

RWorksheet_Paclibar#4b

Jhon Albert Paclibar

2024-10-28

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
summary(cars)
```

```
##      speed      dist
##  Min.   : 4.0    Min.   :  2.00
## 1st Qu.:12.0    1st Qu.: 26.00
##  Median :15.0    Median : 36.00
##   Mean  :15.4    Mean   : 42.98
## 3rd Qu.:19.0    3rd Qu.: 56.00
##   Max.  :25.0    Max.    :120.00
```

Including Plots

You can also embed plots, for example:



Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that

generated the plot. #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.

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

for (i in 0:4) {
  for (j in 0:4) {
    result_matrix[i + 1, j + 1] <- abs(i - j)
  }
}
print(result_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. 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.

```
start_value <- 1

fib1 <- 0
fib2 <- 1

repeat {
  fib_next <- fib1 + fib2
  if (fib_next >= start_value) {
    cat(fib_next, "\n")
  }
  if (fib_next > 500) {
    break
  }

  fib1 <- fib2
  fib2 <- fib_next
}
```

```
## 1
## 2
## 3
## 5
## 8
```

```
## 13
## 21
## 34
## 55
## 89
## 144
## 233
## 377
## 610
```

#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.csv")
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
## 7          9.5   70.0      F
## 8          9.0   71.0      F
## 9         13.0   72.0      M
## 10         7.5   64.0      F
## 11         10.5   74.5      M
## 12          8.5   67.0      F
## 13         12.0   71.0      M
## 14         10.5   71.0      M
## 15         13.0   77.0      M
## 16         11.5   72.0      M
## 17          8.5   59.0      F
## 18          5.0   62.0      F
## 19         10.0   72.0      M
## 20          6.5   66.0      F
## 21          7.5   64.0      F
## 22          8.5   67.0      M
## 23         10.5   72.0      M
## 24          8.5   69.0      F
## 25         10.5   72.0      M
## 26         11.0   70.0      M
## 27          9.0   69.0      M
## 28         13.0   70.0      M
```

#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.

```
maleSubset <- subset(ShoeSize, Gender == "M")
femaleSubset <- subset(ShoeSize, Gender == "F")

numMale <- nrow(maleSubset)
numFemale <- nrow(femaleSubset)

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

```
## Number of Male observations: 14
```

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

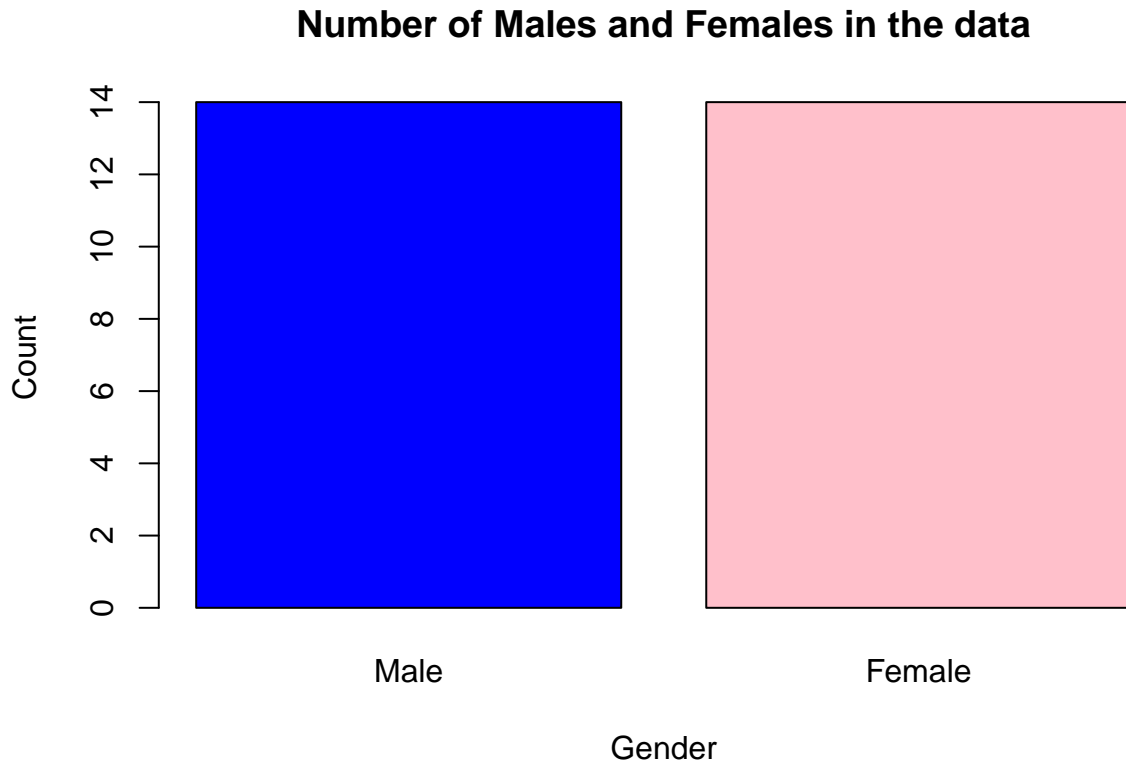
```
## Number of Female observations: 14
```

#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.

```
genderCount <- c(Male = numMale, Female = numFemale)
```

```
barColors <- c("blue", "pink")
```

```
SGender <- barplot(genderCount, main = "Number of Males and Females in the data", xlab = "Gender", ylab =
```



```
SGender
```

```
##      [,1]
```

```
## [1,]  0.7
```

```
## [2,]  1.9
```

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

```
Food <- 60
```

```
Electricity <- 10
```

```
Savings <- 5
```

```
Miscellaneous <- 25
```

```
delaCruz <- data.frame(Food, Electricity, Savings, Miscellaneous)
```

```
delaCruz
```

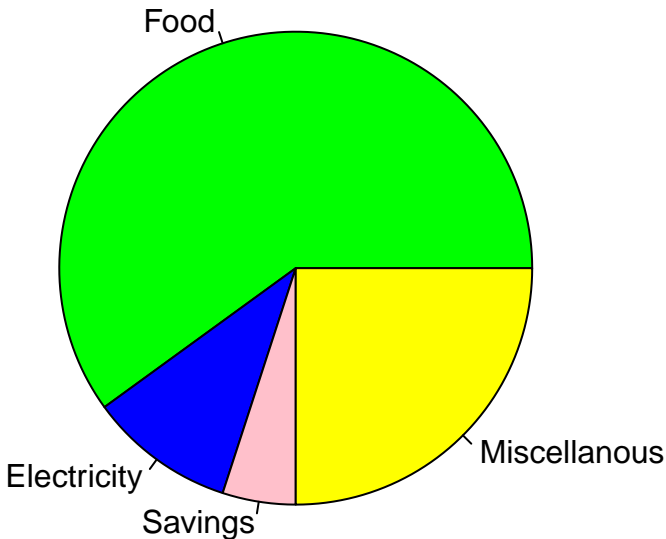
```
##   Food Electricity Savings Miscellaneous
```

```
## 1    60          10      5             25
```

#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.

```
dela <-pie(c(Food, Electricity, Savings, Miscellaneous),
  labels = c("Food", "Electricity", "Savings", "Miscellaneous"),
  main = "Dela Cruz family monthly spending",
  col = c("green", "blue", "pink", "yellow"),
  radius = 1)
```

Dela Cruz family monthly spending



```
dela
```

```
## NULL
```

```
#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 0.4 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 data about the iris flower including the length and width of the petals and sepal. It also shows the species of the irises.

#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?

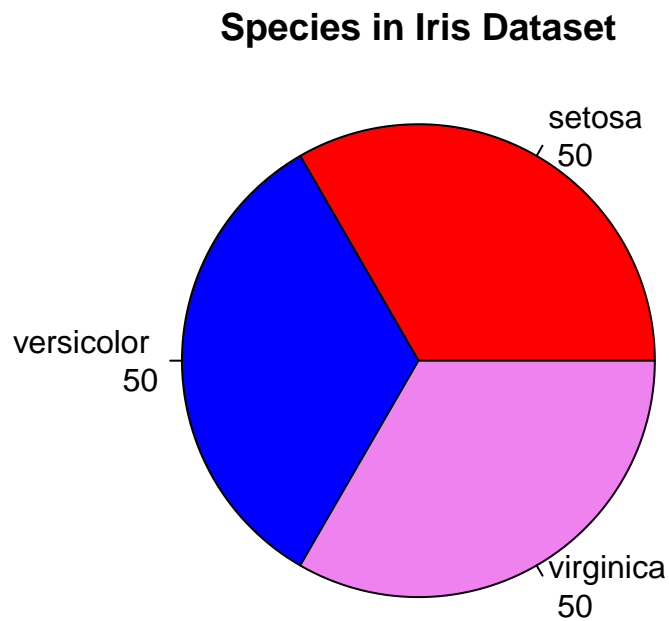
```
lengthNwidth <- colMeans(iris[, 1:4])
lengthNwidth
```

```
## 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.

```
speciesCounts <- table(iris$Species)

PieS <- pie(speciesCounts,
  labels = paste(names(speciesCounts), "\n", speciesCounts),
  main = "Species in Iris Dataset",
  col = c("red", "blue", "violet"),
  radius = 1)
```



PieS

NULL

#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")
Seeto<-tail(setosa)
Veersi<- tail(versicolor)
Viirgi<- tail(virginica)
Seeto
```

```
##      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
```

Veersi

```
##      Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 95          5.6         2.7         4.2         1.3 versicolor
```

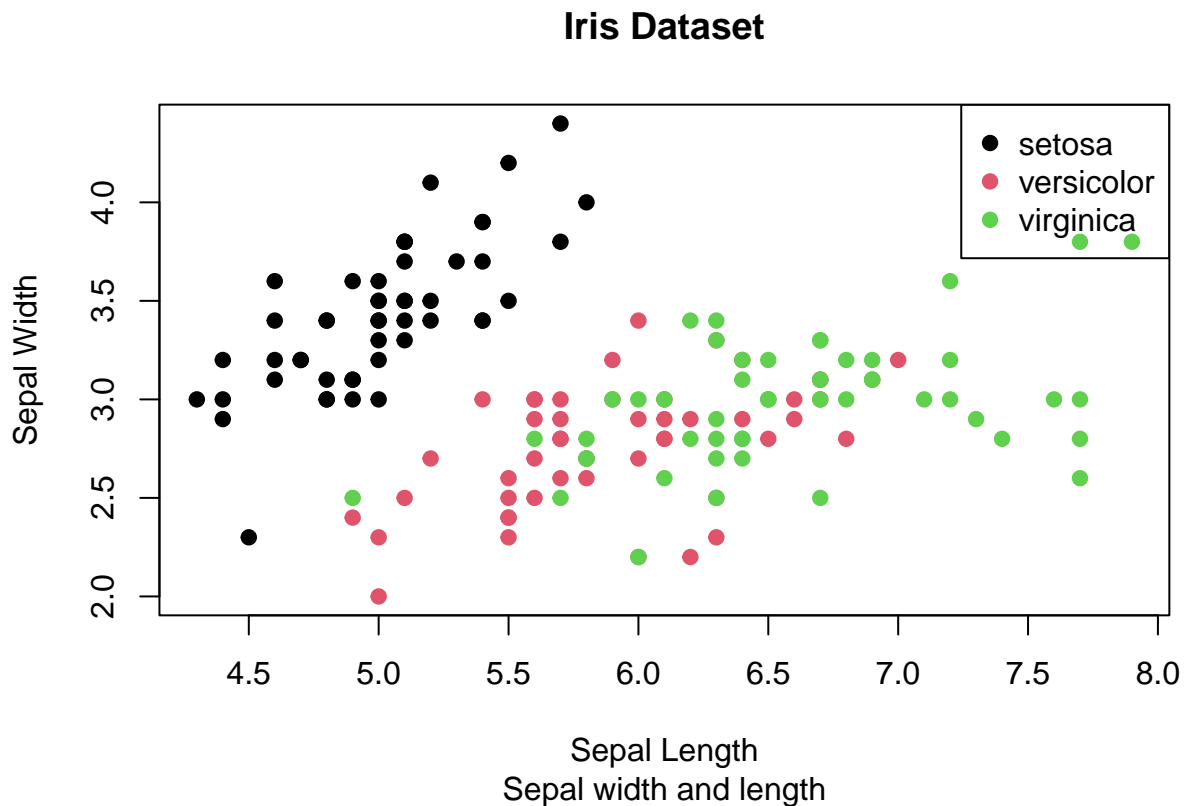
```
## 96      5.7      3.0      4.2      1.2 versicolor
## 97      5.7      2.9      4.2      1.3 versicolor
## 98      6.2      2.9      4.3      1.3 versicolor
## 99      5.1      2.5      3.0      1.1 versicolor
## 100     5.7      2.8      4.1      1.3 versicolor
```

Viirgi

```
##      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.

```
plot(iris$Sepal.Length, iris$Sepal.Width, main = "Iris Dataset", sub = "Sepal width and length", xlab =
legend("topright", legend = levels(iris$Species), pch = 19, col = 1:3))
```



#f. Interpret the result. #The result separates the 3 species by color to easily identify them and also to mark the locations on where the species can be seen.

#7. Import the alexa-file.xlsx. Check on the variations. Notice that there are extra 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).