RWorksheet_Tiad#4b

James Cedrick Tiad

2024-10-28

```
#USING LOOP FUNCTION
#1.
vectorA <- c(1, 2, 3, 4, 5)


matrixB <- matrix(0, nrow = 5, ncol = 5)


for (i in 1:5) {
  for (j in 1:5) {
    difference <- abs(i - j)
    matrixB[i, j] <- vectorA[difference + 1]
  }
}

print(matrixB)
```

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

```
#2.
for(i in 1:5){
  cat(rep("*", i), "\n")
}
```

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

```
#3.

Startnum  <- 10

num1 <- 0
```

```r
num2 <- 1
next_num <- Startnum

repeat {
  cat(next_num, " ")

  num1 <- num2
  num2 <- next_num
  next_num <- num1 + num2

  if (next_num >= 500) {
    break
  }
}
```

```
## 10  11  21  32  53  85  138  223  361
```

```r
#Using Basic Graphics (plot(),barplot(),pie(),hist())
#4.
#a.
shoes <- read.csv(file = "Shoe_Table.csv")
shoes
```

```
##     X Shoe_size Height Gender
## 1   1       6.5   66.0      F
## 2   2       9.0   68.0      F
## 3   3       8.5   64.5      F
## 4   4       8.5   65.0      F
## 5   5      10.5   70.0      M
## 6   6       7.0   64.0      F
## 7   7       9.5   70.0      F
## 8   8       9.0   71.0      F
## 9   9      13.0   72.0      M
## 10 10       7.5   64.0      F
## 11 11      10.5   74.5      M
## 12 12       8.5   67.0      F
## 13 13      12.0   71.0      M
## 14 14      10.5   71.0      M
## 15 15      13.0   77.0      M
## 16 16      11.5   72.0      M
## 17 17       8.5   59.0      F
## 18 18       5.0   62.0      F
## 19 19      10.0   72.0      M
## 20 20       6.5   66.0      F
## 21 21       7.5   64.0      F
## 22 22       8.5   67.0      M
## 23 23      10.5   73.0      M
## 24 24       8.5   69.0      F
## 25 25      10.5   72.0      M
## 26 26      11.0   70.0      M
## 27 27       9.0   69.0      M
## 28 28      13.0   70.0      M
```

```r
head(shoes, n = 6L)
```

```
##   X Shoe_size Height Gender
## 1 1       6.5   66.0      F
## 2 2       9.0   68.0      F
## 3 3       8.5   64.5      F
## 4 4       8.5   65.0      F
## 5 5      10.5   70.0      M
## 6 6       7.0   64.0      F
```

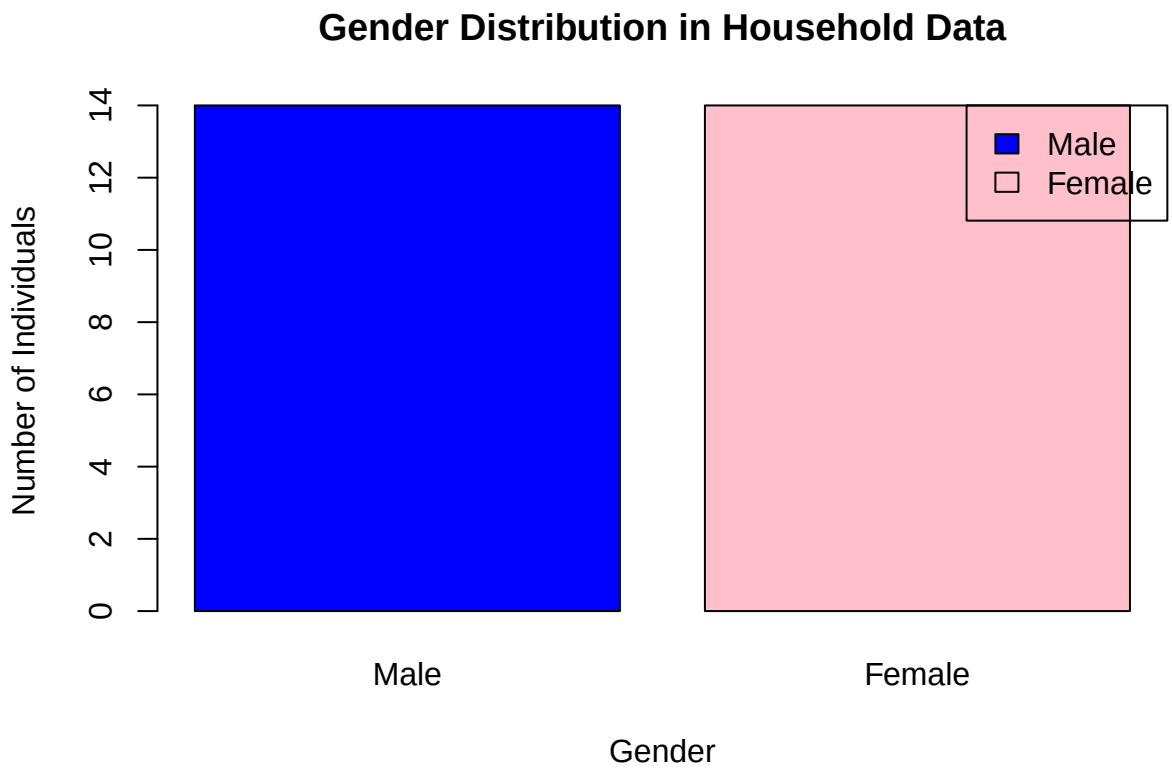
```r
#b.
female_data <- subset(shoes, Gender == "F")
male_data <- subset(shoes, Gender == "M")

num_males <- nrow(male_data)
num_females <- nrow(female_data)

cat("Number of males:", num_males, "\n")
```

```
## Number of males: 14
```

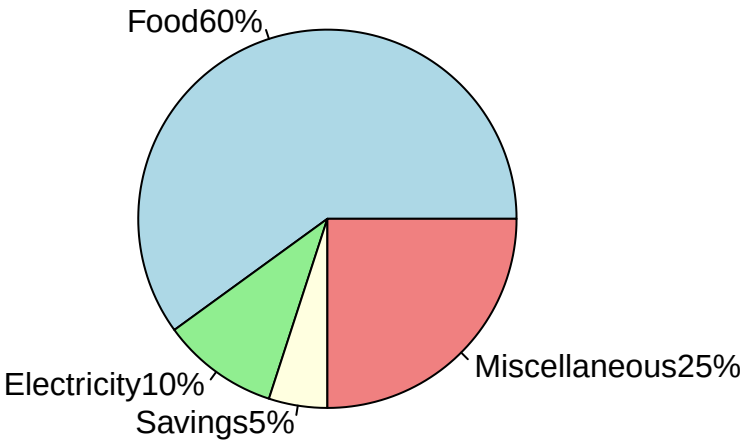
```r
cat("Number of females:", num_females)
```

```
## Number of females: 14
```

```r
#c.
gender_counts <- c(num_males, num_females)
gender_labels <- c("Male", "Female")

barplot(gender_counts, names.arg = gender_labels, col = c("blue", "pink"),
        main = "Gender Distribution in Household Data",
        xlab = "Gender", ylab = "Number of Individuals",
        legend.text = c("Male", "Female"), args.legend = list(x = "topright"))
```




```r
#5.
#a.
expenses <- c(60, 10, 5, 25)
expense_labels <- c("Food", "Electricity", "Savings", "Miscellaneous")

percentages <- round(expenses / sum(expenses) * 100, 1)

pie(expenses, labels = paste(expense_labels, percentages, "%", sep = ""),
    col = c("lightblue", "lightgreen", "lightyellow", "lightcoral"),
    main = "Dela Cruz Family Monthly Expenses")
```



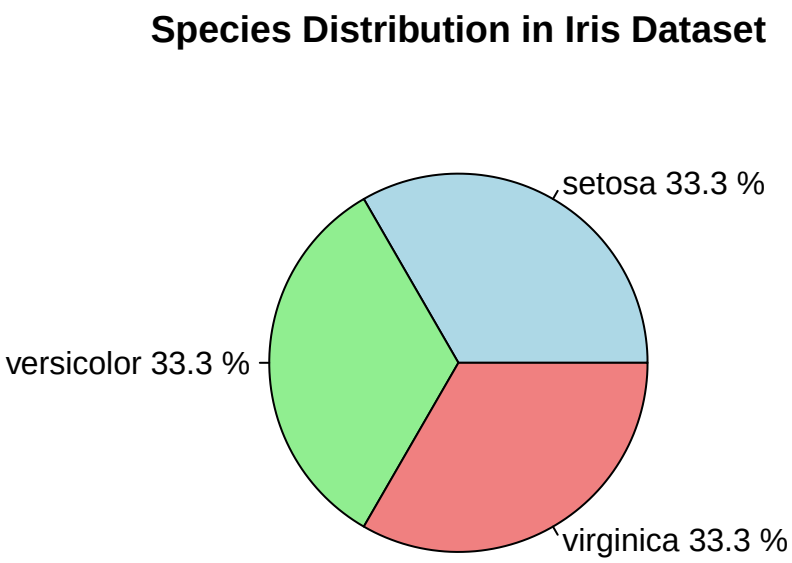

```
#6.
#a.
data(iris)
str(iris)
```

```
## '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 ...
```

```
#b.
iris_means <- c(mean(iris$Sepal.Length), mean(iris$Sepal.Width), mean(iris$Petal.Length), mean(iris$Peta
iris_means
```

```
## [1] 5.843333 3.057333 3.758000 1.199333
```

```
#c.
pie(table(iris$Species),
    labels = paste(names(table(iris$Species)), round(table(iris$Species)/nrow(iris)*100, 1), "%", sep =
    col = c("lightblue", "lightgreen", "lightcoral"),
    main = "Species Distribution in Iris Dataset")
```




```r
#d.
setosa <- iris[iris$Species == "setosa", ]
versicolor <- iris[iris$Species == "versicolor", ]
virginica <- iris[iris$Species == "virginica", ]

tail(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
```

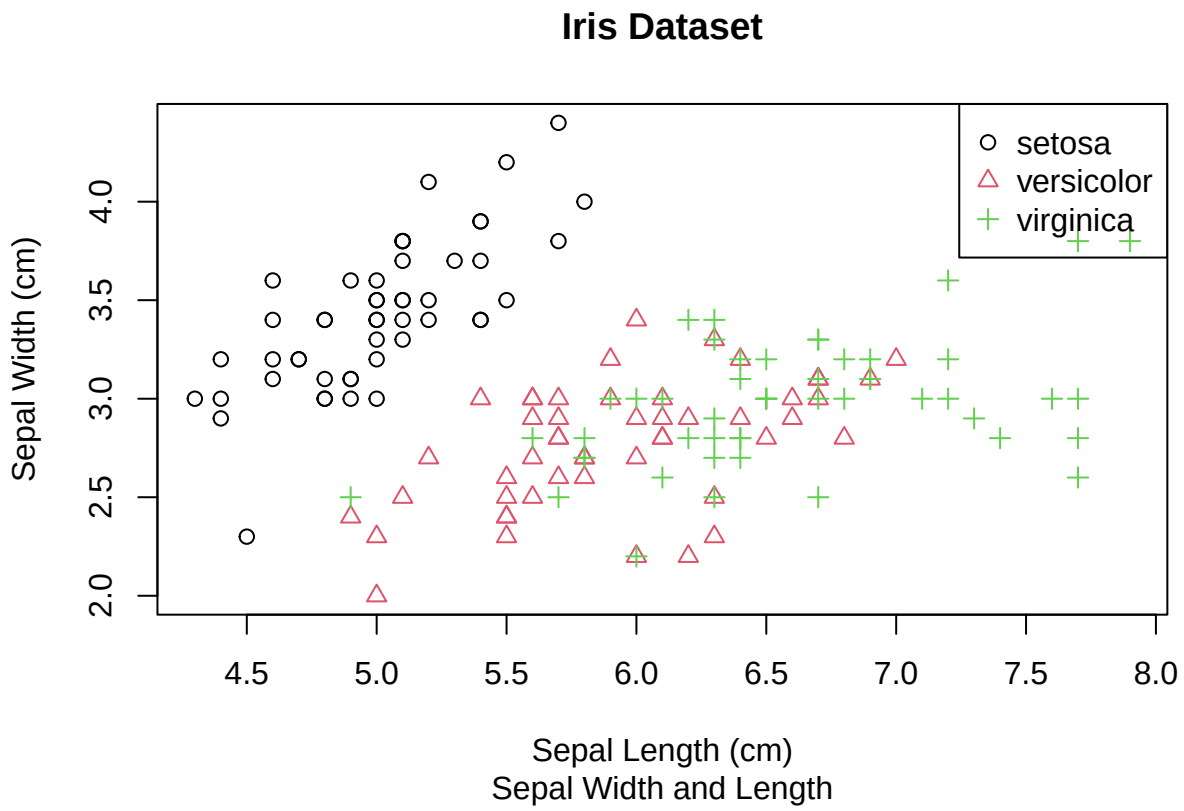
```r
tail(versicolor, 6)
```

```
##     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
tail(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
#e.
plot(iris$Sepal.Length, iris$Sepal.Width,
     col = as.numeric(iris$Species),
     pch = as.numeric(iris$Species),
     main = "Iris Dataset",
     sub = "Sepal Width and Length",
     xlab = "Sepal Length (cm)",
     ylab = "Sepal Width (cm)")

legend("topright", legend = levels(iris$Species),
       col = 1:3, pch = 1:3)
```




```r
#f.
#The scatterplot shows us that the different iris species have distinct features
```

```
  #when it comes to their sepals.

  #Setosa flowers tend to have shorter sepals and wider sepals compared to the
      #other two species.

  #Versicolor and Virginica have more similar sepal sizes, but Virginica flowers
    #generally have longer sepals than Versicolor flowers.

#This means that by looking at the sepal length and width, we can get a good
#idea of which species of iris we're dealing with!
```

```
#7.
#a.
library(readxl)
alexa_data <- read_excel("alexa_file.xlsx")


alexa_data$variation <- gsub("Black\\s+Dot", "Black Dot", alexa_data$variation)
alexa_data$variation <- gsub("Black\\s+Plus", "Black Plus", alexa_data$variation)
alexa_data$variation <- gsub("Black\\s+Show", "Black Show", alexa_data$variation)
alexa_data$variation <- gsub("Black\\s+Spot", "Black Spot", alexa_data$variation)
alexa_data$variation <- gsub("White\\s+Dot", "White Dot", alexa_data$variation)
alexa_data$variation <- gsub("White\\s+Plus", "White Plus", alexa_data$variation)
alexa_data$variation <- gsub("White\\s+Show", "White Show", alexa_data$variation)
alexa_data$variation <- gsub("White\\s+Spot", "White Spot", alexa_data$variation)


table(alexa_data$variation)
```

```
##
##                         Black                     Black Dot
##                           261                           516
##                    Black Plus                    Black Show
##                           270                           265
##                    Black Spot              Charcoal Fabric
##                           241                           430
## Configuration: Fire TV Stick          Heather Gray Fabric
##                           350                           157
##                    Oak Finish             Sandstone Fabric
##                            14                            90
##                 Walnut Finish                         White
##                             9                            91
##                     White Dot                    White Plus
##                           184                            78
##                    White Show                    White Spot
##                            85                           109
```

```
#b.
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

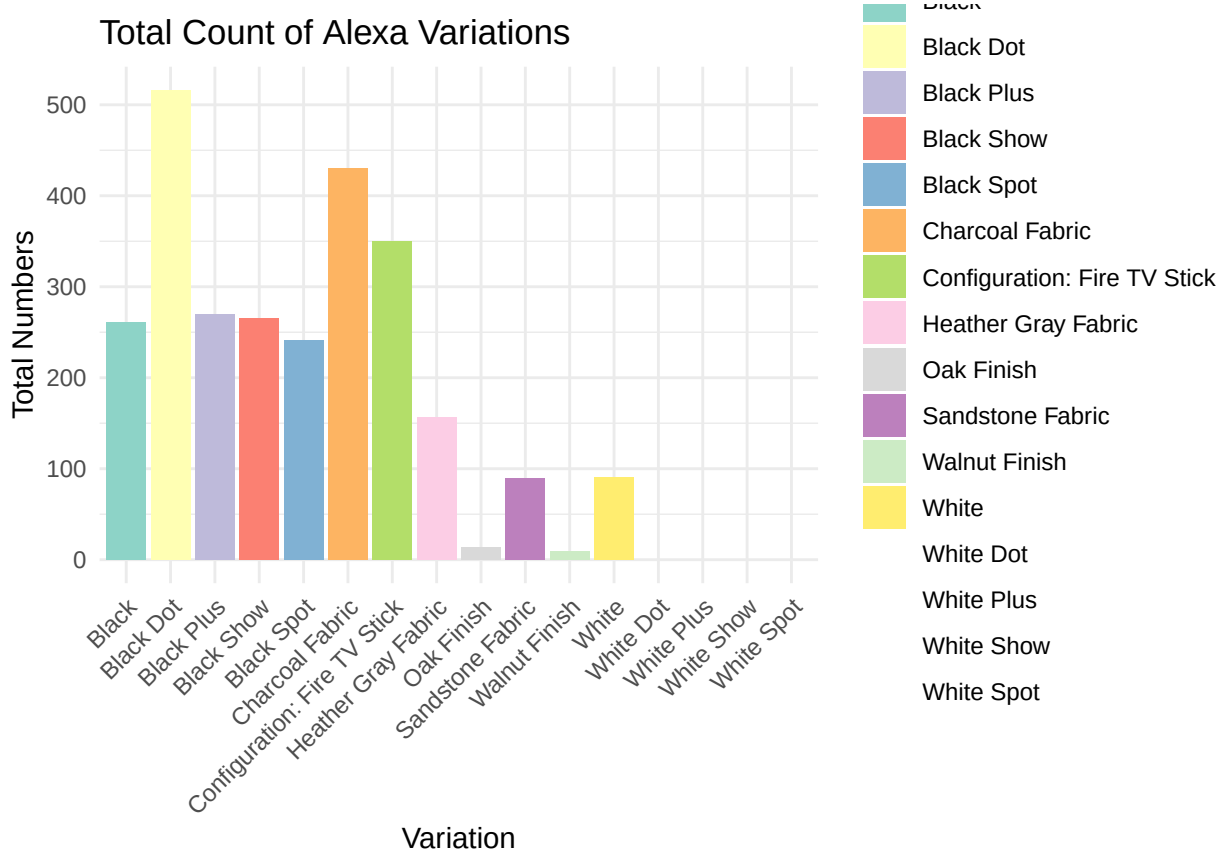
```
## The following objects are masked from 'package:stats':
##
##     filter, lag

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

```
variations_count <- alexa_data %>%
  count(variation, name = "Total")


save(variations_count, file = "variations.RData")


print(variations_count)
```

```
## # A tibble: 16 x 2
##    variation                    Total
##    <chr>                        <int>
##  1 Black                          261
##  2 Black Dot                      516
##  3 Black Plus                     270
##  4 Black Show                     265
##  5 Black Spot                     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 White Dot                      184
## 14 White Plus                      78
## 15 White Show                      85
## 16 White Spot                     109
```

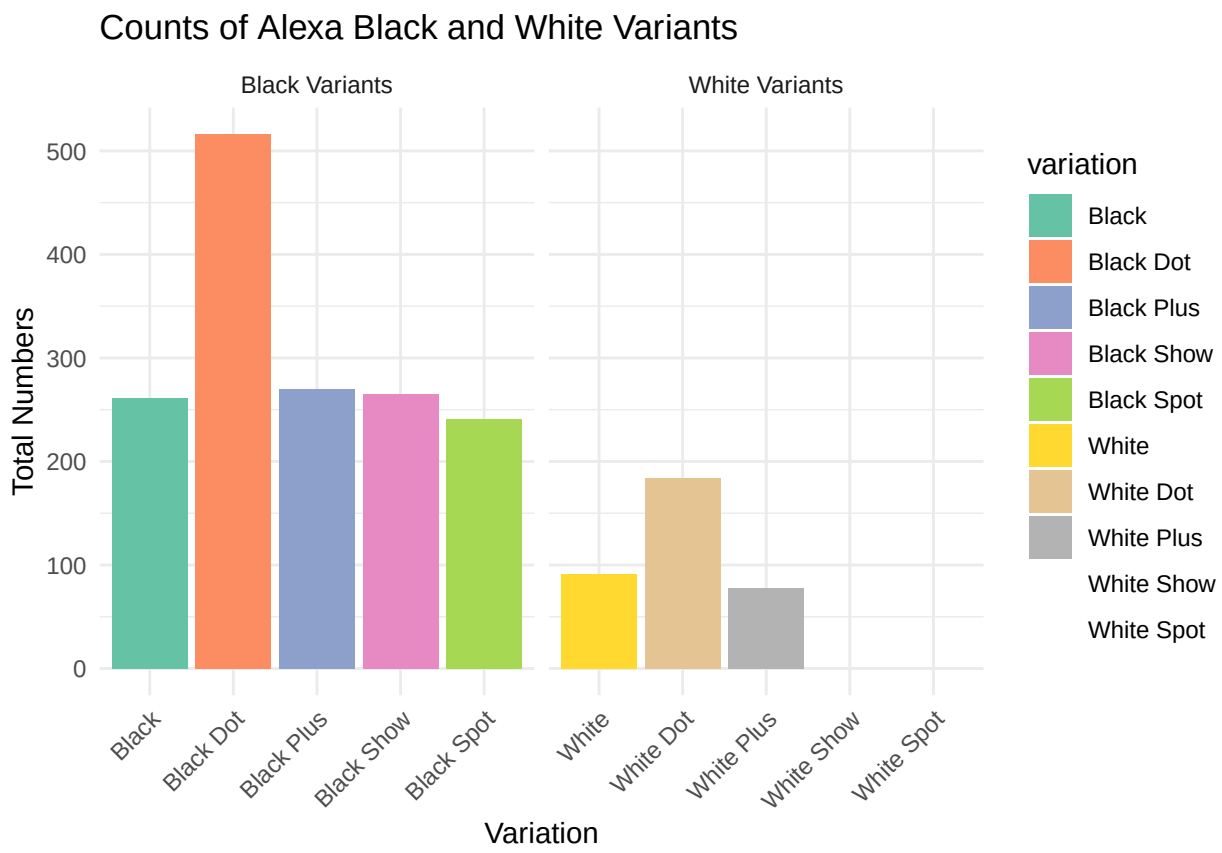
```
#c.
library(ggplot2)

load("variations.RData")


ggplot(variations_count, aes(x = variation, y = Total, fill = variation)) +
  geom_bar(stat = "identity") +
  ggtitle("Total Count of Alexa Variations") +
  xlab("Variation") +
  ylab("Total Numbers") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  scale_fill_brewer(palette = "Set3")
```

```
## Warning in RColorBrewer::brewer.pal(n, pal): n too large, allowed maximum for palette Set3 is 12
## Returning the palette you asked for with that many colors
```




```
#d.
variations_count$Category <- ifelse(grepl("Black", variations_count$variation), "Black Variants",
                                    ifelse(grepl("White", variations_count$variation), "White Variants"

black_white_variants <- variations_count %>% filter(!is.na(Category))


ggplot(black_white_variants, aes(x = variation, y = Total, fill = variation)) +
  geom_bar(stat = "identity") +
  facet_wrap(~ Category, scales = "free_x") +
  ggtitle("Counts of Alexa Black and White Variants") +
  xlab("Variation") +
  ylab("Total Numbers") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  scale_fill_brewer(palette = "Set2")
```

```
## Warning in RColorBrewer::brewer.pal(n, pal): n too large, allowed maximum for palette Set2 is 8
## Returning the palette you asked for with that many colors
```


Counts of Alexa Black and White Variants
Black Variants
White Variants
Total Numbers
500
400
300
200
100
0
Black
Black Dot
Black Plus
Black Show
Black Spot
White
White Dot
White Plus
White Show
White Spot
Variation
variation
Black
Black Dot
Black Plus
Black Show
Black Spot
White
White Dot
White Plus
White Show
White Spot