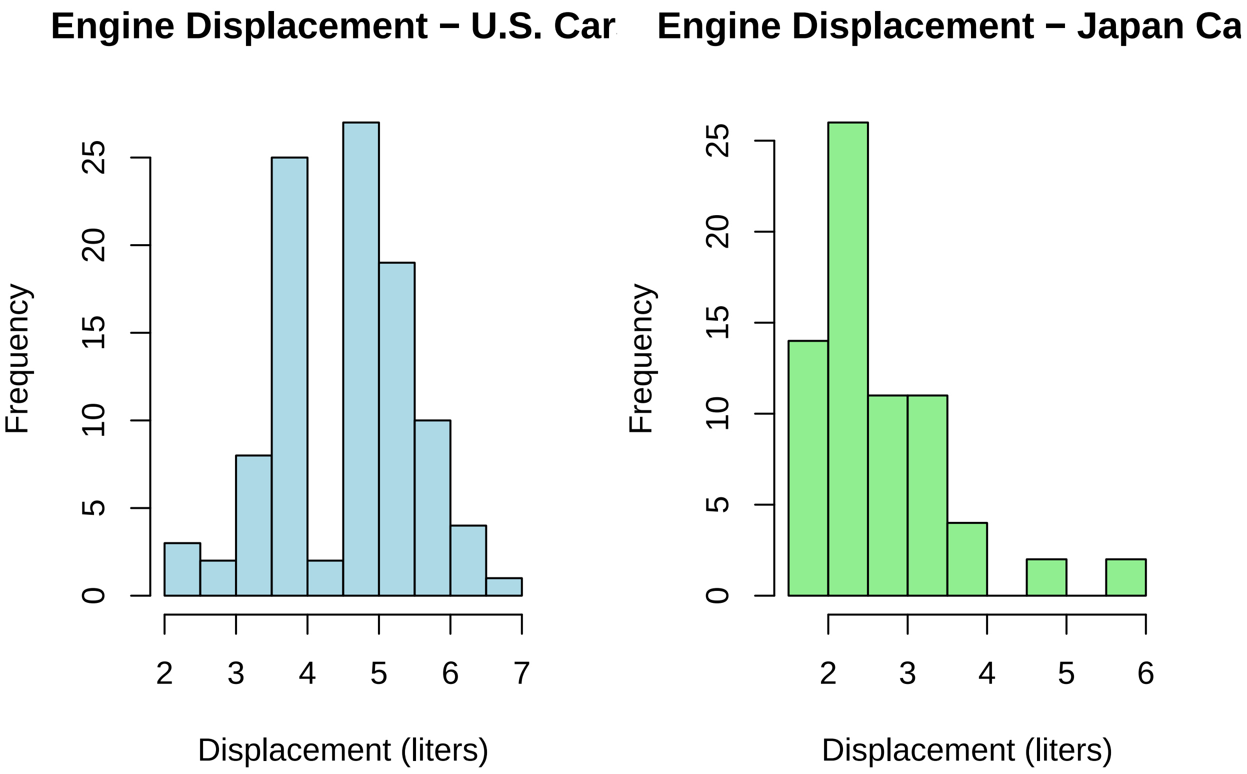
United States: - Mean: 16.64 - Median: 15.5 - Standard Deviation: 3.3 - IQR: 4.5

Japan: - Mean: 22.66 - Median: 22.5 - Standard Deviation: 4.6 - IQR: 7.62

Japanese cars have higher fuel efficiency on average compared to U.S. cars.

j)

```r
# Create histograms for engine displacement
par(mfrow = c(1, 2))

hist(us_cars$displ,
     main = "Engine Displacement - U.S. Cars",
     xlab = "Displacement (liters)",
     col = "lightblue",
     breaks = 10)

hist(japan_cars$displ,
     main = "Engine Displacement - Japan Cars",
     xlab = "Displacement (liters)",
     col = "lightgreen",
     breaks = 10)
```



```r
par(mfrow = c(1, 1))
```

Shape descriptions:

U.S. Cars: The distribution of engine displacement is roughly bimodal or multimodal, with peaks around 3.5-4.0 liters and another concentration around 5.0-6.0 liters. The distribution is somewhat right-skewed, showing that U.S. manufacturers tend to produce cars with larger engines.

Japan Cars: The distribution of engine displacement is right-skewed with the majority of values concentrated in the 1.5-2.5 liter range. There is a long tail extending toward larger engine sizes, but most Japanese cars in this dataset have smaller, more fuel-efficient engines. “‘