

Rworksheet_cadiz#4b

John Dave R. Cadiz

2024-10-30

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.

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

```
##      [,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. Print the string “*” using `for()` function. The output should be the same as shown in Figure.

```
for ( i in 1:5) {
  for( j in 1:i) {
    cat("* ")
  }
  cat("\n")
}
```

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

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

```
printFibonacci <- function(start){
  first <- 0
  second <- 1
  next_num <- 0

  if (start == 1){
    cat(first, "", second, "", second, " ")
  }

  for (i in 0:start){
    next_num <- first + second
    first <- second
    second <- next_num
  }

  repeat{
    if (next_num > 500) break
    cat(next_num, " ")
    next_num <- first + second
    first <- second
    second <- next_num
  }
}

#start <- readline(prompt = "Enter starting term: ")
start <- 1
printFibonacci(start)

## 0 1 1 2 3 5 8 13 21 34 55 89 144 233 377
```

4. Import the dataset as shown in Figure 1 you have created priviously.
a. What us the R script for importing an excel or a csv file? Display the the first 6 rows of the dataset? Show your codes and its result.

```
shoedata <- read.csv("shoesizes - Sheet1.csv")
shoedata[(1:6), ]

##   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
```

b. Create a subset for gender (female and male). How many observations are there in Male? How about in Female? Show your code and its results.

```
male <- subset(shoedata, Gender == "M")
print(male)
```

```
##      Shoe.size Height Gender Shoe.size.1 Height.1 Gender.1
## 5         10.5   70.0      M         10.0       72         M
## 9         13.0   72.0      M         10.5       73         M
## 11        10.5   74.5      M         10.5       72         M
## 13        12.0   71.0      M          9.0       69         M
## 14        10.5   71.0      M         13.0       70         M
```

```
female <- subset(shoedata, Gender == "F")
print(female)
```

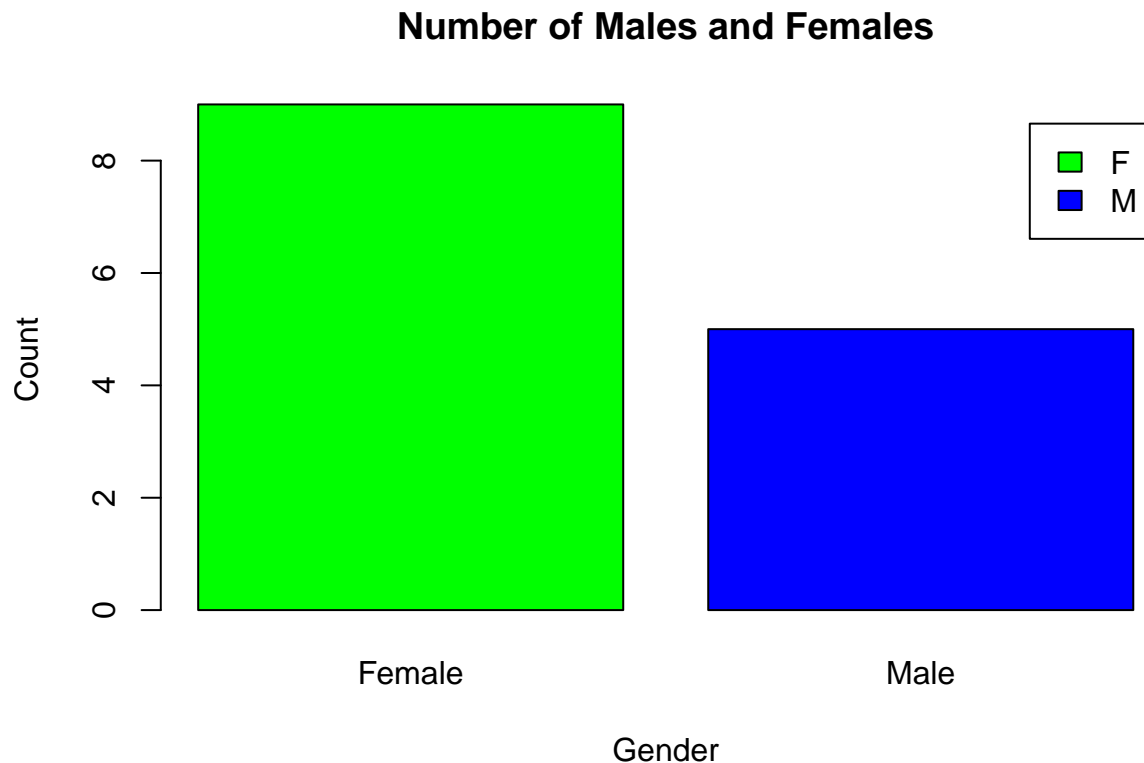
```
##      Shoe.size Height Gender Shoe.size.1 Height.1 Gender.1
## 1          6.5   66.0      F         13.0       77         M
## 2          9.0   68.0      F         11.5       72         M
## 3          8.5   64.5      F          8.5       59         F
## 4          8.5   65.0      F          5.0       62         F
## 6          7.0   64.0      F          6.5       66         F
## 7          9.5   70.0      F          7.5       64         F
## 8          9.0   71.0      F          8.5       67         M
## 10         7.5   64.0      F          8.5       69         F
## 12         8.5   67.0      F         11.0       70         M
```

c. Create a graph for the numbers of male 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.

```
householddata <- table(shoedata$Gender)

barplot(householddata,
        main = "Number of Males and Females",
        xlab = "Gender",
        ylab = "Count",
        col = c("Green", "Blue"),
        names.arg = c("Female", "Male"),
        legend = rownames(householddata))
```



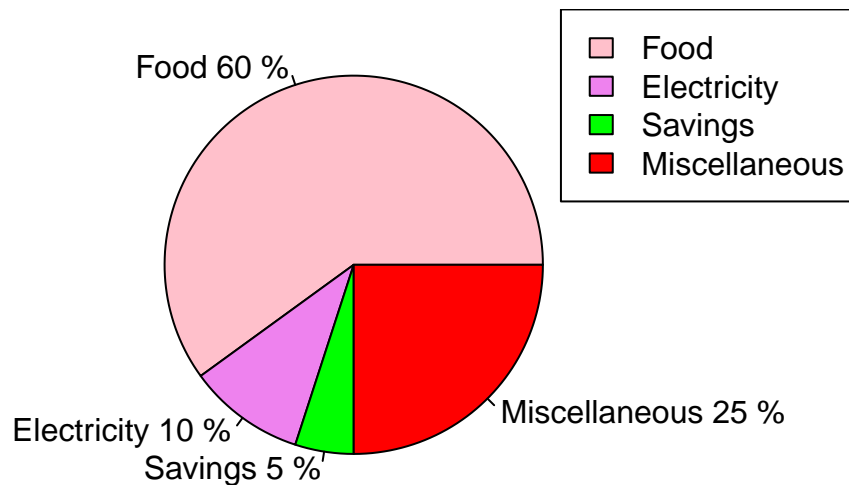
5. The monthly income of Dela Cruz family was spent on the following:
- Create a piechart that will include labels in percentage. Add some colors and title of the chart.. Write the R scripts and show its output.

```
category <- c("Food", "Electricity", "Savings", "Miscellaneous")
value <- c(60, 10, 5, 25 )
color = c("pink", "violet", "green", "red")
percentage <- round(value / sum(value) * 100)
percent_label <- paste(category, percentage, "%")

pie(
  value,
  col = color,
  main = "Dela Cruz Expenses",
  label = percent_label
)

legend("topright", category, fill = color)
```

Dela Cruz Expenses



6. Use the iris dataset.

a. Check for the structure of the dataset using the `str()` function. Describe what you have seen in the output.

```
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 ...
```

the output shows the data frame of the Iris data that contains the length and width of Sepal and Petal

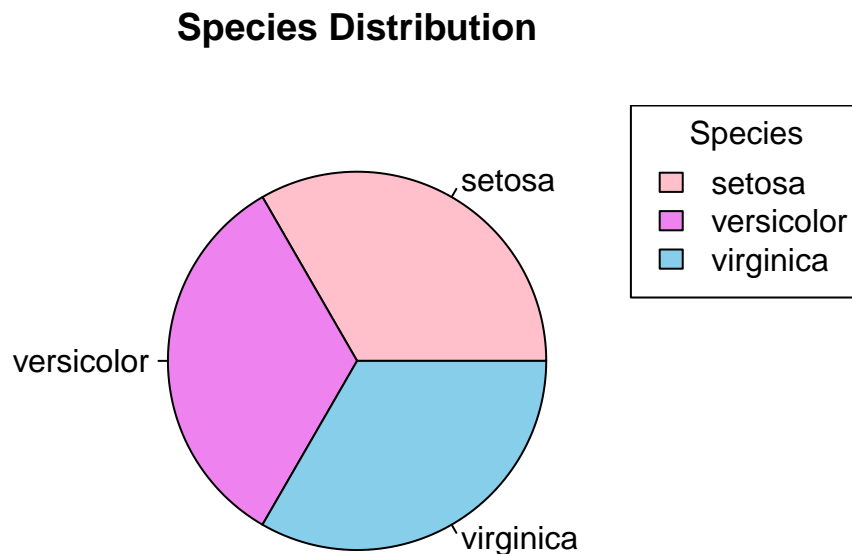
b. Create an R object that will contain the mean of the sepal.length, sepal.width, petal.length, and petal.width. What is the R scripts and its result?

```
datameans <-c(
  Sepal.Length = mean(iris$Sepal.Length),
  Sepal.Width = mean(iris$Sepal.Width),
  Petal.Length = mean(iris$Petal.Length),
  Petal.Width = mean(iris$Petal.Width)
)
print(datameans)
```

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

c. Create a pie chart for the Species distribution. Add titles, legends, and colors. Write the R script and its result.

```
iris_species <- table(iris$Species)
species_color <- c("pink", "violet", "skyblue")
pie(
  iris_species,
  main = "Species Distribution",
  col = species_color
)
legend("topright", names(iris_species), fill = species_color, title = "Species")
```



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
```

```
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
```

```
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
```

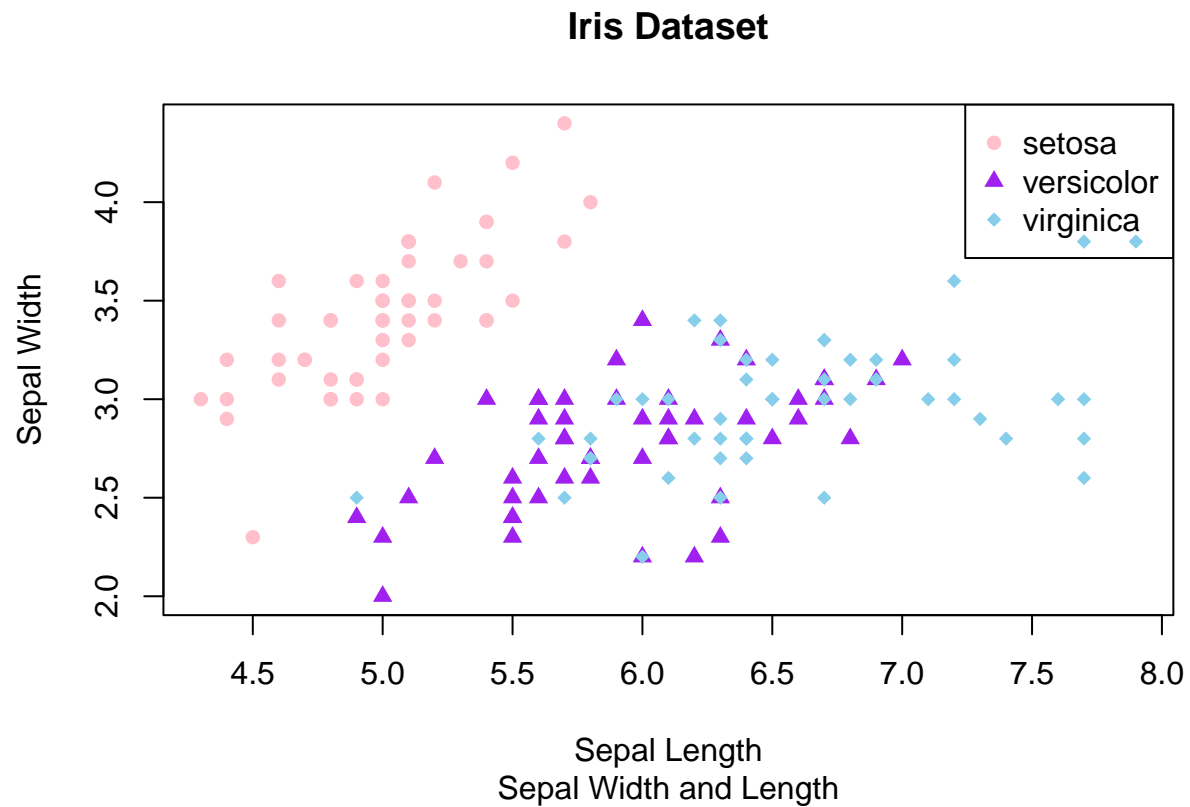
e. Create a scatterplot of the sepal.length and sepal.width using the different species(setosa,versicolor, virginica). Add a title = “Iris Dataset, subtitle = “Sepal width and length, labels for the x and y axis, the pch symbol and colors should be based on the species.

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

colors <- c("pink", "purple", "skyblue")
pch_symbols <- c(16, 17, 18)

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

legend("topright", legend = levels(iris$Species),
     col = colors, pch = pch_symbols)
```



f. Interpret the result.

*#it shows that the scatter plot displays the relationship between Sepal.Length
#and Sepal.Width of the three given species in the iris data.*

*#secondly, the colors helps to track the species which shows in the figure,
#by this we we can see the overlapping relationship of the versicolor and
#virginica mean while, the setosa also portray above the overlapping relationship
#of the two, in this scatterplot helps me to identify that there is a positive
#relationship*