RWorksheet_MAMINTA#4b.Rmd

Ansarah Juliana Maminta

2025-11-19

```r
#Using Loop Function
#for() loop
#1. Using the for loop, create an R script that will display a 5x5 matrix as shown in
#Figure 1. It must contain vectorA = [1,2,3,4,5] and a 5 x 5 zero matrix.
#Hint Use abs() function to get the absolute value

vec1 <- c(1, 2, 3, 4, 5)

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


for (i in 1:5) {
for (j in 1:5) {
Mtrix[i, j] <- abs(i - j) + 1
}
}
print(Mtrix)
```

```
##      [,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
```

```r
#2. Print the string "*" using for() function. The output should be the same as shown
#in Figure

for (i in 1:5) {
line <- ""
for (j in 1:i) {
line <- paste(line, "*")
}
print(line)
}
```

```
## [1] " *"
## [1] " * *"
## [1] " * * *"
## [1] " * * * *"
## [1] " * * * * *"
```

```r
#3. Get an input from the user to print the Fibonacci sequence starting from the 1st input
#up to 500. Use repeat and break statements. Write the R Scripts and its output.
input <- readline("Enter starting Fibonacci number: ")
```

Enter starting Fibonacci number:

```r
if (input == "") {
  start_val <- 10    # default value when knitting
} else {
  start_val <- as.numeric(input)
}

if (is.na(start_val)) {
  stop("Invalid input. Please enter a numeric value.")
}


a <- 0
b <- 1

repeat {
  fib <- a

  if (fib >= start_val) {
    print(fib)
  }

  if (fib > 500) {
    break
  }

  temp <- a + b
  a <- b
  b <- temp
}
```

```
## [1] 13
## [1] 21
## [1] 34
## [1] 55
## [1] 89
## [1] 144
## [1] 233
## [1] 377
## [1] 610
```

```r
#4. Import the dataset as shown in Figure 1 you have created previously.
#if csv
data <- read.csv("Householddata.csv")
head(data)
```

```
##   Respondents Sex Fathers_Occupation Persons_at_Home Siblings_at_School
```

```
## 1          1   2                1          5             6
## 2          2   2                3          7             4
## 3          3   1                3          3             4
## 4          4   2                3          8             1
## 5          5   2                1          5             2
## 6          6   2                2          9             1
##    Type_of_Houses
## 1             1
## 2             2
## 3             3
## 4             1
## 5             1
## 6             3
```

```r
#if excel
library(readxl)
data <- read_excel("Householddata.xlsx")
head(data)
```

```
## # A tibble: 6 x 6
##    Respondents   Sex Fathers_Occupation Persons_at_Home Siblings_at_School
##          <dbl> <dbl>              <dbl>           <dbl>              <dbl>
## 1            1     2                  1               5                  6
## 2            2     2                  3               7                  4
## 3            3     1                  3               3                  4
## 4            4     2                  3               8                  1
## 5            5     2                  1               5                  2
## 6            6     2                  2               9                  1
## # i 1 more variable: Type_of_Houses <dbl>
```

```r
#b. Create a subset for gender(female and male). How many observations are there in
#Male? How about in Female? Write the R scripts and its output.

female <- subset(data, Sex == "F")
male <- subset(data, Sex == "M")
num_female <- nrow(female)
num_male <- nrow(male)
num_female
```

```
## [1] 0
```

```r
num_male
```

```
## [1] 0
```

```r
#c. Create a graph for the number of males and females for Household Data. Use plot(),
#chart type = barplot. Make sure to place title, legends, and colors. Write the R scripts and
#its result.
counts <- c(num_male, num_female)
barplot(
  counts,
  names.arg = c("Male", "Female"),
```
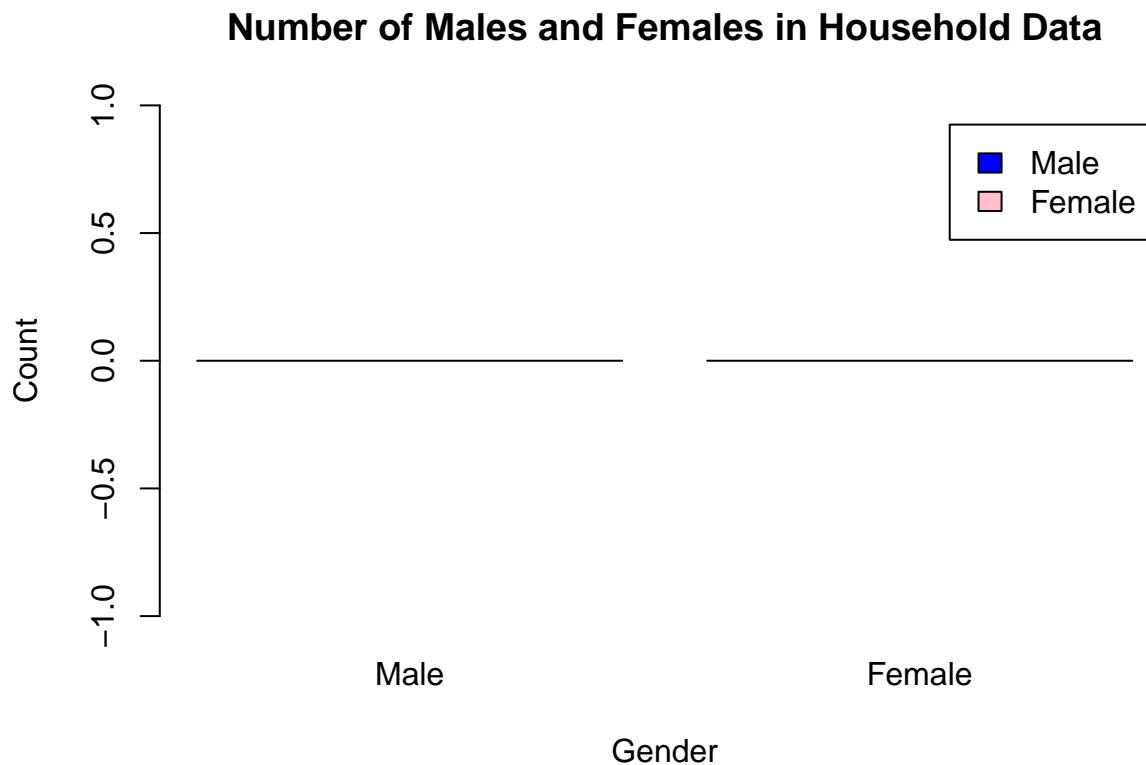
```
  main = "Number of Males and Females in Household Data",
  xlab = "Gender",
  ylab = "Count",
  col = c("blue", "pink"),
  legend.text = c("Male", "Female")
)
```

Number of Males and Females in Household Data
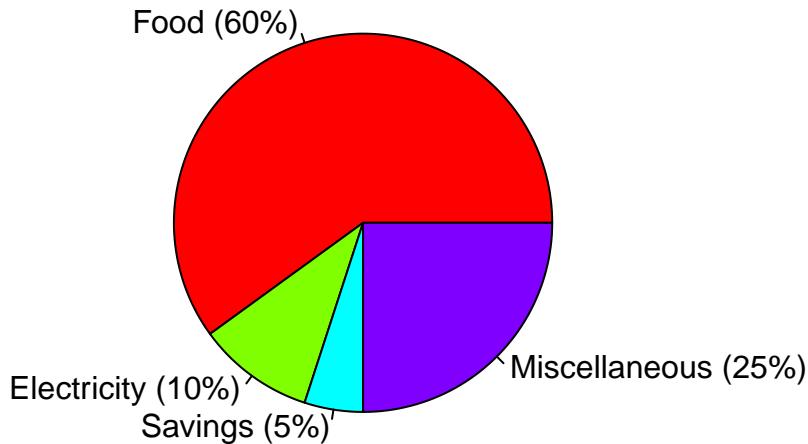
Monthly Expenses of Dela Cruz Family

Food (60%)
Electricity (10%)
Savings (5%)
Miscellaneous (25%)

```r
#6. Use the iris dataset.
data(iris)
#a. Check for the structure of the dataset using the str() function. Describe what you have seen in the
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
#b. Create an R object that will contain the mean of the sepal.length,
#sepal.width,petal.length,and petal.width. What is the R script and its result?
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
#c. Create a pie chart for the Species distribution. Add title, legends, and colors. Write
#the R script and its result.

species_counts <- table(iris$Species)

pie(
  species_counts,
  labels = names(species_counts),
  main = "Distribution of Iris Species",
  col = c("red", "green", "blue")
)
legend(
  "topright",
  legend = names(species_counts),
```
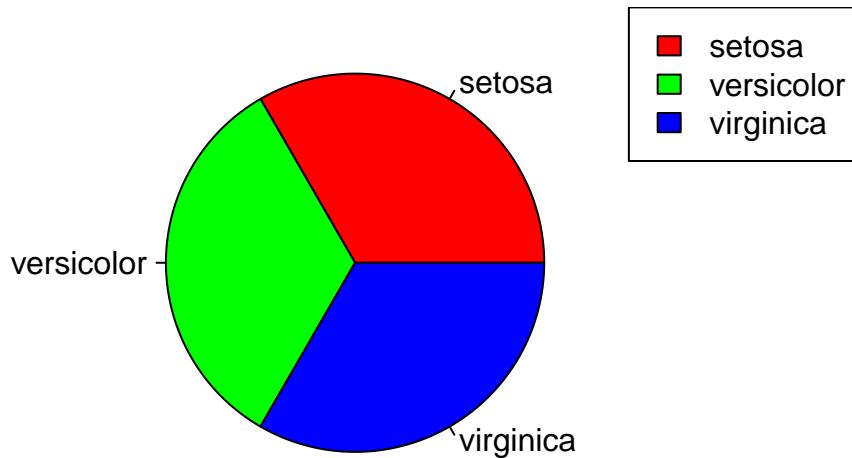
```r
  fill = c("red", "green", "blue")
)
```

Distribution of Iris Species


```r
#d. Subset the species into setosa, versicolor, and virginica. Write the R scripts
#and show the last six (6) rows of each species.
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
```

```r
#e. Create a scatterplot of the sepal.length and sepal.width using the different
#species(setosa,versicolor,virginica). Add a title = "Iris Dataset", subtitle = "Sepal
#width and length, labels for the x and y axis, the pch symbol and colors should be based on
#the species.

iris$Species <- as.factor(iris$Species)
colors <- c("red", "green", "blue")[iris$Species]
plot(
  iris$Sepal.Length,
  iris$Sepal.Width,
  col = colors,
  pch = 19,
  main = "Iris Dataset",
  sub = "Sepal width and length",
  xlab = "Sepal Length",
  ylab = "Sepal Width"
)

legend(
  "topright",
  legend = levels(iris$Species),
  col = c("red", "green", "blue"),
  pch = 19
)
```
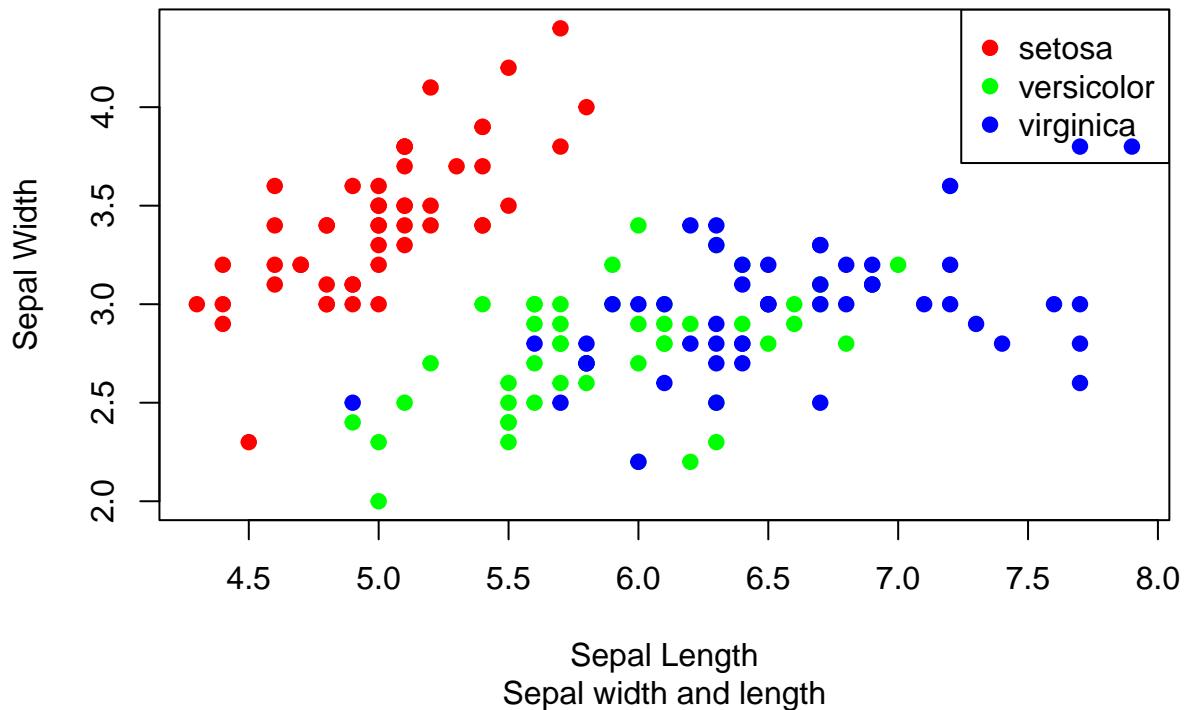
Iris Dataset

Sepal Length
Sepal width and length

```
#Interpretation

#Setosa, shown in red, has the largest sepal widths and the smallest sepal
#lengths, forming a tight and easily identifiable cluster. Versicolor, shown in
#green, has moderate values for both sepal length and sepal width, with slight
#overlap with Virginica. Virginica, shown in blue, has the longest sepal lengths
#and moderate sepal widths. Overall, Setosa is clearly separable from the other
#two species, while Versicolor and Virginica overlap somewhat, although Virginica
#generally has larger measurements. This indicates that sepal measurements
#effectively distinguish Setosa, but are less reliable for separating Versicolor
#from Virginica.


#7. Import the alexa-file.xlsx. Check on the variations. Notice that there are ex-
#tra whitespaces among black variants (Black Dot, Black Plus, Black Show, Black
#Spot). Also on the white variants (White Dot, White Plus, White Show, White Spot).

## THERE IS NO ALEXA-FILE
```