RWork- sheet_LAGUDA4b

Rey-Ann S. Laguda

2023-11-07

#Number1

```
matrixA <- matrix(0, nrow = 5, ncol = 5)
vectorA <- c(1, 2, 3, 4, 5)
for (i in 1:5) {
  for (j in 1:5) {
    matrixA[i, j] <- abs(vectorA[i] - vectorA[j])
  }
}

matrixA
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]    0    1    2    3    4
## [2,]    1    0    1    2    3
## [3,]    2    1    0    1    2
## [4,]    3    2    1    0    1
## [5,]    4    3    2    1    0
```

#Number2

```
num_rows <- 5

for (i in 1:num_rows) {
  for (j in 1:i) {
    cat("* ")
  }
  cat("\n")
}
```

```
## *
## * *
## * * *
## * * * *
## * * * * *
```

#Nuumber3

```
n <- as.numeric(readline("Enter the starting number for the Fibonacci sequence: "))
```

```
## Enter the starting number for the Fibonacci sequence:
```

```
a <- 0
b <- 1
```

```r
cat(a, " ")
```

```
## 0
```

```r
repeat {
  c <- a + b
  if (c > 500) {
    break
  }
  cat(c, " ")
  a <- b
  b <- c
}
```

```
## 1  2  3  5  8  13  21  34  55  89  144  233  377
```

```r
cat("\n")
```

#Number4

```r
#a
library(readr)
Shoesize <- read_csv("/cloud/project/RWorksheet4/Shoesize.csv", show_col_types = FALSE)
Shoesize
```

```
## # A tibble: 28 x 3
##    ShoeSize Height Gender
##       <dbl>  <dbl> <chr>
##  1      6.5   66   F
##  2      9     68   F
##  3      8.5   64.5 F
##  4      8.5   65   F
##  5     10.5   70   M
##  6      7     64   F
##  7      9.5   70   F
##  8      9     71   F
##  9     13     72   M
## 10      7.5   64   F
## # i 18 more rows
```

```r
 head(Shoesize, n = 6)
```

```
## # A tibble: 6 x 3
##   ShoeSize Height Gender
##      <dbl>  <dbl> <chr>
## 1      6.5   66   F
## 2      9     68   F
## 3      8.5   64.5 F
## 4      8.5   65   F
## 5     10.5   70   M
## 6      7     64   F
```

```r
 head
```

```
## function (x, ...)
## UseMethod("head")
## <bytecode: 0x55c51cf7f208>
## <environment: namespace:utils>
```

```r
#b
male_subset <- subset(Shoesize, Gender == "M")
female_subset <- subset(Shoesize, Gender == "F")

male_subset
```

```
## # A tibble: 14 x 3
##    ShoeSize Height Gender
##       <dbl>  <dbl> <chr>
##  1     10.5   70   M
##  2     13     72   M
##  3     10.5   74.5 M
##  4     12     71   M
##  5     10.5   71   M
##  6     13     77   M
##  7     11.5   72   M
##  8     10     72   M
##  9      8.5   67   M
## 10     10.5   73   M
## 11     10.5   72   M
## 12     11     70   M
## 13      9     69   M
## 14     13     70   M
```

```r
female_subset
```

```
## # A tibble: 14 x 3
##    ShoeSize Height Gender
##       <dbl>  <dbl> <chr>
##  1      6.5   66   F
##  2      9     68   F
##  3      8.5   64.5 F
##  4      8.5   65   F
##  5      7     64   F
##  6      9.5   70   F
##  7      9     71   F
##  8      7.5   64   F
##  9      8.5   67   F
## 10      8.5   59   F
## 11      5     62   F
## 12      6.5   66   F
## 13      7.5   64   F
## 14      8.5   69   F
```

```r
num_male_observations <- nrow(male_subset)
num_female_observations <- nrow(female_subset)

cat("Number if observations in Male: ", num_male_observations, "\n")
```

```
## Number if observations in Male:  14
```

```r
cat("Number if observations in Female: ", num_female_observations, "\n")
```

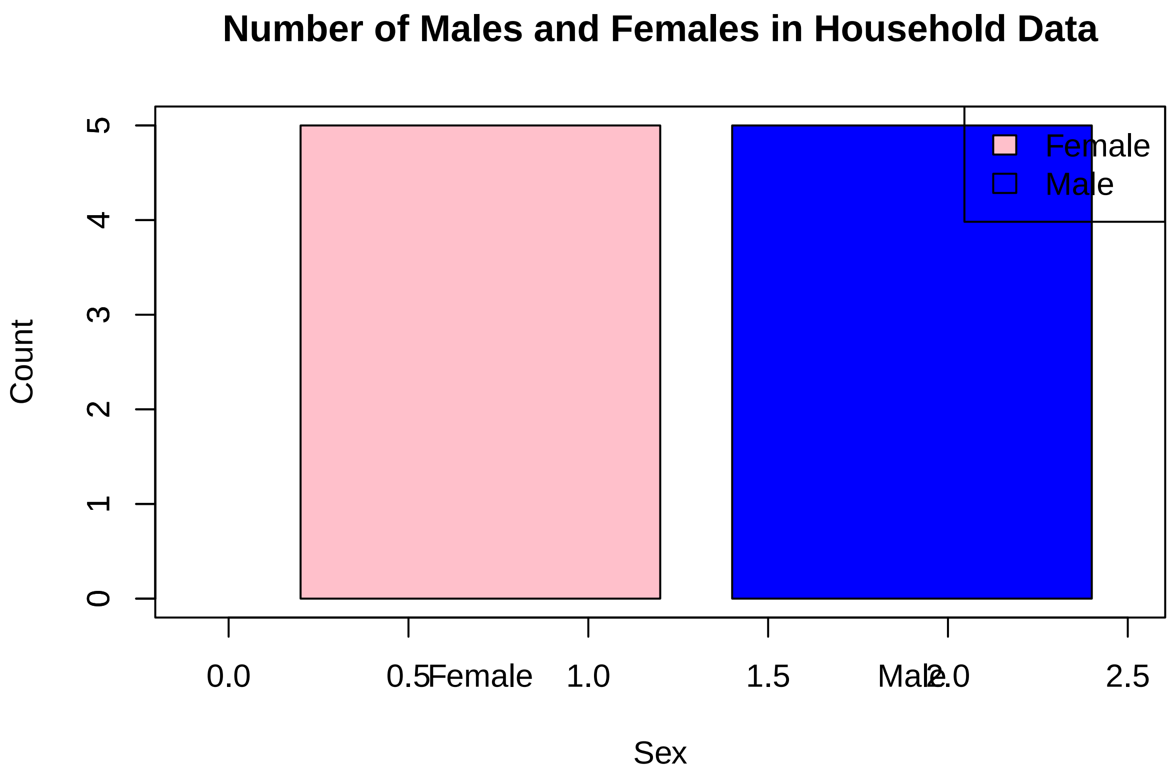
```
## Number if observations in Female:  14
```

#4c

```r
household <- read.csv("HouseholdData.csv")
household
```

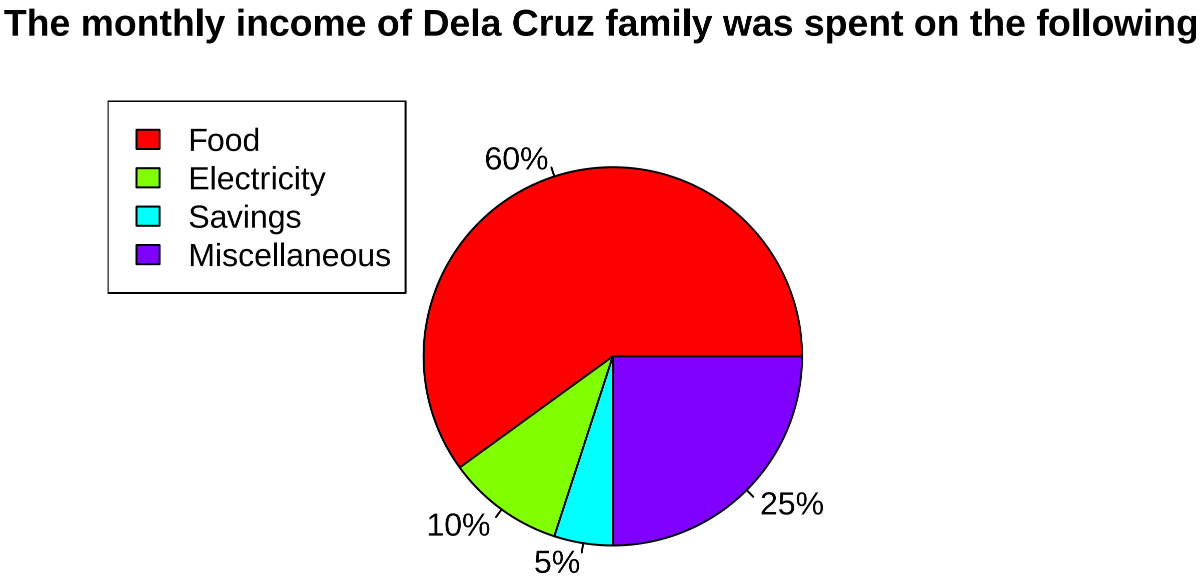
```
##    Respondents    Sex Fathers_Occupation Person_at_Home Siblings_at_School
## 1            1   Male                  1              5                  2
## 2            2 Female                  2              7                  3
## 3            3 Female                  3              3                  0
## 4            4   Male                  3              8                  5
## 5            5   Male                  1              6                  2
## 6            6 Female                  2              4                  3
## 7            7 Female                  2              4                  1
## 8            8   Male                  3              2                  2
## 9            9 Female                  1             11                  6
## 10          10   Male                  3              6                  2
##    Types_of_Houses
## 1             Wood
## 2        Congcrete
## 3        Congcrete
## 4             Wood
## 5  Semi-Crongcrete
## 6   Semi-Congcrete
## 7             Wood
## 8   Semi-Congcrete
## 9   Semi-Congcrete
## 10       Congcrete
```

```r
gender_counts <- table(household$Sex)

plot(1, type = "n", main = "Number of Males and Females in Household Data",
     xlab = "Sex", ylab = "Count", xlim = c(-0.1, 2.5), ylim = c(0, max(gender_counts)))

barplot(gender_counts, col = c("pink", "blue"), add = TRUE)

legend("topright", legend = levels(as.factor(household$Sex)), fill = c("pink", "blue"))
```



#5

```r
pie_chart <- c(60, 10, 5, 25)
pie(pie_chart, labels = paste0(pie_chart, "%"),
    main = "The monthly income of Dela Cruz family was spent on the following: ", col = rainbow(length(

legend("topleft", legend = c("Food", "Electricity", "Savings", "Miscellaneous"),
       fill = rainbow(length(pie_chart)))
```



#6

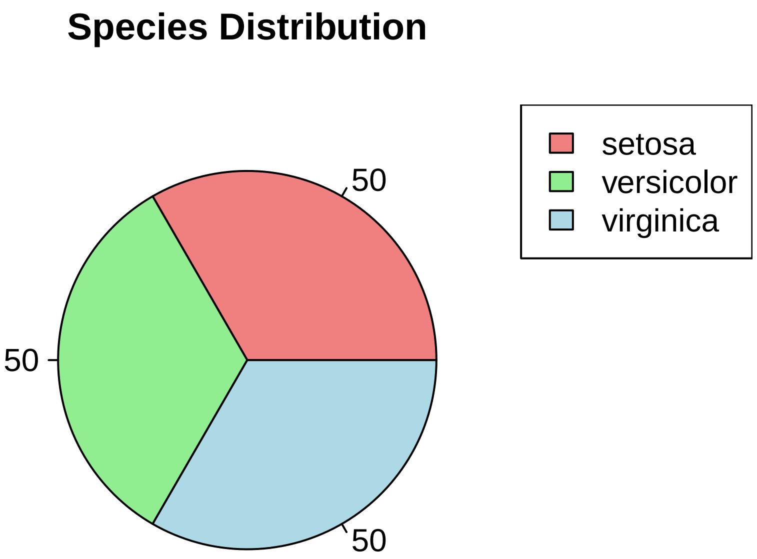
```r
data (iris)
str(iris)
```

```
## 'data.frame':    150 obs. of  5 variables:
##  $ Sepal.Length: num  5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...
##  $ Sepal.Width : num  3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...
##  $ Petal.Length: num  1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...
##  $ Petal.Width : num  0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...
##  $ Species     : Factor w/ 3 levels "setosa","versicolor",..: 1 1 1 1 1 1 1 1 1 1 ...
```

```r
mean_values <- colMeans(iris[, 1:4])
mean_values
```

```
## Sepal.Length  Sepal.Width Petal.Length  Petal.Width
##     5.843333     3.057333     3.758000     1.199333
```

```r
species_counts <- table(iris$Species)

colors <- c("lightcoral", "lightgreen", "lightblue")

pie (species_counts, labels = species_counts, col= colors, main = "Species Distribution")

legend("topright", legend = levels(iris$Species), fill = colors)
```



```r
cat("Last six rows of Setosa subset:\n")
```

```
## Last six rows of Setosa subset:
```

```r
tail(subset(iris, Species == "setosa"), 6)
```

```
##    Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 45          5.1         3.8          1.9         0.4  setosa
## 46          4.8         3.0          1.4         0.3  setosa
## 47          5.1         3.8          1.6         0.2  setosa
## 48          4.6         3.2          1.4         0.2  setosa
## 49          5.3         3.7          1.5         0.2  setosa
## 50          5.0         3.3          1.4         0.2  setosa
```

```r
cat("\nLast six rows of Versicolor subset:\n")
```

```
##
## Last six rows of Versicolor subset:
```

```r
tail(subset(iris, Species == "versicolor"), 6)
```

```
##     Sepal.Length Sepal.Width Petal.Length Petal.Width    Species
## 95           5.6         2.7          4.2         1.3 versicolor
## 96           5.7         3.0          4.2         1.2 versicolor
## 97           5.7         2.9          4.2         1.3 versicolor
## 98           6.2         2.9          4.3         1.3 versicolor
## 99           5.1         2.5          3.0         1.1 versicolor
## 100          5.7         2.8          4.1         1.3 versicolor
```

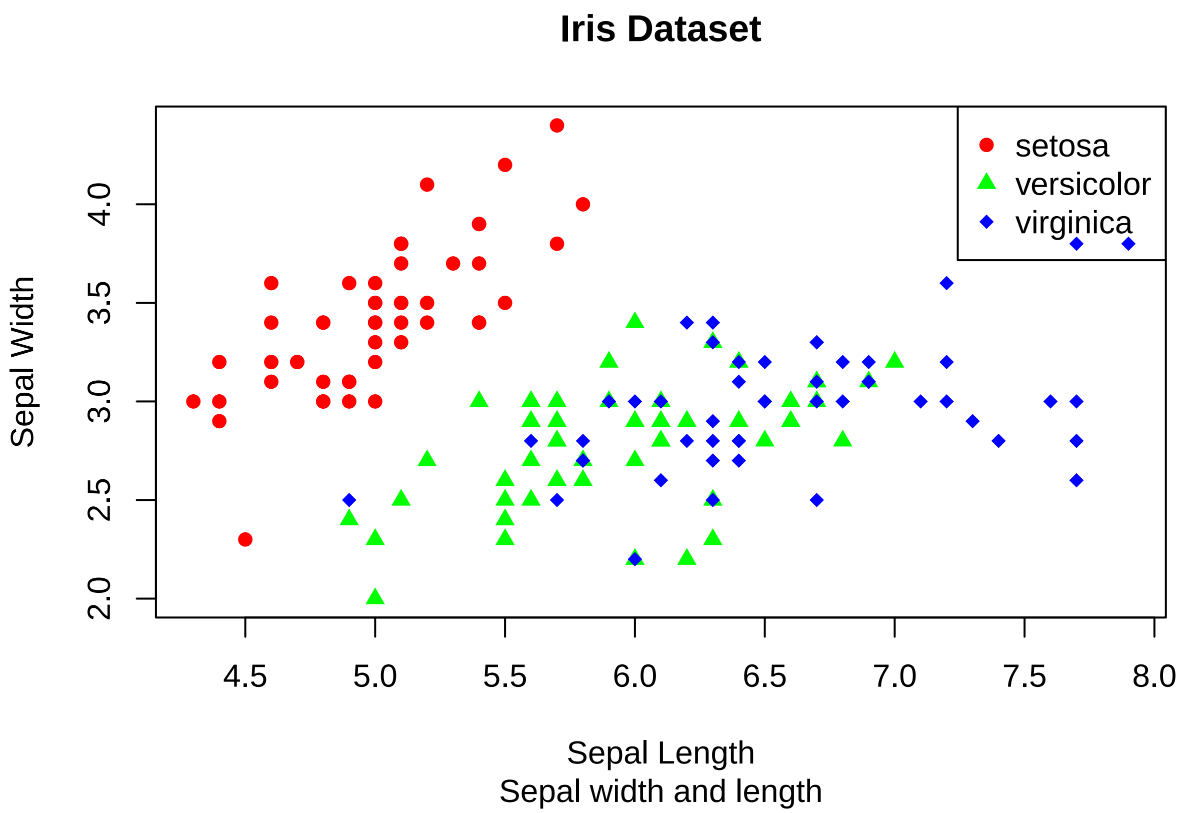
```r
cat("\nLast six rows of Virginica subset:\n")
```

```
##
## Last six rows of Virginica subset:
```

```r
tail(subset(iris, Species == "virginica"), 6)
```

```
##     Sepal.Length Sepal.Width Petal.Length Petal.Width   Species
## 145          6.7         3.3          5.7         2.5 virginica
## 146          6.7         3.0          5.2         2.3 virginica
## 147          6.3         2.5          5.0         1.9 virginica
## 148          6.5         3.0          5.2         2.0 virginica
## 149          6.2         3.4          5.4         2.3 virginica
## 150          5.9         3.0          5.1         1.8 virginica
```

```r
colors <- c("setosa" = "red", "versicolor" = "green", "virginica" = "blue")
symbols <- c("setosa" = 16, "versicolor" = 17, "virginica" = 18)

# Create a scatterplot
plot(iris$Sepal.Length, iris$Sepal.Width,
     col = colors[as.character(iris$Species)],
     pch = symbols[as.character(iris$Species)],
     main = "Iris Dataset",
     sub = "Sepal width and length",
     xlab = "Sepal Length",
     ylab = "Sepal Width")

# Add legend
legend("topright", legend = levels(iris$Species), col = colors, pch = symbols)
```



#7

```
library(readxl)
alexa_file <- read_excel("/cloud/project/RWorksheet4/alexa_file.xlsx")
alexa_file
```

```
## # A tibble: 3,150 x 5
##    rating date                variation          verified_reviews     feedback
##     <dbl> <dttm>              <chr>              <chr>                   <dbl>
##  1      5 2018-07-31 00:00:00 Charcoal Fabric    Love my Echo!               1
##  2      5 2018-07-31 00:00:00 Charcoal Fabric    Loved it!                   1
##  3      4 2018-07-31 00:00:00 Walnut Finish      Sometimes while play~       1
##  4      5 2018-07-31 00:00:00 Charcoal Fabric    I have had a lot of ~       1
##  5      5 2018-07-31 00:00:00 Charcoal Fabric    Music                       1
##  6      5 2018-07-31 00:00:00 Heather Gray Fabric I received the echo ~       1
##  7      3 2018-07-31 00:00:00 Sandstone Fabric   Without having a cel~       1
##  8      5 2018-07-31 00:00:00 Charcoal Fabric    I think this is the ~       1
##  9      5 2018-07-30 00:00:00 Heather Gray Fabric looks great                 1
## 10      5 2018-07-30 00:00:00 Heather Gray Fabric Love it! I've listen~       1
## # i 3,140 more rows
```

#7a

```
# Remove extra whitespaces in black variants
alexa_file$variation <- gsub("\\s+", " ", alexa_file$variation)
alexa_file$variation <- gsub("Black ", "Black", alexa_file$variation)

# Remove extra whitespaces in white variants
alexa_file$variation <- gsub("\\s+", " ", alexa_file$variation)
alexa_file$variation <- gsub("White ", "White", alexa_file$variation)
```

#7b

```r
# Install and load the dplyr package
if (!require(dplyr)) {
  install.packages("dplyr")
}
```

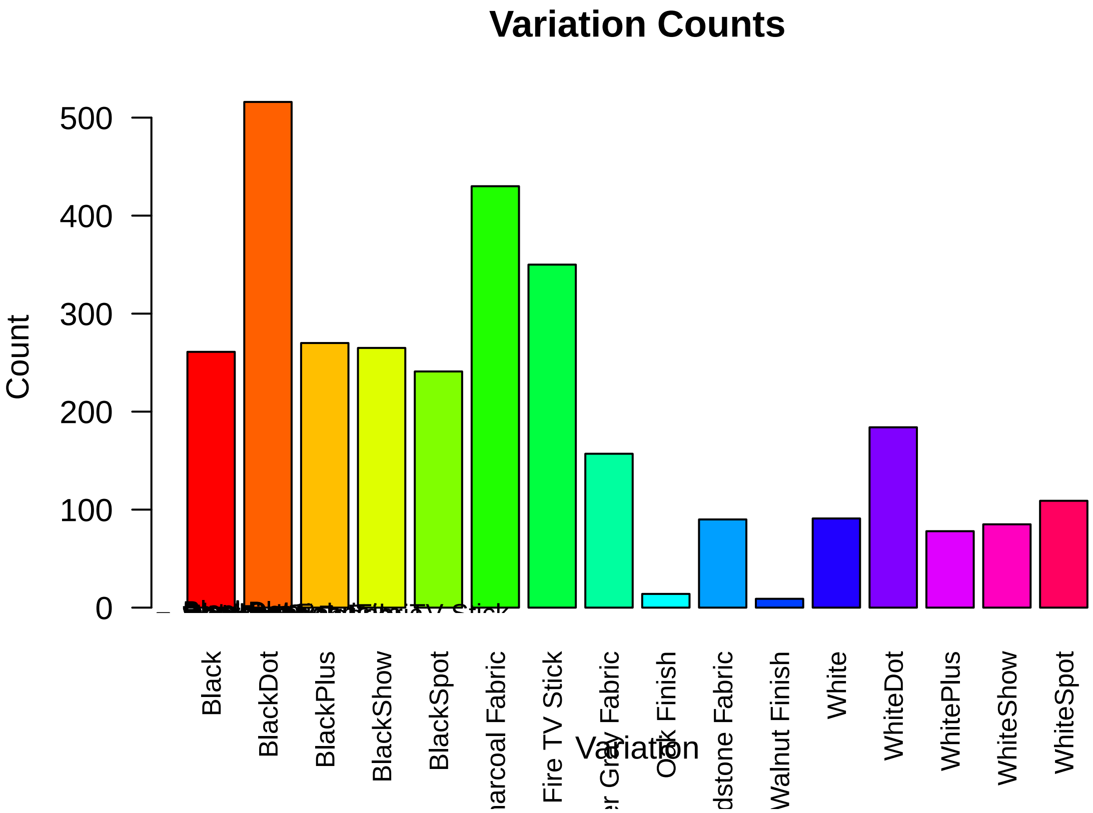
```
## Loading required package: dplyr

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(dplyr)



# Group by Variation and calculate the total count
variation_counts <- alexa_file %>%
  group_by(variation) %>%
  summarise(Count = n())

# Save the object as variations.RData
save(variation_counts, file = "variations.RData")


variation_counts
```

```
## # A tibble: 16 x 2
##    variation                    Count
##    <chr>                        <int>
##  1 Black                          261
##  2 BlackDot                       516
##  3 BlackPlus                      270
##  4 BlackShow                      265
##  5 BlackSpot                      241
##  6 Charcoal Fabric                430
##  7 Configuration: Fire TV Stick   350
##  8 Heather Gray Fabric            157
##  9 Oak Finish                      14
## 10 Sandstone Fabric                90
## 11 Walnut Finish                    9
## 12 White                           91
## 13 WhiteDot                       184
## 14 WhitePlus                       78
## 15 WhiteShow                       85
## 16 WhiteSpot                      109
```

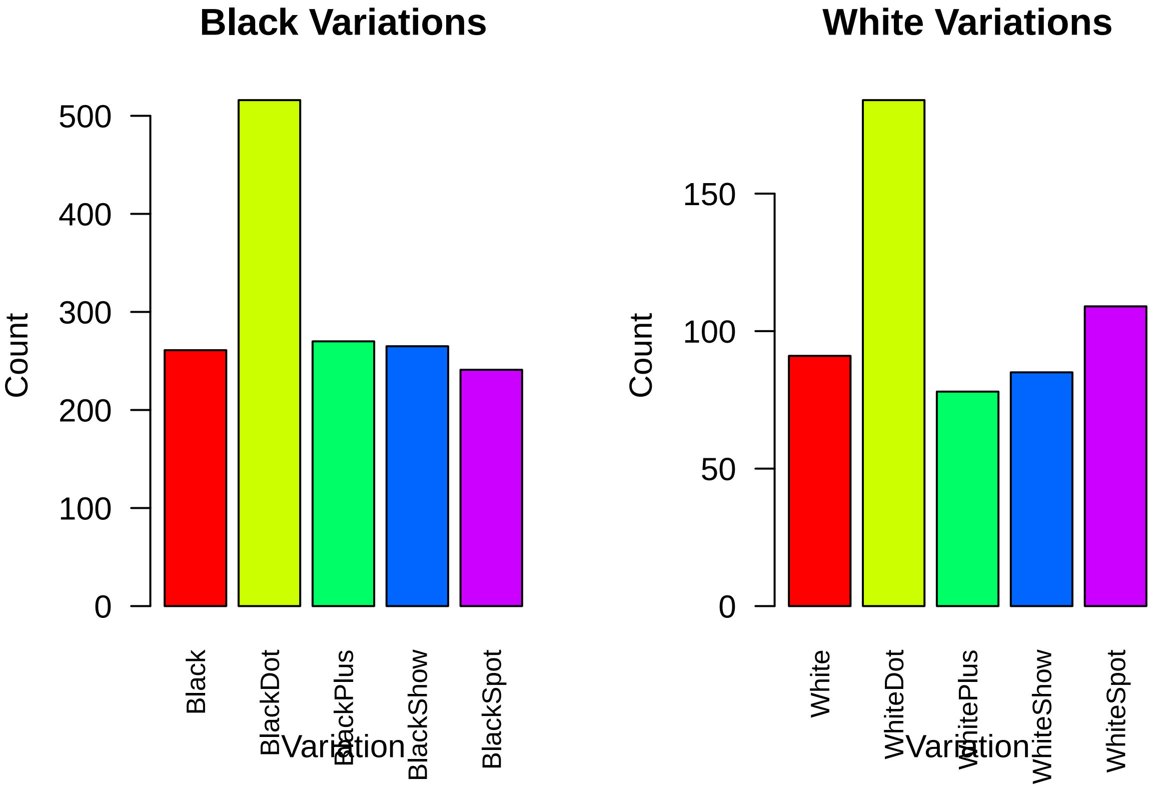
#7c

```r
# Load the variations.RData file
load("variations.RData")

# Increase the size of the plot
par(mar = c(5, 5, 4, 2) + 0.1)  # Adjust the margins

# Create a barplot with rotated x-axis labels
barplot(variation_counts$Count,
        names.arg = variation_counts$variation,
        col = rainbow(length(variation_counts$variation)),
        main = "Variation Counts",
        xlab = "Variation",
        ylab = "Count",
        las = 2,      # Rotate x-axis labels 90 degrees
        cex.names = 0.8,  # Adjust the size of the x-axis labels
        width = 0.8)  # Adjust the width of the bars

# Manually add legend in topright
legend_labels <- variation_counts$variation
legend_colors <- rainbow(length(legend_labels))

for (i in seq_along(legend_labels)) {
  rect(max(par("usr")[1]) + 0.1,
       max(par("usr")[3]) - i * 0.5,
       max(par("usr")[1]) + 0.3,
       max(par("usr")[3]) - (i + 1) * 0.5,
       col = legend_colors[i])

  text(max(par("usr")[1]) + 0.4,
       max(par("usr")[3]) - i * 0.5,
       labels = legend_labels[i],
       pos = 4,
       offset = 0.2,
       cex = 0.8)
}
```



#7d

```
# Load the variations.RData file
load("variations.RData")

# Extract data for black and white variations
black_variations <- variation_counts[variation_counts$variation %in% c("Black", "BlackDot", "BlackPlus"
white_variations <- variation_counts[variation_counts$variation %in% c("White", "WhiteDot", "WhitePlus"

# Set up the plotting area
par(mfrow = c(1, 2))  # 1 row, 2 columns

# Barplot for black variations
barplot(black_variations$Count,
        names.arg = black_variations$variation,
        col = rainbow(length(black_variations$variation)),
        main = "Black Variations",
        xlab = "Variation",
        ylab = "Count",
        las = 2,  # Rotate x-axis labels 90 degrees
        cex.names = 0.8,  # Adjust the size of the x-axis labels
        width = 0.8)  # Adjust the width of the bars

# Barplot for white variations
barplot(white_variations$Count,
        names.arg = white_variations$variation,
        col = rainbow(length(white_variations$variation)),
        main = "White Variations",
        xlab = "Variation",
        ylab = "Count",
        las = 2,  # Rotate x-axis labels 90 degrees
        cex.names = 0.8,  # Adjust the size of the x-axis labels
        width = 0.8)  # Adjust the width of the bars
```


Black Variations
Count
Black
BlackDot
BlackPlus
BlackShow
BlackSpot
Variation
White Variations
Count
White
WhiteDot
WhitePlus
WhiteShow
WhiteSpot
Variation