

RWorksheet_Delatina#4b

Angel

2024-10-28

#1

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

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

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

for (i in 1:5) {
  for (j in 1:5) {
    final_matrix[i, j] <- abs(vector[i] - vector[j])
  }
}

final_matrix
```

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

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

#3

```

generate_fibonacci <- function(start) {
  fib1 <- 0
  fib2 <- 1
  fib_seq <- c()

  repeat {
    fib_next <- fib1 + fib2

    fib1 <- fib2
    fib2 <- fib_next

    if (fib_next > 500) {
      break
    }

    if (!is.na(fib_next) && fib_next >= start) {
      fib_seq <- c(fib_seq, fib_next)
    }
  }

  return(fib_seq)
}

start_value <- as.numeric(readline(prompt = "Enter a starting number for the Fibonacci sequence: "))

```

Enter a starting number for the Fibonacci sequence:

```

if (is.na(start_value) || start_value < 0) {
  cat("Please enter a valid non-negative number.\n")
} else {

  fibonacci_sequence <- generate_fibonacci(start_value)

  cat("Fibonacci sequence starting from", start_value, "up to 500:\n")
  print(fibonacci_sequence)
}

```

Please enter a valid non-negative number.

#4

```

data<- read.csv("C:/Documents/RBasics/RWorksheet_Delatina#4B/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

```

```
data <- read.csv("C:/Documents/RBasics/RWorksheet_Delatina#4B/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
```

#4b

```
male_data <- subset(data, Gender == "M")
female_data <- subset(data, Gender == "F")

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

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

```
## Number of Male observations: 5
```

```
cat("Number of Female observations:", num_females, "\n")
```

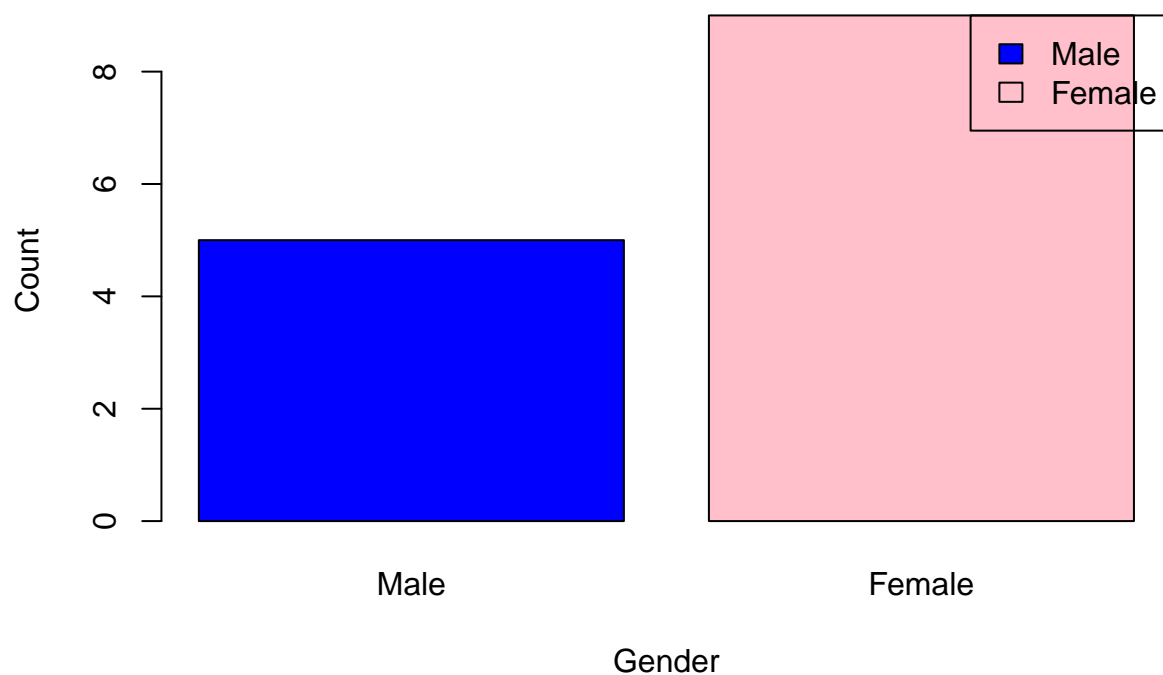
```
## Number of Female observations: 9
```

#4c

```
gender_counts <- c(Male = num_males, Female = num_females)

barplot(gender_counts,
        main = "Number of Males and Females in Household Data",
        xlab = "Gender",
        ylab = "Count",
        col = c("blue", "pink"),
        legend = names(gender_counts),
        args.legend = list(x = "topright"))
```

Number of Males and Females in Household Data

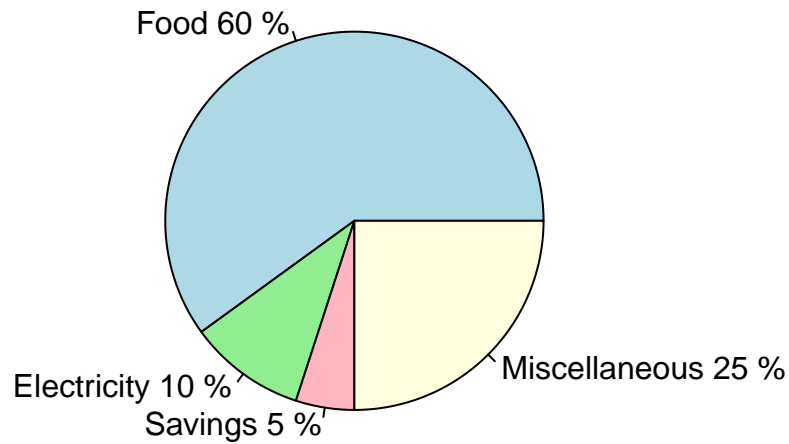


#5

```
expenses <- c(60, 10, 5, 25)
labels <- c("Food", "Electricity", "Savings", "Miscellaneous")
colors <- c("lightblue", "lightgreen", "lightpink", "lightyellow")

pie(expenses,
    labels = paste(labels, round(expenses/sum(expenses) * 100, 1), "%"),
    col = colors,
    main = "Dela Cruz Family Monthly Income Distribution")
```

Dela Cruz Family Monthly Income Distribution



#6

```
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 ...
```

#6b

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

```
means
```

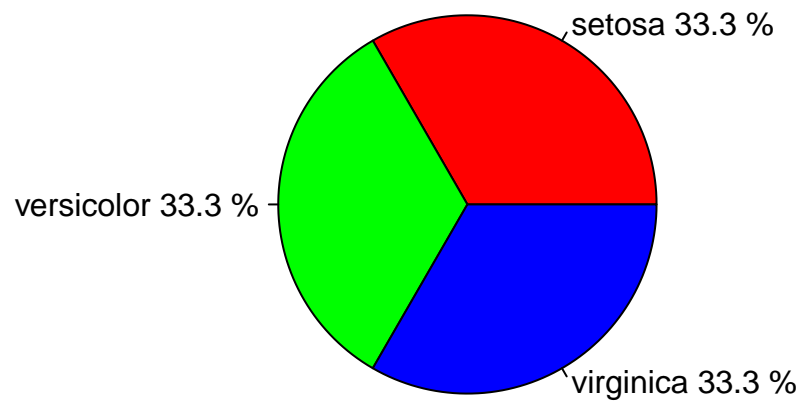
```
## Sepal.Length Sepal.Width Petal.Length Petal.Width
## 5.843333 3.057333 3.758000 1.199333
```

#6c

```
species_count <- table(iris$Species)
colors <- c("red", "green", "blue")

pie(species_count,
     labels = paste(names(species_count), round(species_count/sum(species_count) * 100, 1), "%"),
     col = colors,
     main = "Species Distribution in Iris Dataset")
```

Species Distribution in Iris Dataset



#6d

```
setosa <- iris[iris$Species == "setosa", ]
versicolor <- iris[iris$Species == "versicolor", ]
virginica <- iris[iris$Species == "virginica", ]

last_rows_setosa <- tail(setosa, 6)
last_rows_versicolor <- tail(versicolor, 6)
last_rows_virginica <- tail(virginica, 6)

list(Setosa = last_rows_setosa, Versicolor = last_rows_versicolor, Virginica = last_rows_virginica)
```

```
## $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
##
## $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
##
## $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
```

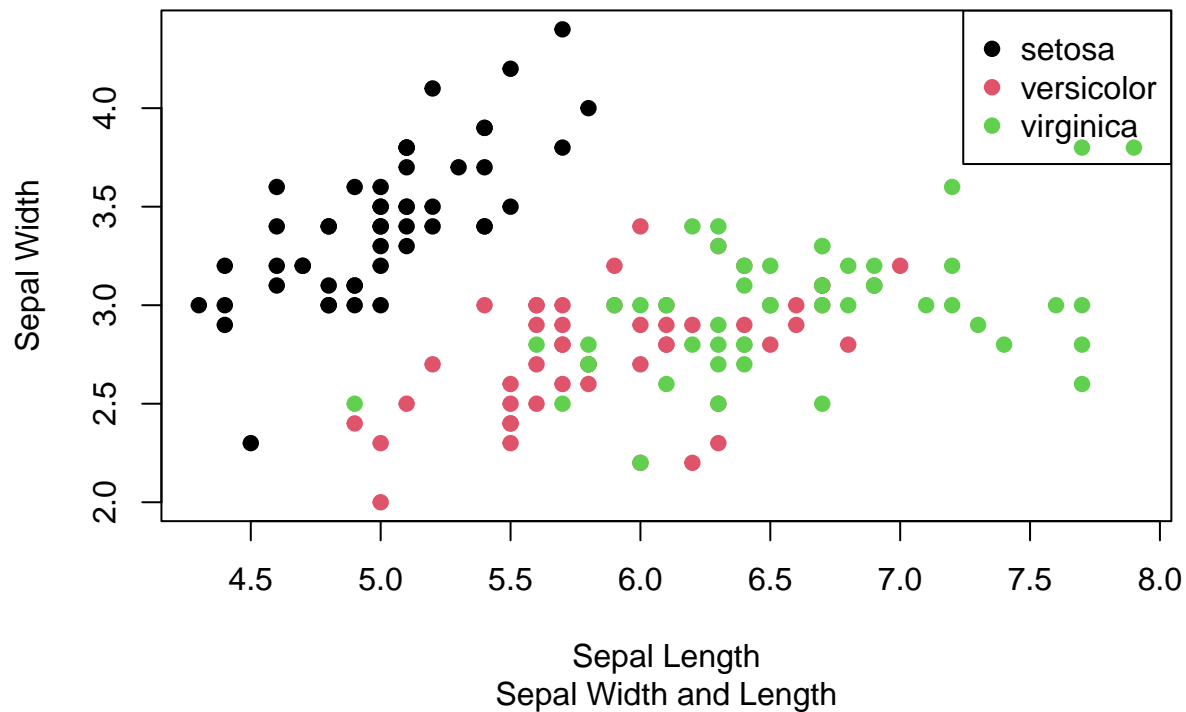
#6e

```
iris$Species <- as.factor(iris$Species)

plot(iris$Sepal.Length, iris$Sepal.Width,
     col = iris$Species, pch = 19,
     main = "Iris Dataset",
     sub = "Sepal Width and Length",
     xlab = "Sepal Length",
     ylab = "Sepal Width")

legend("topright", legend = levels(iris$Species), col = 1:3, pch = 19)
```

Iris Dataset



#7

```
library(readxl)
data <- read_excel("C:/Documents/RBasics/RWorksheet_Delatina#4B/alexa_file.xlsx")

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

```
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
```


#7b

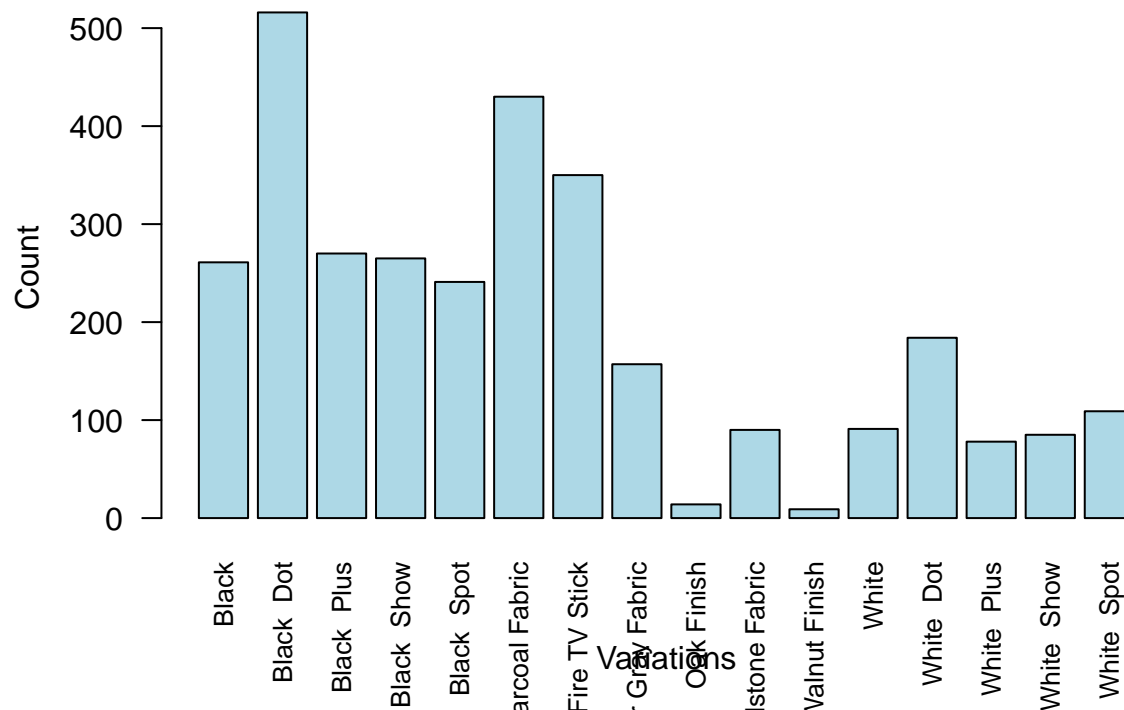
```
variation_count <- data%>%  
  count(variation)  
save(variation_count, file = "variations.RData")  
print(variation_count)
```

```
## # A tibble: 16 x 2  
##   variation      n  
##   <chr>      <int>  
## 1 Black      261  
## 2 Black 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
```

#7c

```
load("variations.RData")  
  
barplot(variation_count$n,  
  names.arg = variation_count$variation,  
  col = "lightblue",  
  main = "Variation Counts in Alexa Data",  
  xlab = "Variations",  
  ylab = "Count",  
  las = 2,  
  cex.names = 0.8)
```

Variation Counts in Alexa Data



#7d

```
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("purple", "blue"),
        main = "Count of Black and White Variations",
        xlab = "Variation Type", ylab = "Count",
        legend = rownames(barplot_matrix), args.legend = list(x = "topright"))
```

