

RWorksheet_Caneso#4b

Rovel Jan Caneso

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)

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

for(i in 1:5){
  for(j in 1:5){
    matr[i, j] <- vectorA[abs(i-j) + 1]
  }
}

for(i in 1:5){
  for(j in 1:5){
    cat(matr[i,j], " ")
  }
  cat("\n")
}
```

```
## 1  2  3  4  5
## 2  1  2  3  4
## 3  2  1  2  3
## 4  3  2  1  2
## 5  4  3  2  1
```

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

for(i in 1:5){
  for(j in 1:5){
    cat(matri[i,j], " ")
  }
  cat("\n")
}
```

```
## 0  0  0  0  0
## 0  0  0  0  0
## 0  0  0  0  0
## 0  0  0  0  0
## 0  0  0  0  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.

```
#inp <- readline("Enter a number: ")  
inp <- 1  
  
f <- 0  
s <- 1  
repeat{  
  if(f >= inp){  
    cat(f, " ")  
  }  
  
  fibona <- f+s  
  f <- s  
  s <- fibona  
  
  if(f > 500) break  
}
```

```
## 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 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 <- read.csv("shoesize_data.csv")  
head(shoesize)
```

```
## Shoe_Size Height Gender  
## 1 6.5 66.0 F  
## 2 9.0 68.0 F  
## 3 8.5 64.5 F  
## 4 8.5 65.0 F  
## 5 10.5 70.0 M  
## 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 Female? Write the R scripts and its output.

```
female <- subset(shoesize, Gender == "F", select = Gender)
female
```

```
##      Gender
## 1         F
## 2         F
## 3         F
## 4         F
## 6         F
## 7         F
## 8         F
## 10        F
## 12        F
## 17        F
## 18        F
## 20        F
## 21        F
## 24        F
```

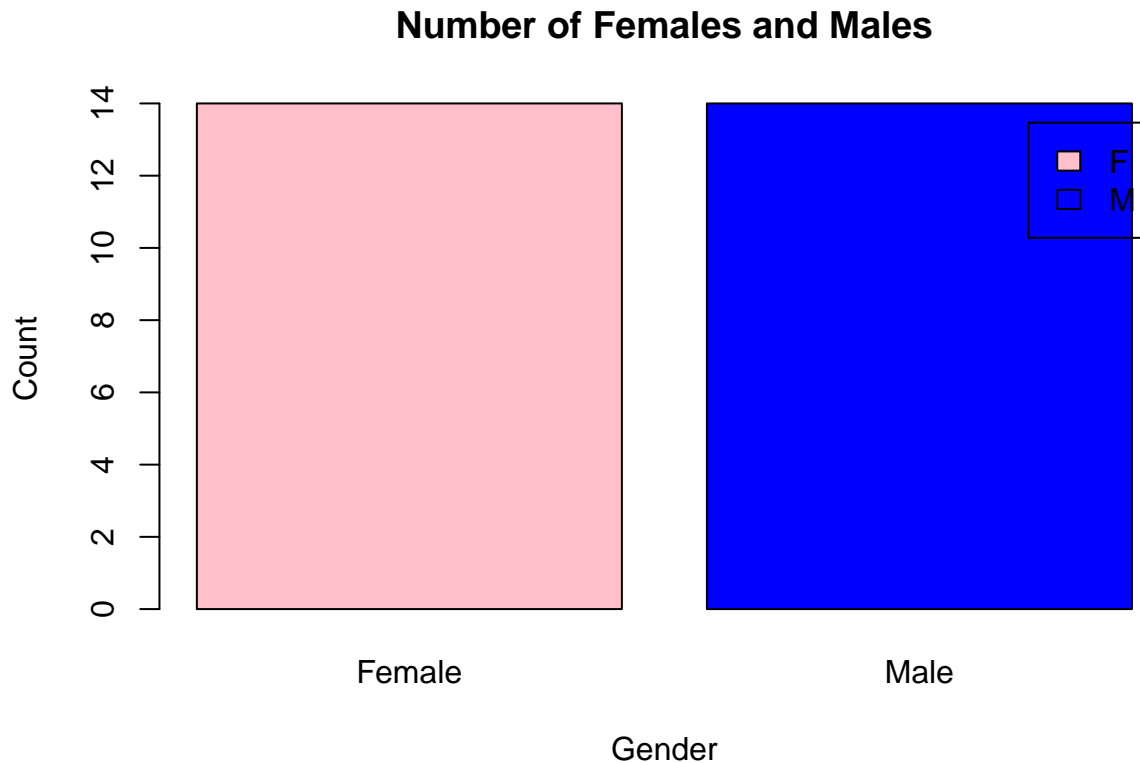
```
male <- subset(shoesize, Gender == "M", select = Gender)
male
```

```
##      Gender
## 5         M
## 9         M
## 11        M
## 13        M
## 14        M
## 15        M
## 16        M
## 19        M
## 22        M
## 23        M
## 25        M
## 26        M
## 27        M
## 28        M
```

#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.

```
genders <- table(shoesize$Gender)

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



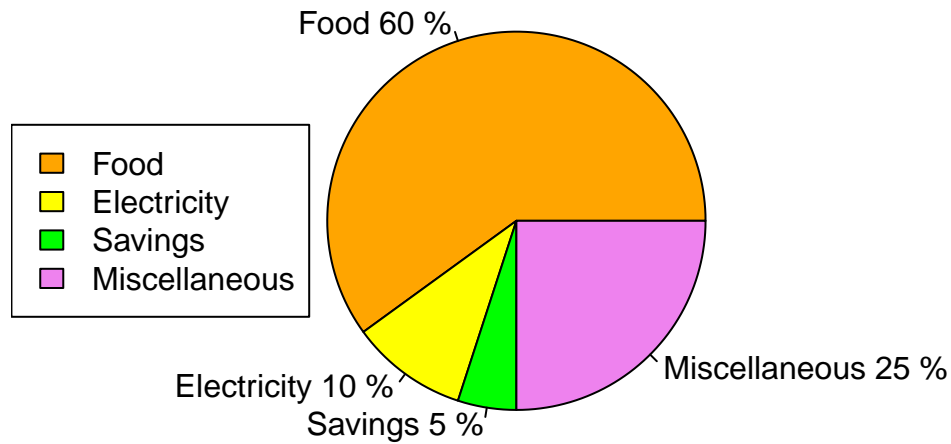
5. The monthly income of Dela Cruz family was spent on the following:

```
# 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.
category <- c("Food", "Electricity", "Savings", "Miscellaneous")
val <- c(60, 10, 5, 25)
color = c("Orange", "Yellow", "Green", "Violet")
percent <- round(val / sum(val) * 100)
per_label <- paste(category, percent, "%")

pie(
  val,
  col = color,
  main = "Dela Cruz Expenses",
  label = per_label
)

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

Dela Cruz Expenses



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 output.
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 0.2 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 ...

#[It shows the a data frame with the Sepal and Petal's lenght and width.]
#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?
slength <- mean(iris$Sepal.Length)
swidth <- mean(iris$Sepal.Width)
plength <- mean(iris$Petal.Length)
pwidth <- mean(iris$Petal.Width)
means <- c(
  Sepal_Length = slength,
  Sepal_Width = swidth,
  Petal_Length = plength,
  Petal_Width = pwidth
)
```

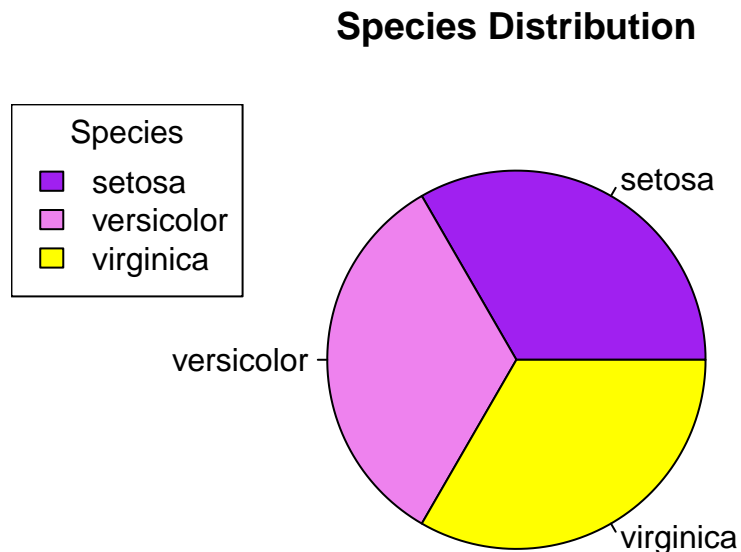
```

)
means

## 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.
iSpecies <- table(iris$Species)
species_color <- c("Purple", "Violet", "Yellow")
pie(
  iSpecies,
  main = "Species Distribution",
  col = species_color
)
legend("topleft", names(iSpecies), 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.
```

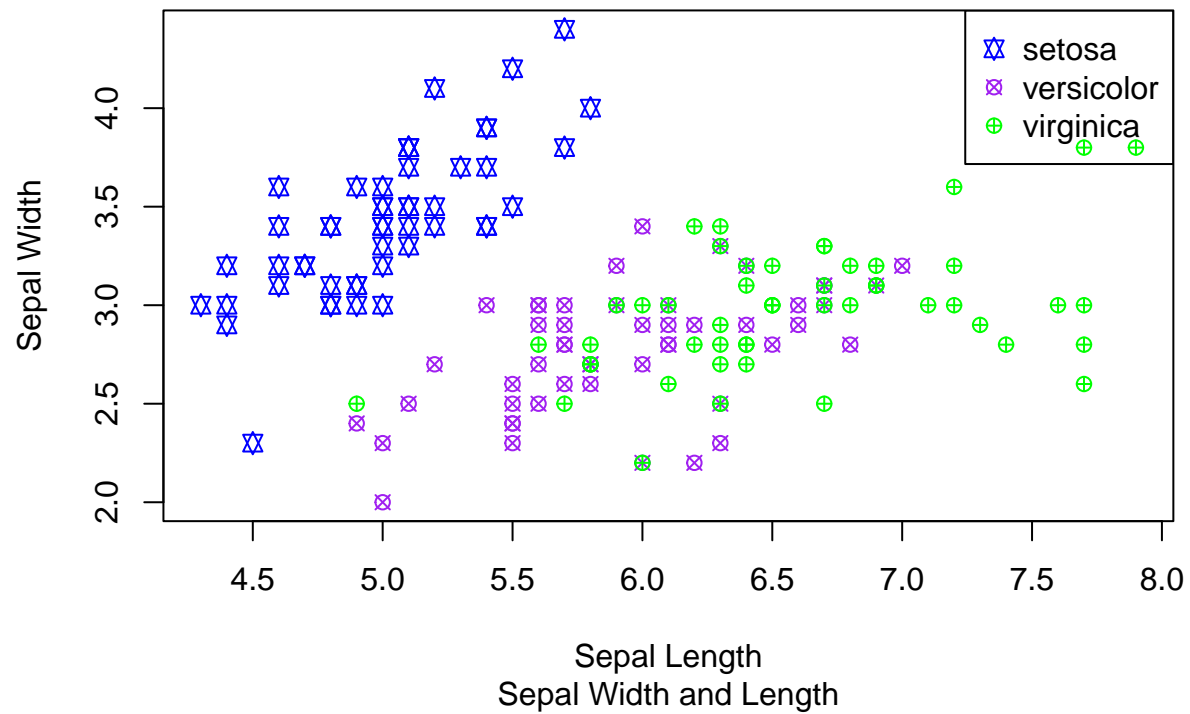
```
iris$Species <- as.factor(iris$Species)

colors <- c("blue", "purple", "green")
pch_symbols <- c(11, 13, 10)

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)
```

Iris Dataset



```
#f. Interpret the result.
#[--The scatterplot provides the correlation of Sepal.Length with Sepal.Width for the species
#  iris-Virginica, #Versicolor, and Setosa.
#--Setosa is easily distinguished from the other species due to its distinct clustering
#  and shorter sepal lengths.
#--In both dimensions, the ranges of virginica and versicolor overlap; in general,
#  virginica has the longest sepals.]
```

““