worksheet#4

Dafnie Jayne Benajiba

2024-10-29

1.

```
vectorA <- c(1, 2, 3, 4, 5)

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

for (i in 0:4) {
  for (j in 0:4) {
    matrix_result[i + 1, j + 1] <- abs(i - j)
  }
}

print(matrix_result)
```

```
##      [,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
```

2.

```
for (i in 1:5) {
  output <- paste(rep('"*"', i), collapse = " ")
  cat(output, "\n")
}
```

```
## "*"
## "*" "*"
## "*" "*" "*"
## "*" "*" "*" "*"
## "*" "*" "*" "*" "*"
```

4-A.

```
data <- read.csv("~/RBasics/CS101_DataScience/worksheet4b/Shoesize.csv")
head(data)
```

```
##   Shoe.size Height Gender Shoe.size.1 Height.1 Gender.1
## 1       6.5   66.0      F        13.0       77        M
## 2       9.0   68.0      F        11.5       72        M
## 3       8.5   64.5      F         8.5       59        F
## 4       8.5   65.0      F         5.0       62        F
## 5      10.5   70.0      M        10.0       72        M
## 6       7.0   64.0      F         6.5       66        F
```

4-B

```r
males <- subset(data, Gender == 'M')
females <- subset(data, Gender == 'F')
nrow(males)
```

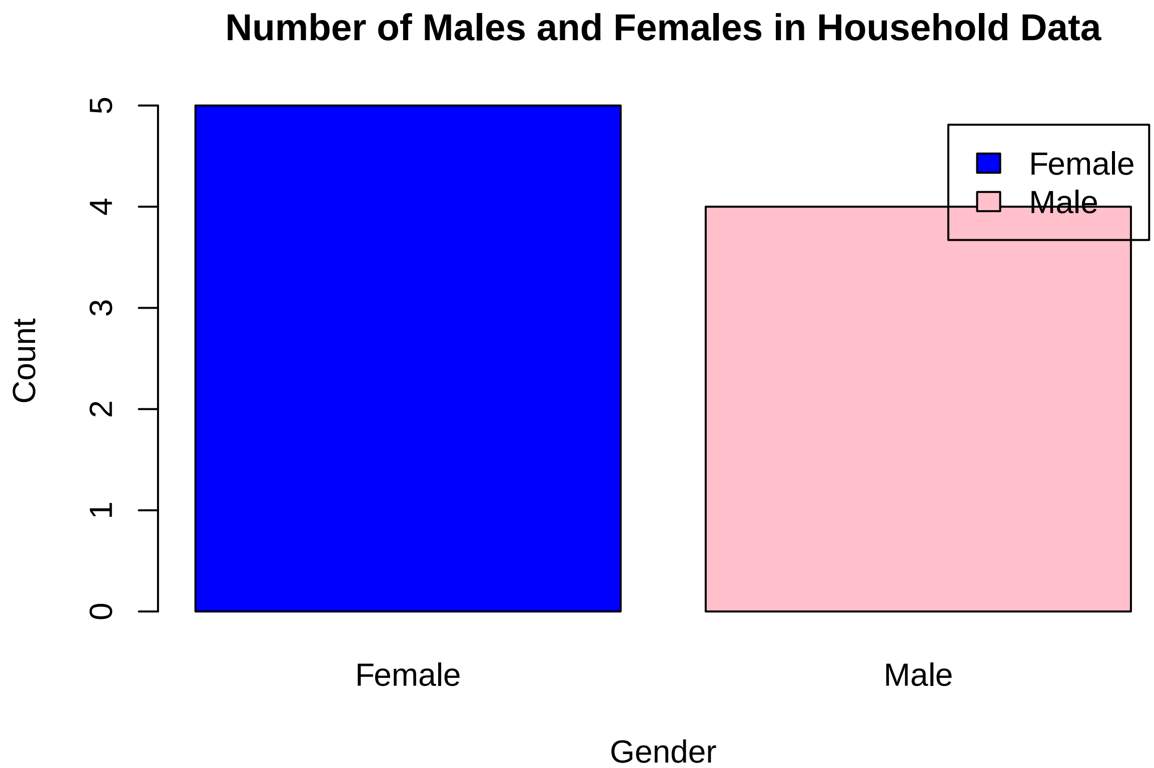
```
## [1] 5
```

```r
nrow(females)
```

```
## [1] 9
```

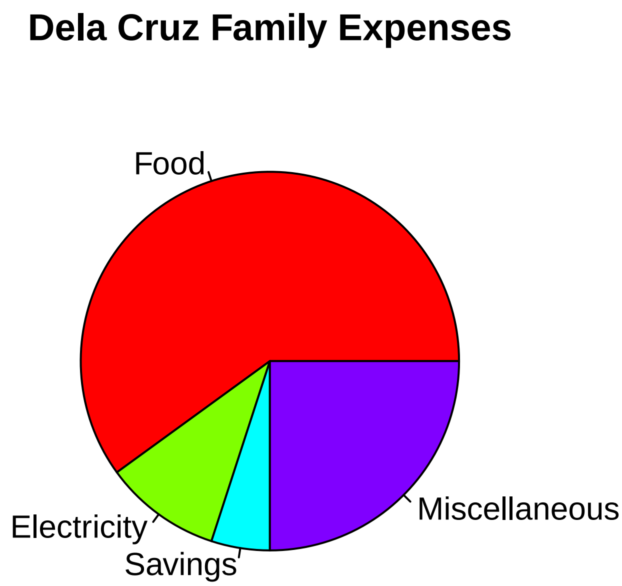
4-C

```r
household_data <- data.frame(
  gender = c("Male", "Female", "Male", "Female", "Male", "Female", "Male", "Female", "Female")
)

gender_counts <- table(household_data$gender)

barplot(
  gender_counts,
  main = "Number of Males and Females in Household Data",
  col = c("blue", "pink"),
  legend = rownames(gender_counts),
  xlab = "Gender",
  ylab = "Count"
)
```



5.

```
expenses <- c(60, 10, 5, 25)
labels <- c("Food", "Electricity", "Savings", "Miscellaneous")
pie(expenses, labels = labels, main = "Dela Cruz Family Expenses", col=rainbow(length(expenses)))
```



6-A

```
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 ...
```

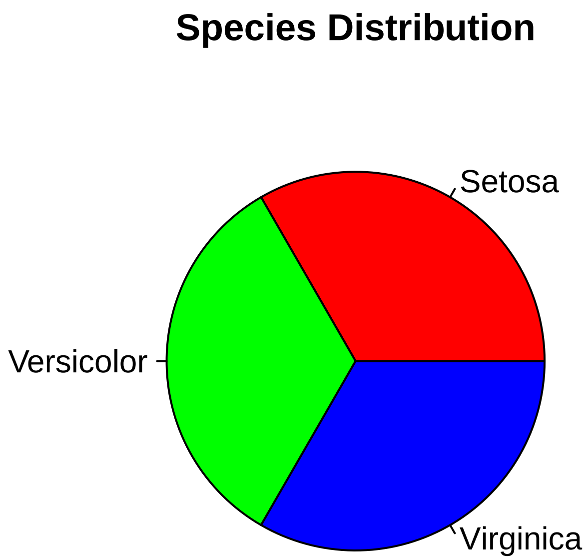
6-B

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

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

6-C

```
species_counts <- table(iris$Species)
pie(species_counts, main="Species Distribution", col=rainbow(3), labels=c("Setosa", "Versicolor", "Virg
```



6-D

```r
setosa <- subset(iris, Species == "setosa")
versicolor <- subset(iris, Species == "versicolor")
virginica <- subset(iris, Species == "virginica")
tail(setosa)
```

```
##    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)
```

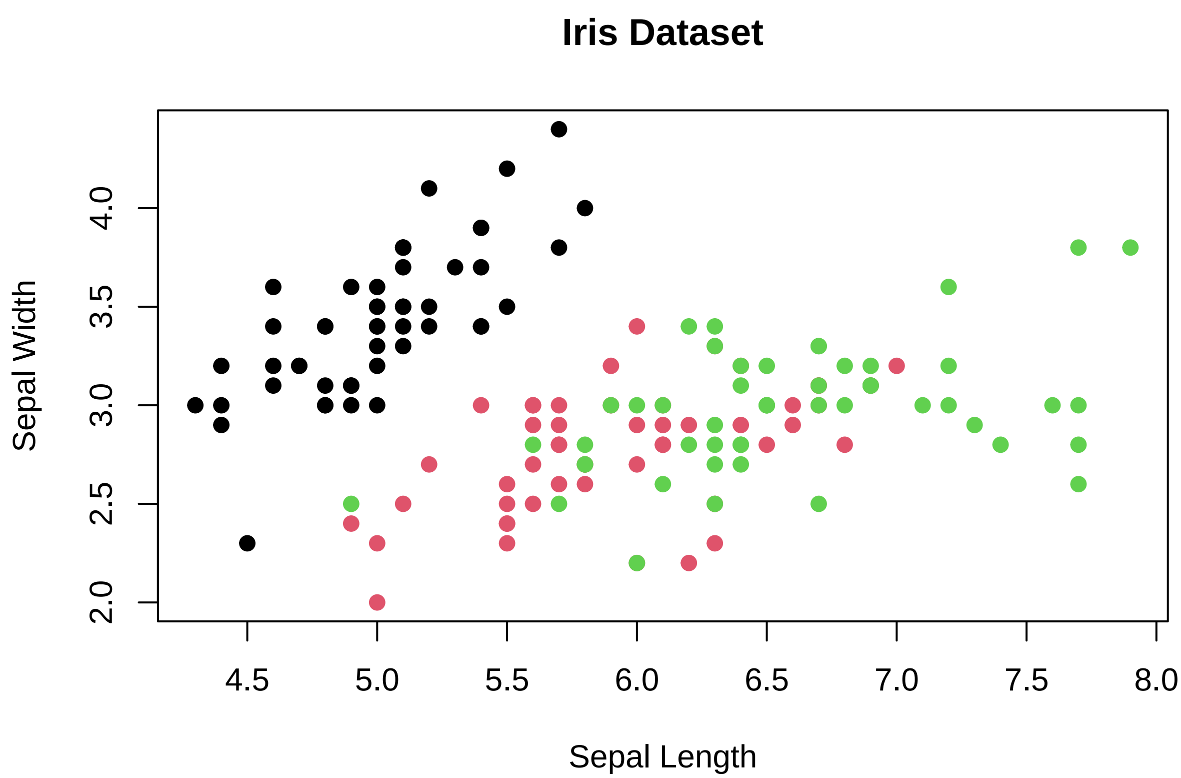
```
##     Sepal.Length Sepal.Width Petal.Length Petal.Width    Species
## 95           5.6         2.7          4.2         1.3 versicolor
## 96           5.7         3.0          4.2         1.2 versicolor
## 97           5.7         2.9          4.2         1.3 versicolor
## 98           6.2         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)
```

```
##     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
```

6-E

```r
plot(iris$Sepal.Length, iris$Sepal.Width, col=iris$Species, pch=19, main="Iris Dataset", xlab="Sepal Le
```



6-F The scatterplot shows clear separation of setosa due to its shorter sepal length and wider width, while versicolor and virginica overlap, making them harder to distinguish. Setosa is the most distinct, with more variability seen between versicolor and virginica.

7. A

```r
library(readxl)
data <- read_excel("~/RBasics/CS101_DataScience/worksheet4b/alexa_file.xlsx")

data$variation <- gsub("Black  Dot", "Black Dot", data$variation)
```

```r
data$variation <- gsub("White  Plus", "White Plus", data$variation)

head(data$variation)
```

```
## [1] "Charcoal Fabric"     "Charcoal Fabric"     "Walnut Finish"
## [4] "Charcoal Fabric"     "Charcoal Fabric"     "Heather Gray Fabric"
```

7-B

```r
library(dplyr)
```

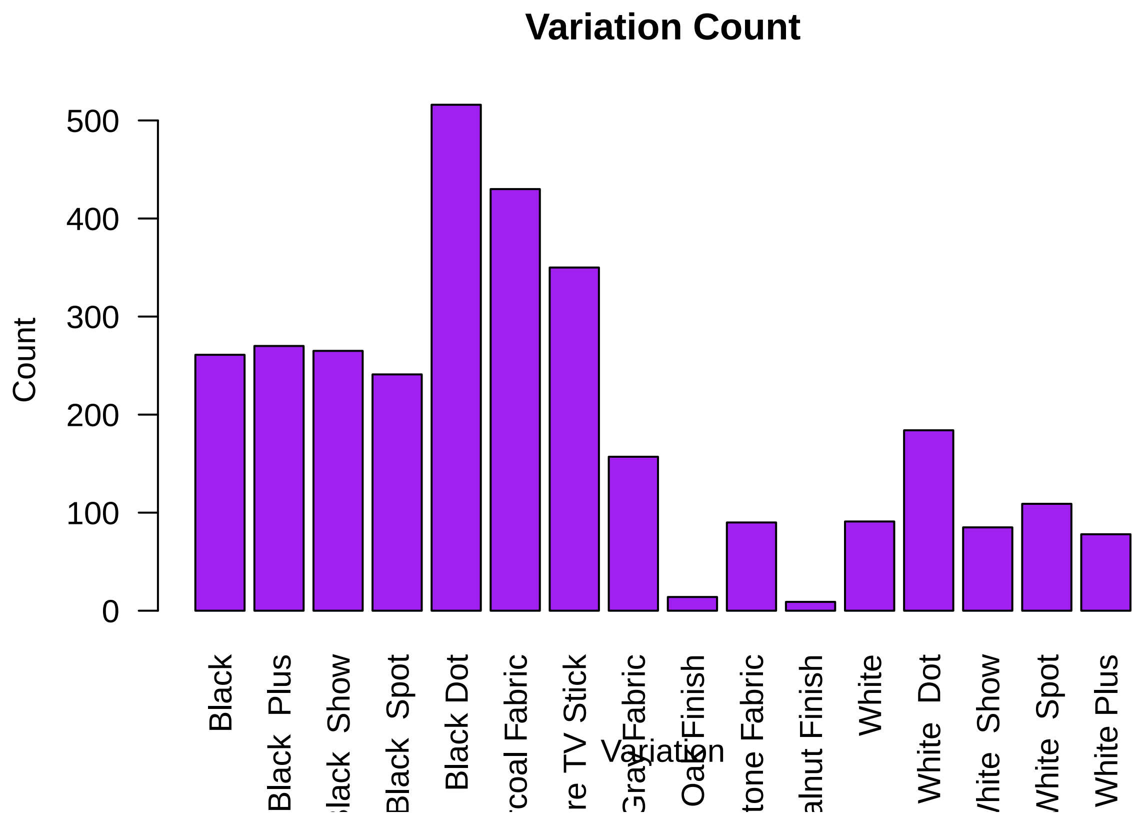
```
##
## Attaching package: 'dplyr'

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

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

```r
variation_count <- data %>%
  count(variation)
save(variation_count, file = "variations.RData")
print(variation_count)
```

```
## # A tibble: 16 x 2
##    variation                        n
##    <chr>                        <int>
##  1 Black                          261
##  2 Black  Plus                    270
##  3 Black  Show                    265
##  4 Black  Spot                    241
##  5 Black Dot                      516
##  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  Show                     85
## 15 White  Spot                    109
## 16 White Plus                      78
```

7b.

```r
library(dplyr)
load("variations.RData")

barplot(variation_count$n, names.arg = variation_count$variation, col = "purple",
        main = "Variation Count", xlab = "Variation", ylab = "Count", las = 2)
```



7

```r
library(readxl)

library(dplyr)

alexa_file <- "~/RBasics/CS101_DataScience/worksheet4b/alexa_file.xlsx"
data <- read_excel(alexa_file)

data$variation <- gsub("Black  Dot", "Black Dot", data$variation)
data$variation <- gsub("Black  Plus", "Black Plus", data$variation)
data$variation <- gsub("Black  Show", "Black Show", data$variation)
data$variation <- gsub("Black  Spot", "Black Spot", data$variation)
data$variation <- gsub("White  Dot", "White Dot", data$variation)
data$variation <- gsub("White  Plus", "White Plus", data$variation)
data$variation <- gsub("White  Show", "White Show", data$variation)
data$variation <- gsub("White  Spot", "White Spot", data$variation)

black_white_data <- data %>%
  filter(grepl("Black|White", variation))
```

```r
variation_count <- black_white_data %>%
  count(variation)

black_variations <- variation_count %>% filter(grepl("Black", variation))
white_variations <- variation_count %>% filter(grepl("White", variation))

black_counts <- setNames(black_variations$n, gsub("Black ", "", black_variations$variation))
white_counts <- setNames(white_variations$n, gsub("White ", "", white_variations$variation))

common_variations <- c("Dot", "Plus", "Show", "Spot")

black_counts <- black_counts[common_variations]
black_counts[is.na(black_counts)] <- 0

white_counts <- white_counts[common_variations]
white_counts[is.na(white_counts)] <- 0

barplot_matrix <- rbind(black_counts, white_counts)

barplot(barplot_matrix, beside = TRUE, col = c("yellow", "pink"),
        main = "Count of Black and White Variations",
        xlab = "Variation Type", ylab = "Count",
        legend = rownames(barplot_matrix), args.legend = list(x = "topright"))
```