RWorksheet_Camarista#4b

John Lyxton Camarista

2024-10-30

#Using Loop Function ##for() loop 1. 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

```
vectorA <- c(1, 2, 3, 4, 5)
matrixA <- matrix(0, nrow = 5, ncol = 5)

for(i in 1:5){

    matrixA[i, ] <- abs(vectorA - i)
}
matrixA
```

```
##      [,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){

    cat(rep("*", i), "\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.

```
#There's an error during knitting in this particular chunk. So I just commented the prompt script an in
#start <- as.integer(readline(prompt = "Enter the starting number: "))
start <- 4
```

```
a <- start
b <- start + 1

cat(a, "\n")
```

```
## 4
```

```
repeat {
  cat(b, "\n")

  next_term <- a + b

  if (next_term > 500) {
    break
  }

  a <- b
  b <- next_term
}
```

```
## 5
## 9
## 14
## 23
## 37
## 60
## 97
## 157
## 254
## 411
```

##Using Basic Graphics (plot(),barplot(),pie(),hist()) 4. Import the dataset as shown in Figure 1 you have created previously. - a. What is the R script for importing an excel or a csv file? Display the first 6 rows of the dataset? Show your codes and its result

```
ShoeSize_Data <- read.csv("ShoeSize.csv")
head(ShoeSize_Data)
```

```
##   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
```

```
- b. Create a subset for gender(female and male). How many observations are there in Male? How about in
```

```
male_sub <- subset(ShoeSize_Data, gender == "M", select = gender)
male_count <- nrow(male_sub)
print(paste("There are", male_count, "males"))
```

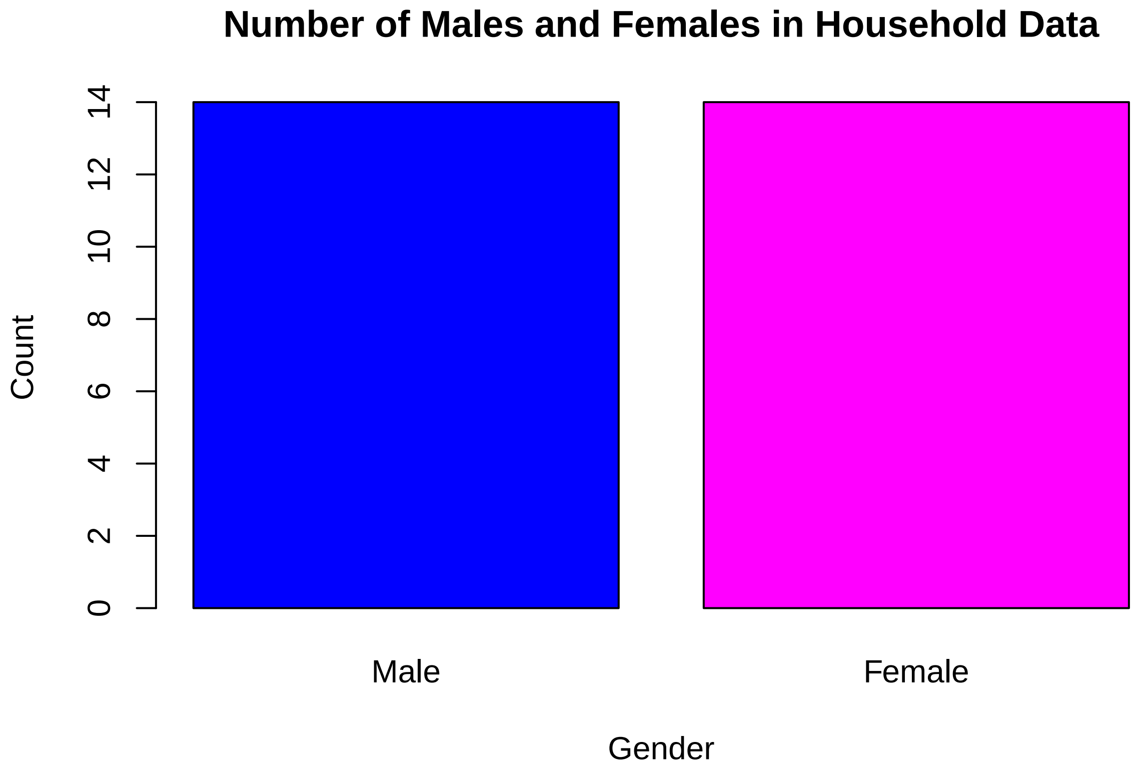
```
## [1] "There are 14 males"
```

```
female_sub <- subset(ShoeSize_Data, gender == "F", select = gender)
female_count <- nrow(female_sub)
print(paste("There are", female_count, "females"))
```

```
## [1] "There are 14 females"
```

```
- c.Create a graph for the number of males and females for Household Data. Use plot(), chart type = bar
```

```
gender_counts <- c(male_count, female_count)
gender_labels <- c("Male", "Female")
barplot(
  gender_counts,
  names.arg = gender_labels,
  main = "Number of Males and Females in Household Data",
  xlab = "Gender",
  ylab = "Count",
  col = c("blue", "#FF00FF")
)
```

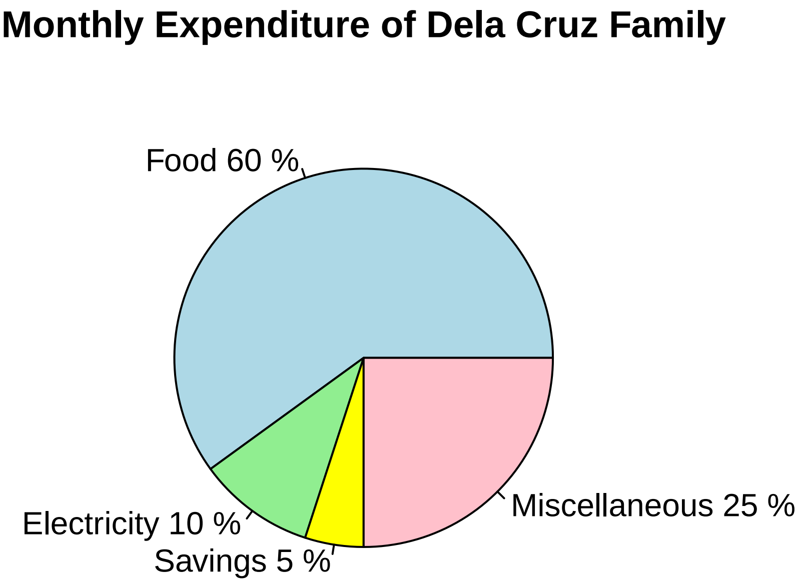


##5. The monthly income of Dela Cruz family was spent on the following: - a. Create a piechart that will include labels in percentage.Add some colors and title of the chart. Write the R scripts and show its output.

```r
expenses <- c(Food = 60, Electricity = 10, Savings = 5, Miscellaneous = 25)

percent_labels <- paste(names(expenses), round(expenses / sum(expenses) * 100, 1), "%")

pie(
  expenses,
  labels = percent_labels,
  main = "Monthly Expenditure of Dela Cruz Family",
  col = c("lightblue", "lightgreen", "yellow", "pink")
)
```



##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.

```r
# Load the iris dataset
data(iris)

# Check the structure of the dataset
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. 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?

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

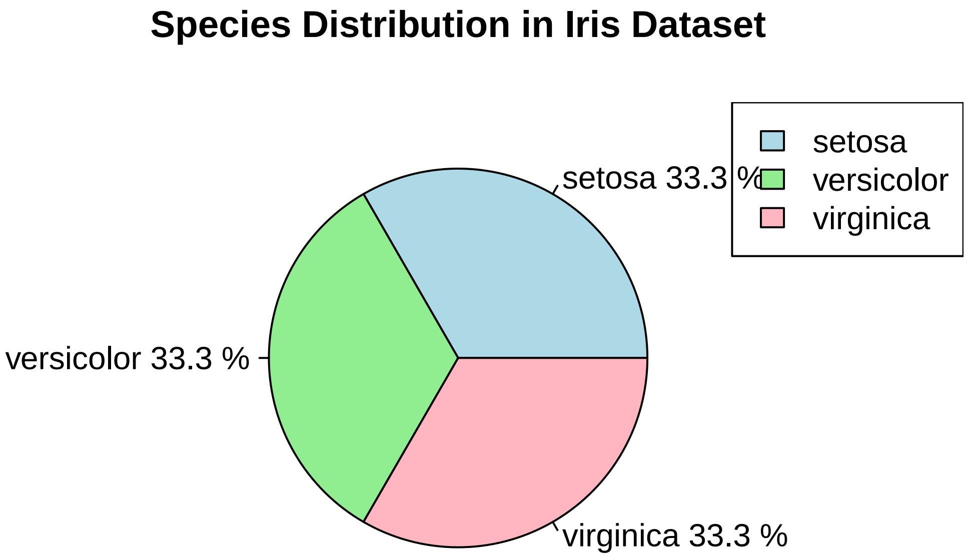
```
## Sepal.Length  Sepal.Width Petal.Length  Petal.Width 
##     5.843333     3.057333     3.758000     1.199333 
```

- 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 = paste(names(species_counts), round(species_counts / sum(species_counts) * 100, 1), "%"),
  main = "Species Distribution in Iris Dataset",
  col = c("lightblue", "lightgreen", "lightpink")
)

legend("topright", legend = names(species_counts), fill = c("lightblue", "lightgreen", "lightpink"))
```



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

```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
```

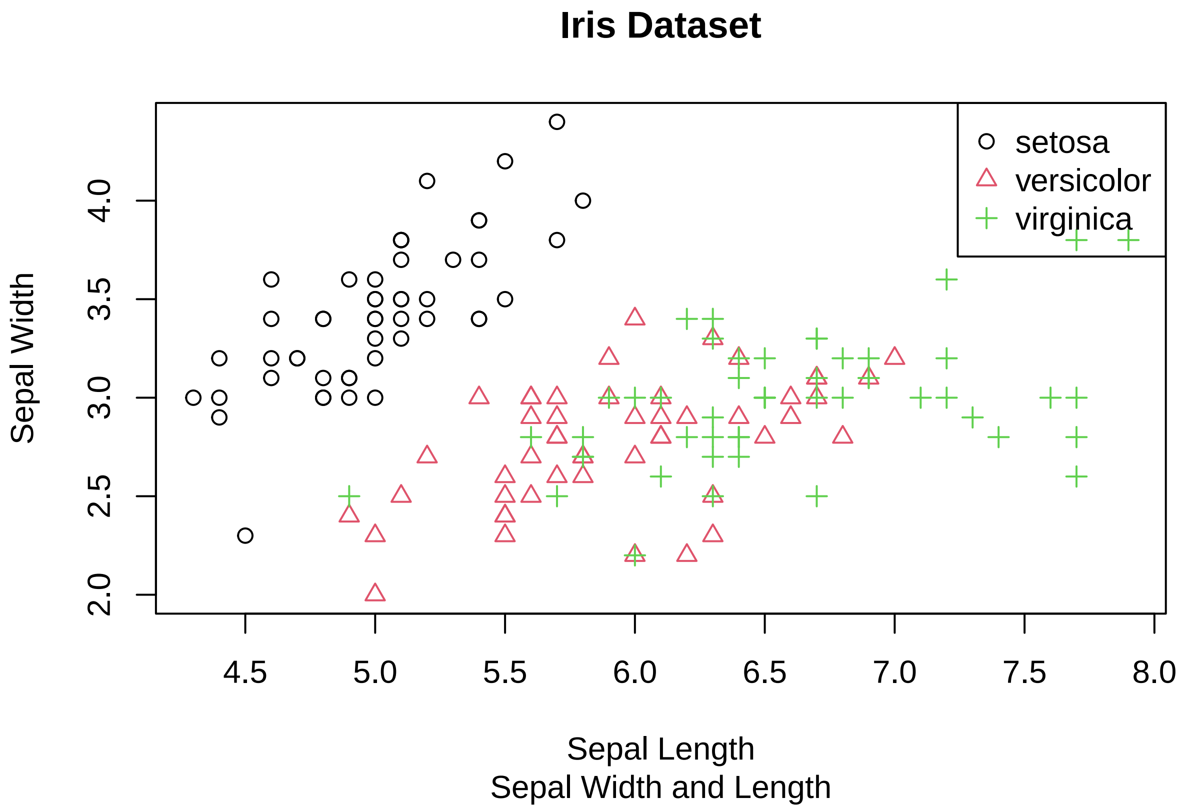
```r
tail(virginica)
```

```
##     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
```

- 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. Hint: Need to convert to factors the species to store categorical variables.

```r
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",
  ylab = "Sepal Width"
)

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



- f. Interpret the result *The Setosa species' sepal width has a relation to its sepal length. The longer the length, the wider it's width.* While the Versicolor and Virginica species tend to overlap in the middle. The longer their length, their width almost stays the same arounf 3.0.