Homework 3

Han Nguyen - TXN200004

09/30/2025

Problem 1

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

a)

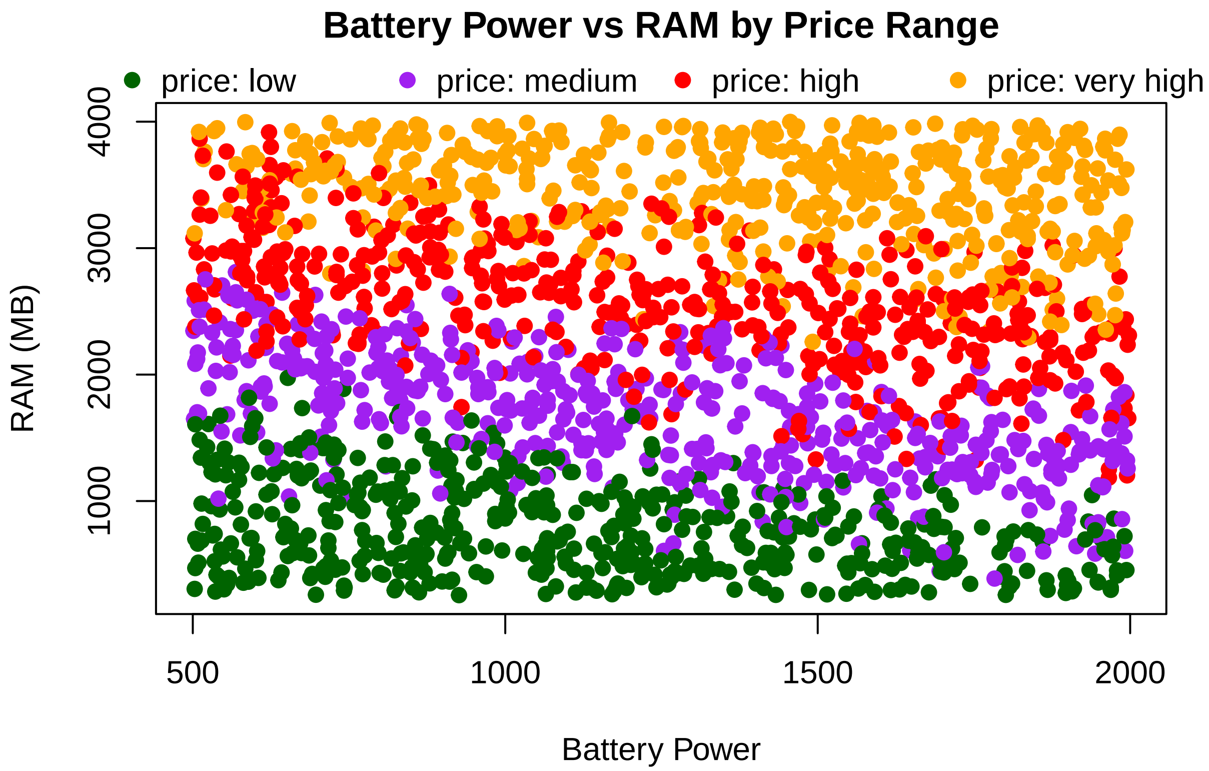
```
# Convert price_range to a factor with proper labels
mobile_data$price_range <- factor(mobile_data$price_range,
                                  levels = c(0, 1, 2, 3),
                                  labels = c("low", "medium", "high", "very high"),
                                  ordered =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

```
plot(mobile_data$battery_power, mobile_data$ram,
     col = ifelse(mobile_data$price_range == "low", "darkgreen",
                  ifelse(mobile_data$price_range == "medium", "purple",
                         ifelse(mobile_data$price_range == "high", "red", "orange"))),
     pch = 19,
     xlab = "Battery Power",
     ylab = "RAM (MB)",
     main = "Battery Power vs RAM by Price Range")

# price_range legends
legend("top", inset = c(0, -0.12),
       legend = paste("price:", levels(mobile_data$price_range)),
       col = c("darkgreen", "purple", "red", "orange"),
       pch = 19,
       horiz = TRUE,
       bty = "n",
       xpd = NA)
```



c)

```
cor_overall <- cor(mobile_data$ram, mobile_data$battery_power)
```

The Pearson correlation between RAM and battery power is r = $-6.5292645 \times 10^{-4}$.

d)

```
# Create four separate datasets by price range
low_set <- subset(mobile_data, price_range == "low")
med_set <- subset(mobile_data, price_range == "medium")
high_set <- subset(mobile_data, price_range == "high")
very_high_set <- subset(mobile_data, price_range == "very high")
```

e)

```
# Correlations for each price range
cor_low <- cor(low_set$ram, low_set$battery_power)
cor_medium <- cor(med_set$ram, med_set$battery_power)
cor_high <- cor(high_set$ram, high_set$battery_power)
cor_very_high <- cor(very_high_set$ram, very_high_set$battery_power)
```

Correlations by price range:

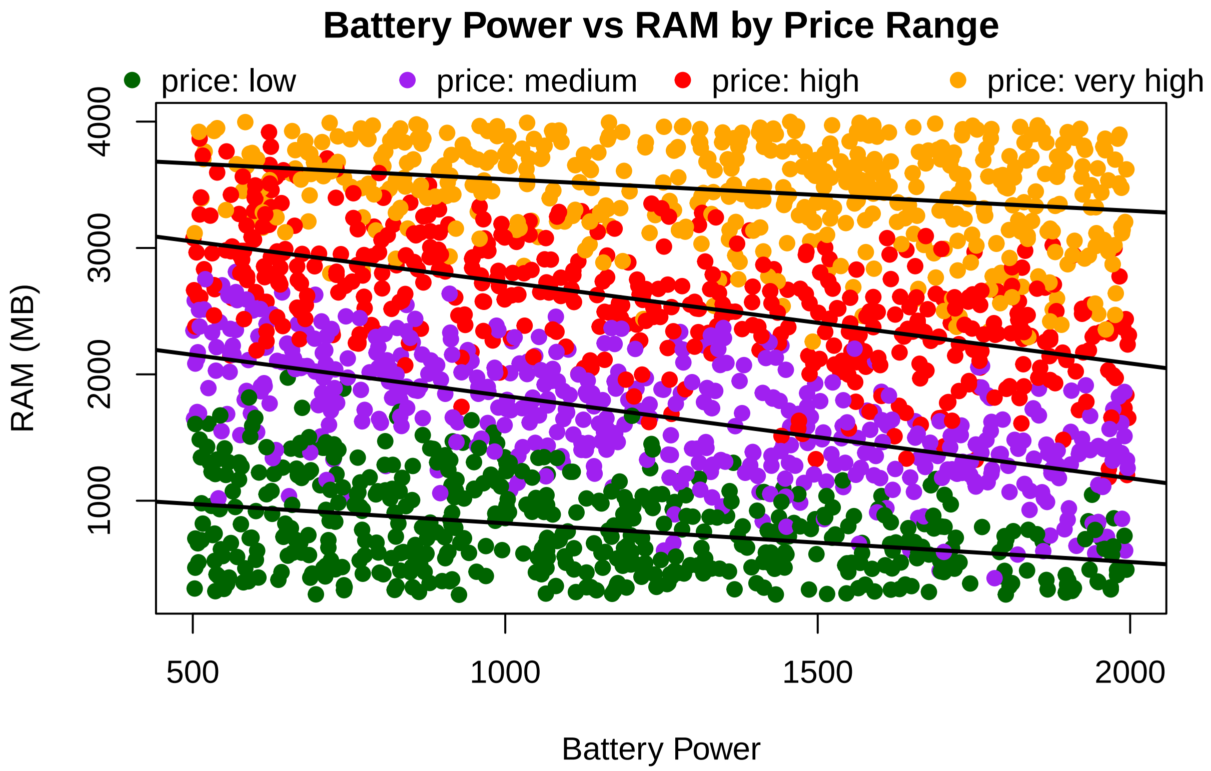
- Low: r = -0.3466

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

The correlations within each price range are negative and moderate in size, roughly between –0.6 and –0.2. By contrast, the overall correlation from part (c) is essentially zero (–0.00006). This means that when all phones are considered together, there is no clear linear relationship between RAM and battery power. However, within each price group there is a moderate negative correlation, indicating that phones with higher RAM often come with lower battery power, or vice versa. This shows the importance of looking at subgroups separately, since the overall result can hide meaningful patterns that appear within categories.

f)

```r
# Recreate scatter plot with trend lines for each price range
plot(mobile_data$battery_power, mobile_data$ram,
     col = ifelse(mobile_data$price_range == "low", "darkgreen",
                  ifelse(mobile_data$price_range == "medium", "purple",
                         ifelse(mobile_data$price_range == "high", "red", "orange"))),
     pch = 19,
     xlab = "Battery Power",
     ylab = "RAM (MB)",
     main = "Battery Power vs RAM by Price Range")

# price_range legends
legend("top", inset = c(0, -0.12),
       legend = paste("price:", levels(mobile_data$price_range)),
       col = c("darkgreen", "purple", "red", "orange"),
       pch = 19,
       horiz = TRUE,
       bty = "n",
       xpd = NA)

# Trend lines for each price range
abline(lm(ram ~ battery_power, data = low_set), col = "black", lwd = 2)
abline(lm(ram ~ battery_power, data = med_set), col = "black", lwd = 2)
abline(lm(ram ~ battery_power, data = high_set), col = "black", lwd = 2)
abline(lm(ram ~ battery_power, data = very_high_set), col = "black", lwd = 2)
```



g)

```
# 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)
```

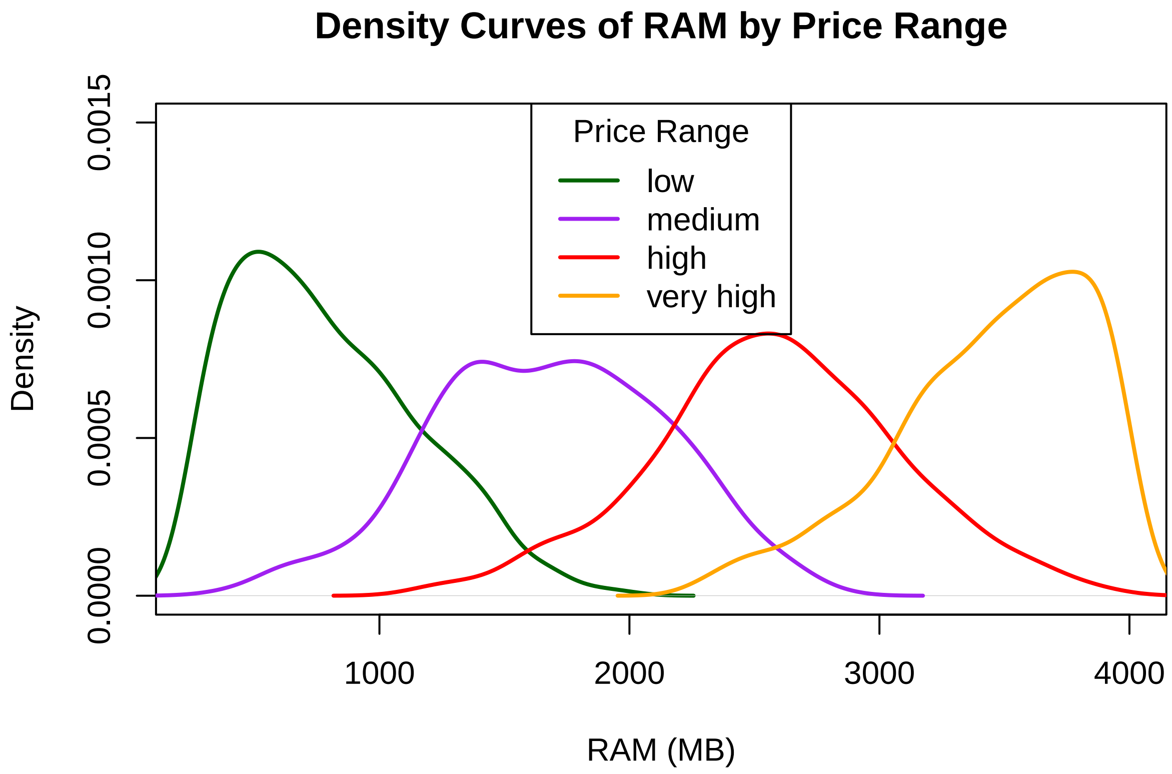
Clock Speed Statistics:

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

The average and median clock speeds don't change across different CPU cores because clock speed and number of cores appear to be independent features in this dataset. This shows that in this dataset, clock speed and number of cores were treated as independent features, rather than being linked, or that manufacturers don't pair higher core counts with different clock speeds in this sample. In other words, having more cores does not imply higher or lower clock speeds here.

h)

```r
# Density curves for RAM by price range
plot(density(low_set$ram), col = "darkgreen", lwd = 2,
     main = "Density Curves of RAM by Price Range",
     xlab = "RAM (MB)",
     ylim = c(0, 0.0015),
     xlim = c(min(mobile_data$ram), max(mobile_data$ram)))
lines(density(med_set$ram), col = "purple", lwd = 2)
lines(density(high_set$ram), col = "red", lwd = 2)
lines(density(very_high_set$ram), col = "orange", lwd = 2)
legend("top", legend = levels(mobile_data$price_range),
       col = c("darkgreen", "purple", "red", "orange"),
       lwd = 2,
       title = "Price Range")
```



Low (green curve):
The distribution is unimodal with a strong peak around 500 MB of RAM, then it gradually declines as RAM increases. This suggests that low-priced phones are concentrated around smaller RAM sizes, with very few exceeding ~1500 MB.

Medium (purple curve):
This curve rises more gradually and has two small bumps (a bimodal shape) around 1200 MB and 1500 MB. It shows that medium-priced phones are spread across a wider range of RAM, but cluster around mid-range values.

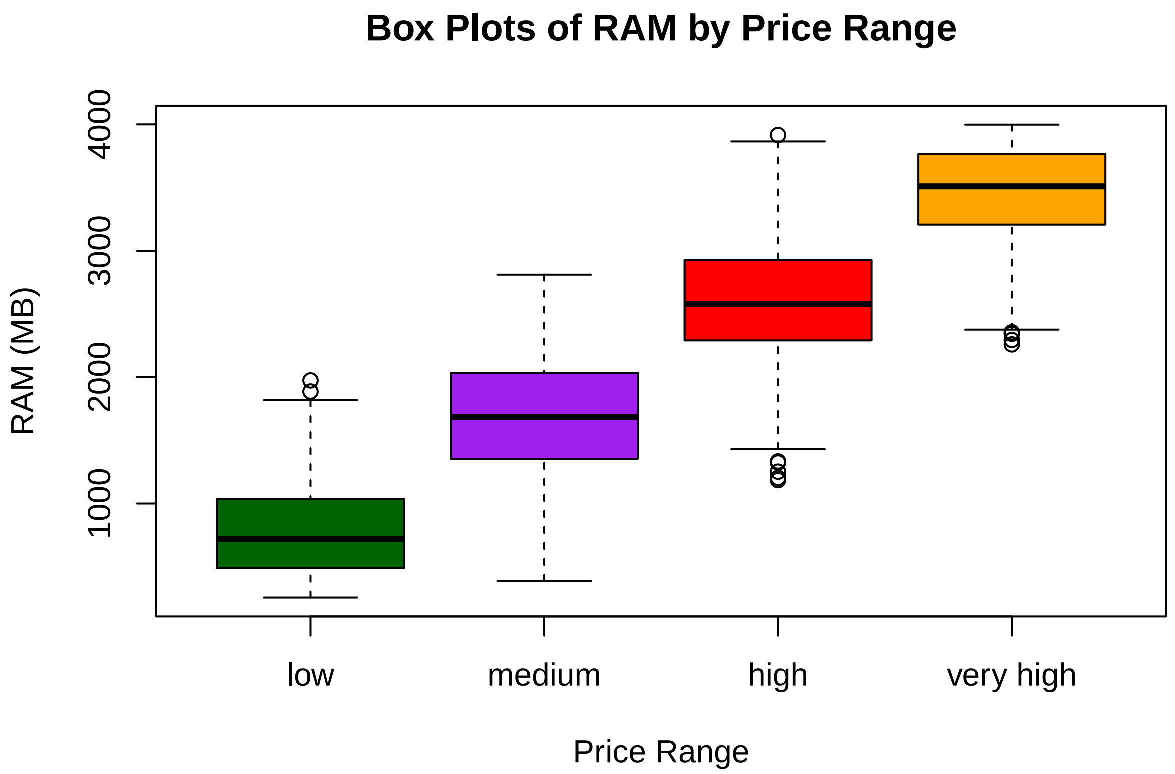
High (red curve):

The density starts later (very few below ~1200 MB) and increases steadily, peaking near 2500 MB. This indicates that high-priced phones mostly occupy the upper RAM values, pretty symmetric shape.

Very High (orange curve):
The distribution is flat near zero until about 2000 MB, then rises sharply with a narrow concentration at the very high end. This suggests that very high-priced phones almost exclusively have very large RAM values (close to or above 2000 MB).

i)

```
# Box plots for RAM by price range
cols <- c("low" = "darkgreen",
          "medium" = "purple",
          "high" = "red",
          "very high" = "orange")

boxplot(ram ~ price_range, data = mobile_data,
        col = cols[levels(mobile_data$price_range)],
        xlab = "Price Range",
        ylab = "RAM (MB)",
        main = "Box Plots of RAM by Price Range")
```



Low (green box)

- Median: around 500-600 MB.
- IQR (middle 50%): mostly below 1,000 MB.
- Several outliers above 1,500 MB.

- Indicates that most low-priced phones have relatively small RAM, with a few exceptions that have much larger RAM.

Medium (purple box)

- Median: about 1,600 MB.
- IQR: roughly 1,300-2,000 MB.
- A few outliers below 1,000 MB and above 2,500 MB.
- Suggests medium-priced phones typically sit in the mid-range RAM values, with a moderate spread.

High (red box)

- Median: around 2,500 MB.
- IQR: approximately 2,200-2,900 MB.
- Outliers appear below 1,500 MB and above 3,500 MB.
- High-priced phones generally cluster around higher RAM values, with some unusual low- or very high-RAM devices.

Very High (orange box)

- Median: around 3,600 MB.
- IQR: ~3,300-3,900 MB.
- A few outliers below 2,500 MB.
- Very high-priced phones mostly have large RAM values, tightly concentrated at the top end, with only a few exceptions.

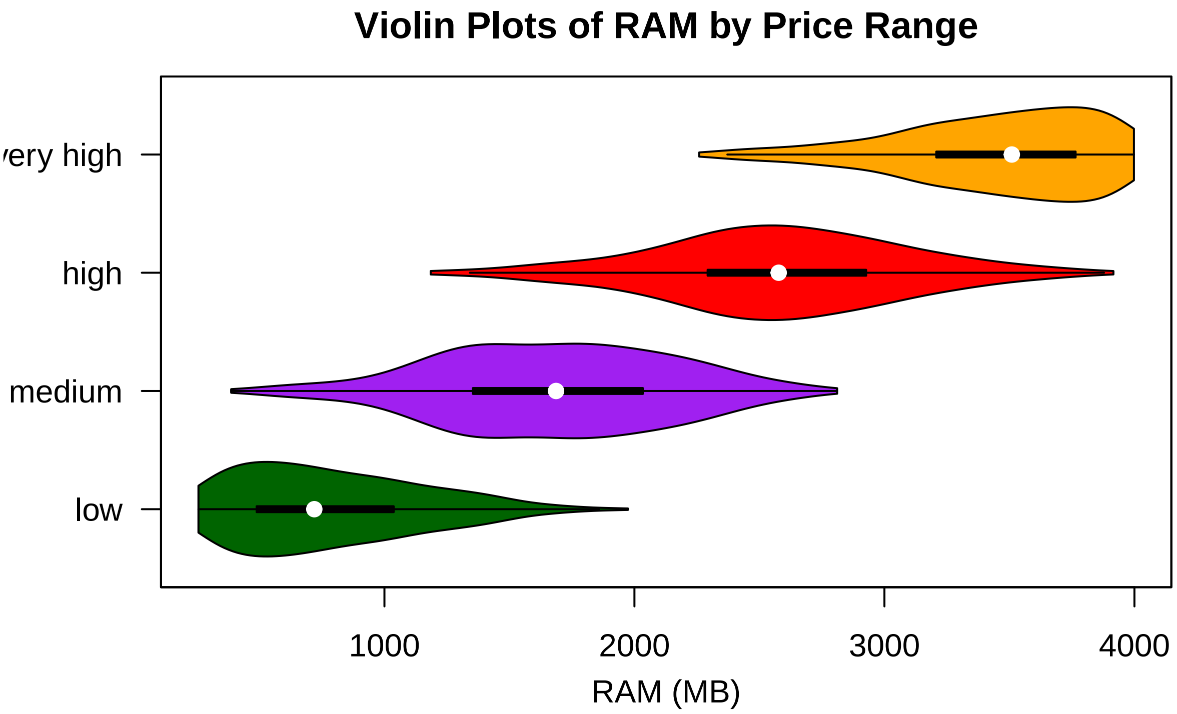
j)

```r
# 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
# widen left margin; tweak label/tick spacing
op <- par(mar = c(4.5, 6.5, 4, 2) + 0.1, mgp = c(2.2, 0.6, 0))
on.exit(par(op), add = TRUE)

ram_all <- c(low_set$ram, med_set$ram, high_set$ram, very_high_set$ram)
ticks   <- pretty(range(ram_all, na.rm = TRUE))

vioplot(low_set$ram, med_set$ram, high_set$ram, very_high_set$ram,
        names = NULL,                  # no default labels
        horizontal = TRUE,
        col = c("darkgreen", "purple", "red", "orange"),
        xaxt = "n", yaxt = "n",        # we'll draw axes ourselves
        xlab = "",                     # avoid overlapping default xlab
        main = "Violin Plots of RAM by Price Range")
```

```r
# X axis (RAM) numbers + label
axis(1, at = ticks, labels = ticks)
mtext("RAM (MB)", side = 1, line = 2.2)

# Y axis (price ranges) + label
axis(2, at = 1:4, labels = levels(mobile_data$price_range), las = 1)
mtext("Price Range", side = 2, line = 4.2)

box()  # redraw box after custom axes
```



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

Low (green)

- Low RAM values dominate; thin right tail, right-skewed.

Medium (purple)

- Centered in the mid-range with moderate spread.

High (red box)

- Uniform pattern with mild tapering at the edges; not skewed.

Very High (orange box)

- Concentrated at the top end with a left tail, left-skewed.

Clear separation and upward shift in RAM distributions as price range increases

k)

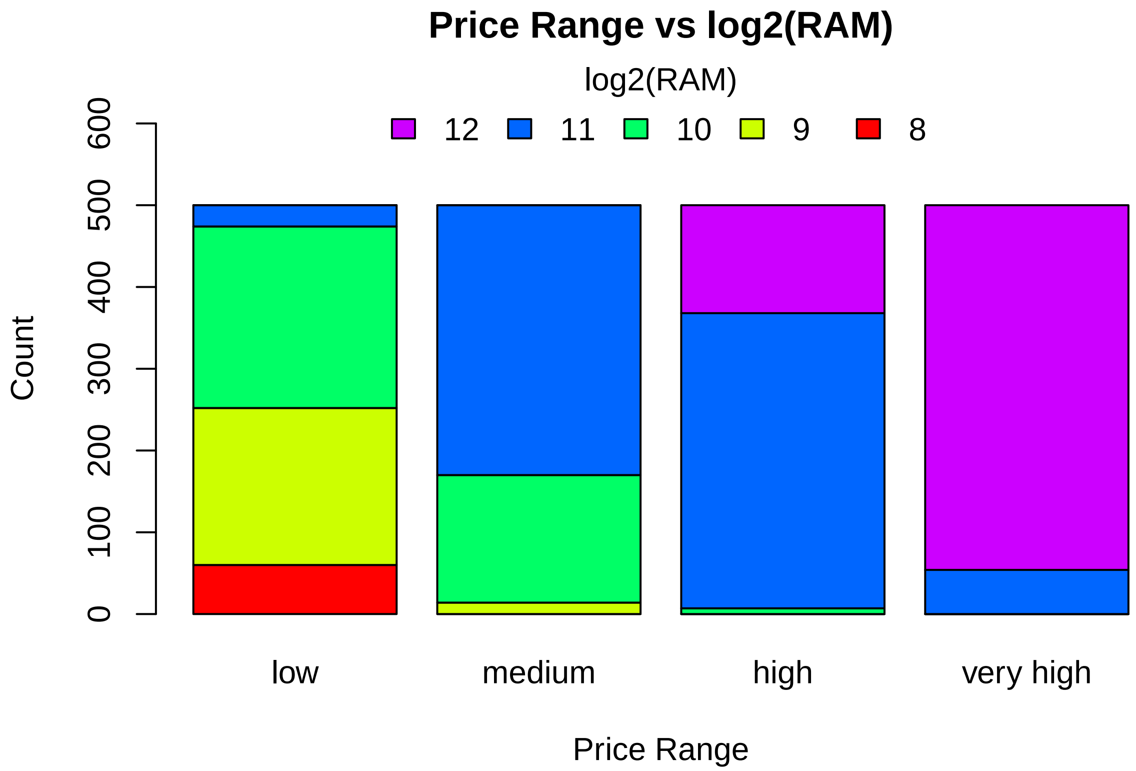
```r
log2ram <- factor(round(log2(mobile_data$ram)))
head(log2ram)
```

```
## [1] 11 11 11 11 10 10
## Levels: 8 9 10 11 12
```

Explain: - RAM Sizes are powers of 2, so taking the log will make comparison simpler and easier to analyze.

l)

```r
tab <- table(mobile_data$price_range, log2ram)

# colors for RAM types (one color per 2^k class)
ram_cols <- rainbow(ncol(tab))

# stacked barplot (counts)
barplot(t(tab),
        beside = FALSE,
        col = ram_cols,
        xlab = "Price Range",
        ylab = "Count",
        ylim = c(0, max(rowSums(tab) * 1.25)),
        main = "Price Range vs log2(RAM)",
        legend.text = colnames(tab),
        args.legend = list(title = "log2(RAM)",
                           x = "top",
                           inset = -0.1,
                           bty = "n", horiz = TRUE))
```



Problem 2

```r
library(ggplot2)
data("mpg")
head(mpg)
```

```
##   manufacturer model displ year cyl      trans drv cty hwy fl   class
## 1         audi    a4   1.8 1999   4   auto(l5)   f  18  29  p compact
## 2         audi    a4   1.8 1999   4 manual(m5)   f  21  29  p compact
## 3         audi    a4   2.0 2008   4 manual(m6)   f  20  31  p compact
## 4         audi    a4   2.0 2008   4   auto(av)   f  21  30  p compact
## 5         audi    a4   2.8 1999   6   auto(l5)   f  16  26  p compact
## 6         audi    a4   2.8 1999   6 manual(m5)   f  18  26  p compact
```

a)

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

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"))
```

```
head(mpg$trans)
```

```
## [1] auto manu manu auto auto manu
## Levels: auto manu
```

c)

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

```
## [1] f f f f f f
## Levels: f < r < 4
```

d)

```
# 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$fl, levels = c("diesel", "gasoline", "other"))
head(mpg$fl)
```

```
## [1] gasoline gasoline gasoline gasoline gasoline gasoline
## Levels: diesel gasoline other
```

```
head(mpg)
```

```
##   manufacturer model displ year cyl trans drv cty hwy       fl   class
## 1         audi    a4   1.8 1999   4  auto   f  18  29 gasoline compact
## 2         audi    a4   1.8 1999   4  manu   f  21  29 gasoline compact
## 3         audi    a4   2.0 2008   4  manu   f  20  31 gasoline compact
## 4         audi    a4   2.0 2008   4  auto   f  21  30 gasoline compact
## 5         audi    a4   2.8 1999   6  auto   f  16  26 gasoline compact
## 6         audi    a4   2.8 1999   6  manu   f  18  26 gasoline compact
```

e)

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

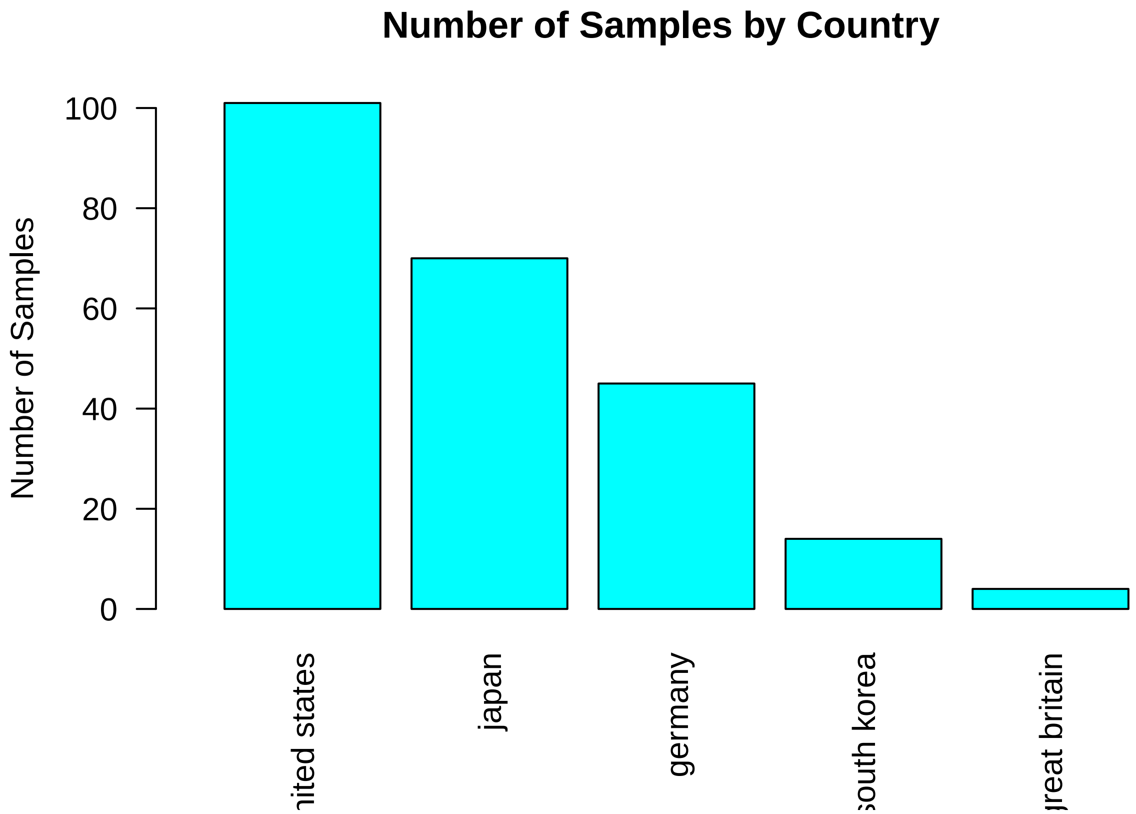
f)

```
# 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)
```

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    234 obs. of  12 variables:
##  $ manufacturer: chr  "audi" "audi" "audi" "audi" ...
##  $ model       : chr  "a4" "a4" "a4" "a4" ...
##  $ displ       : num  1.8 1.8 2 2 2.8 2.8 3.1 1.8 1.8 2 ...
##  $ year        : int  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  18 21 20 21 16 18 18 18 16 20 ...
##  $ hwy         : int  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  "germany" "germany" "germany" "germany" ...
```

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",
        ylab = "Number of Samples",
        col = "cyan",
        las = 2,
        xlim = c(0, length(country_counts) + 1))
```



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).

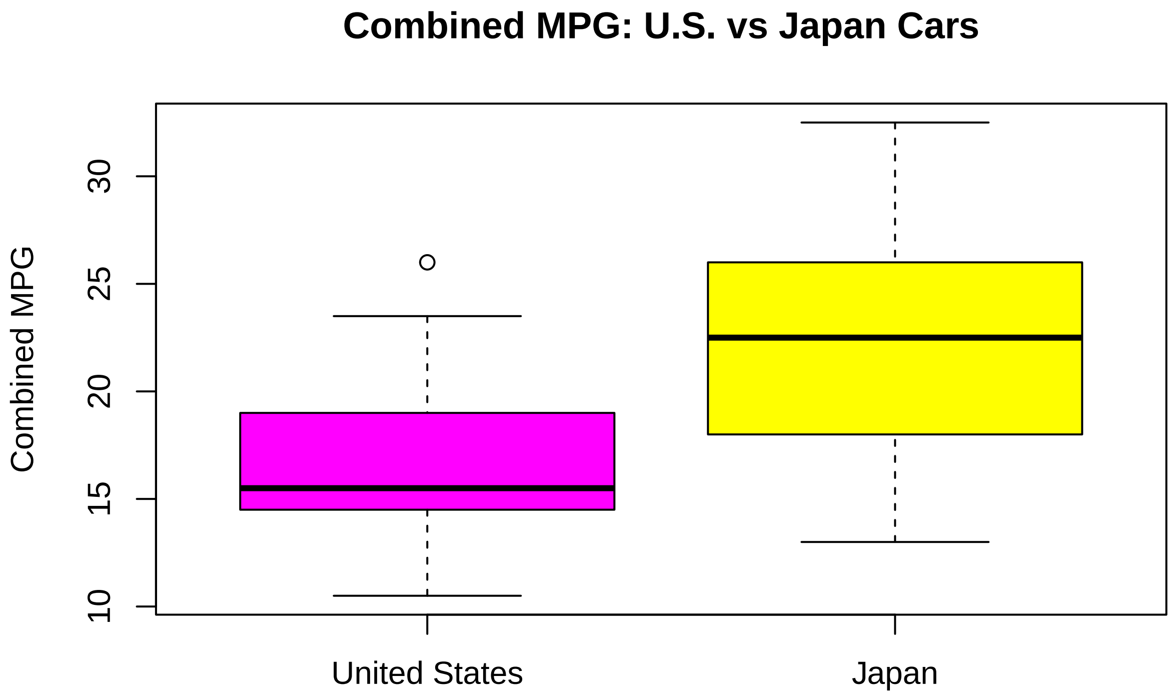
h)

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

i)

```
# 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("magenta", "yellow"))
```



```
# 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:

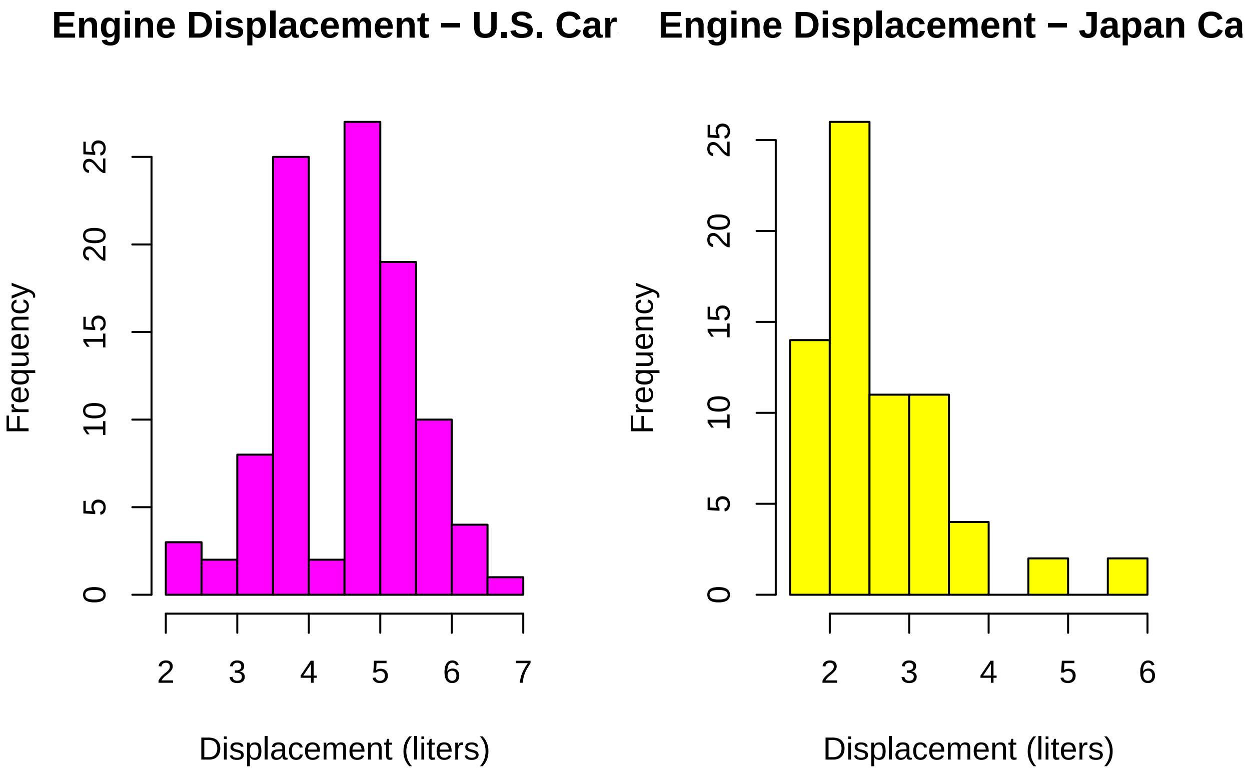
United States:
- Mean: 16.64
- Median: 15.5
- Standard Deviation: 3.3
- IQR: 4.5

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

Japanese cars have higher fuel efficiency on average compared to United States 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 = "magenta",
     breaks = 10)

hist(japan_cars$displ,
     main = "Engine Displacement - Japan Cars",
     xlab = "Displacement (liters)",
     col = "yellow",
     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 and concentration around 5.0-6.0 liters and another high around 3.5-4.0 liters. The distribution shows that U.S. manufacturers tend to produce cars with mid and large 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, this shows Japanese manufacturers tend to produce cars with smaller engines.