RWork-sheet_gener#4b.Rmd

Kent Hervey D. Gener BSIT2-B

2023-11-21

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.

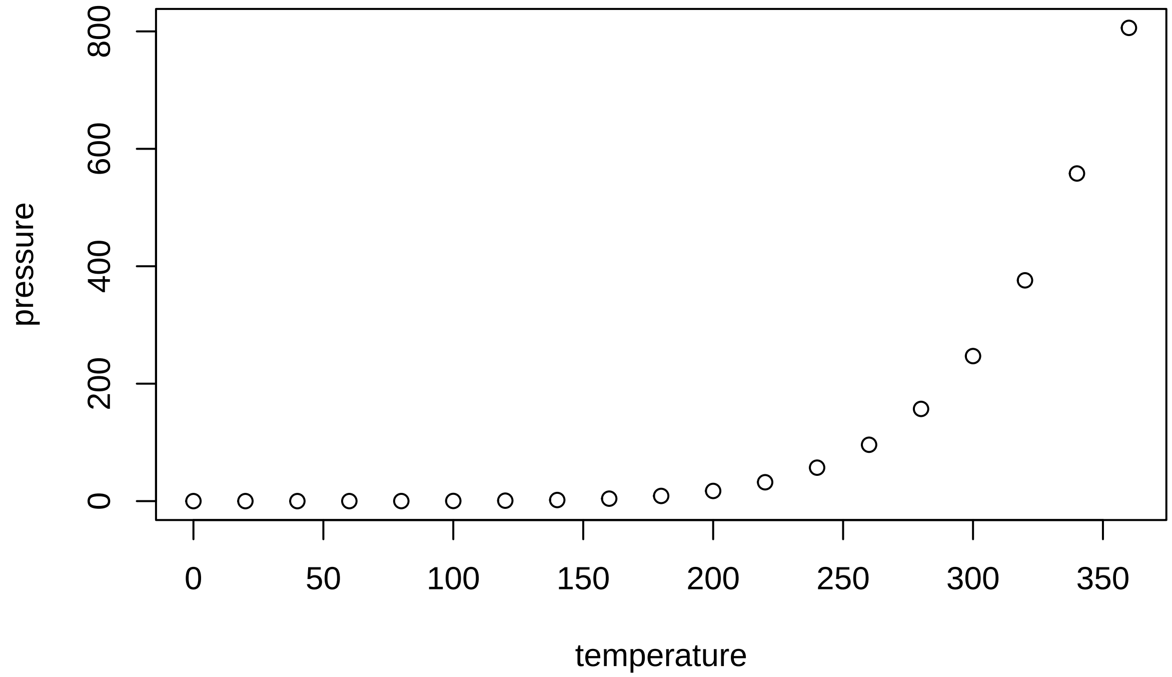
When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```r
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.

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

```r
vectorA <- c(1, 2, 3, 4, 5)

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

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

print(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

```
num_rows <- 5

for (i in 1:num_rows) {
  for (j in 1:i) {
    cat("* ", sep = "")
  }
  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.

```
fibonacci <- function(n) {
  if (n <= 1) {
    return(n)
  } else {
    return(fibonacci(n-1) + fibonacci(n-2))
  }
}

user_input <- as.numeric(readline(prompt = "Enter the number of terms for Fibonacci sequence: "))
```

```
## Enter the number of terms for Fibonacci sequence:
```

```
current_term <- 1
fibonacci_value <- 0

repeat {
  fibonacci_value <- fibonacci(current_term)

  if (fibonacci_value > 500) {
    break
  }

  cat(fibonacci_value, " ")

  current_term <- current_term + 1
}
```

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

Using Basic Graphics (plot(),barplot(),pie(),hist()) 4. Import the dataset as shown in Figure 1 you have created previously.

```
figure1matrix <- read.csv("Figure1Matrix.csv")
```

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

```
Shoe_Size <- read.csv("ShoeSize.csv")

head(Shoe_Size, 6)
```

```
##   X ShoeSize Height Gender
## 1 1      6.5   66.0      F
## 2 2     90.0   68.0      F
## 3 3     85.0   64.5      F
## 4 4     85.0   65.0      F
## 5 5     10.5   70.0      M
## 6 6     70.0   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 <- subset(Shoe_Size, Gender == "F")

male_subset <- subset(Shoe_Size, Gender == "M")

nrow(female_subset)
```

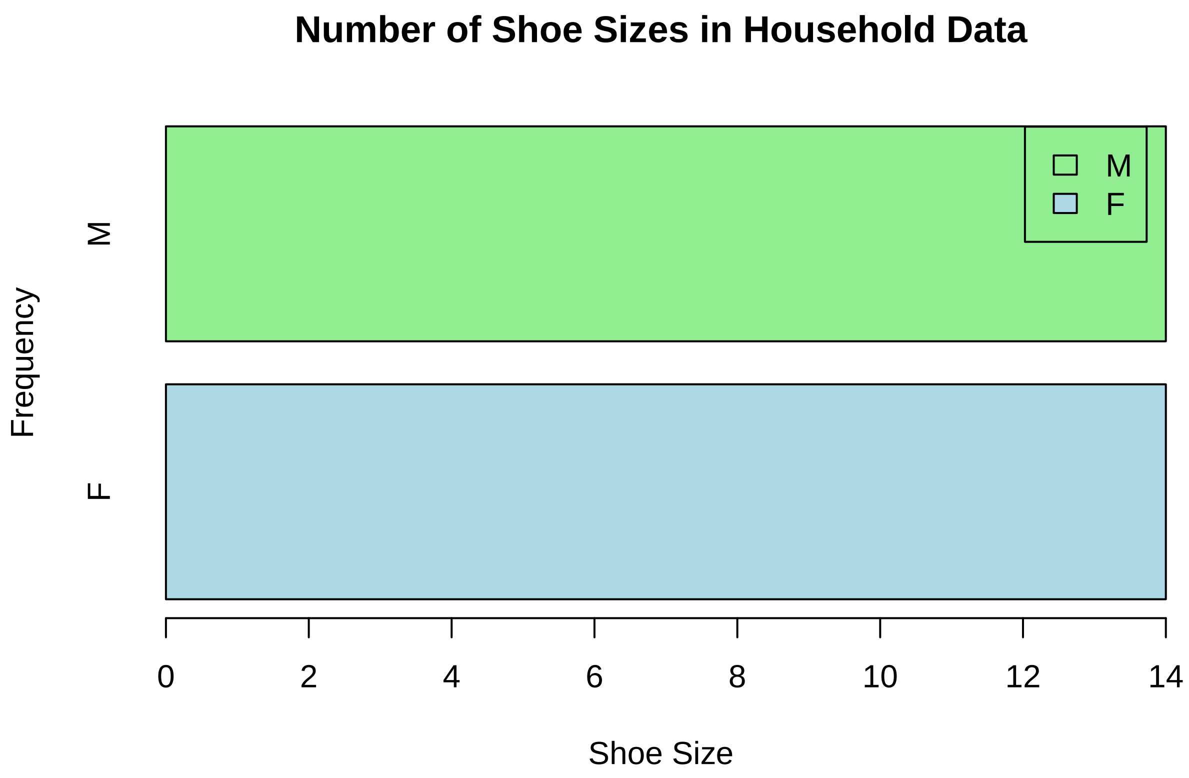
```
## [1] 14
```

```
nrow(male_subset)
```

```
## [1] 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.

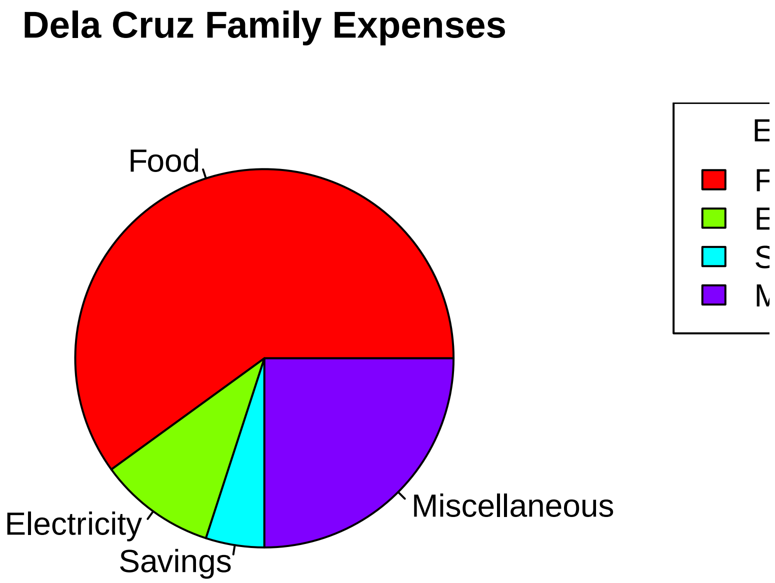
```
freq_table <- table(Shoe_Size$Gender)

colors <- c("lightblue", "lightgreen")

barplot(freq_table,
        main = "Number of Shoe Sizes in Household Data",
        xlab = "Shoe Size",
        ylab = "Frequency",
        col = colors,
        border = "black",
        legend = rownames(freq_table),
        horiz = TRUE)
```



5. The monthly income of Dela Cruz family was spent on the following: Food Electricity Savings Miscellaneous 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.

```r
dela_cruz <- data.frame(category = c("Food", "Electricity", "Savings", "Miscellaneous"),
                  value = c(60, 10, 5, 25))

pie(dela_cruz$value, labels = dela_cruz$category, col = rainbow(nrow(dela_cruz)), main = "Dela Cruz Fam
legend("topright", inset = c(-0.2, 0), legend = dela_cruz$category, fill = rainbow(nrow(dela_cruz)), ti
```



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 ...
```

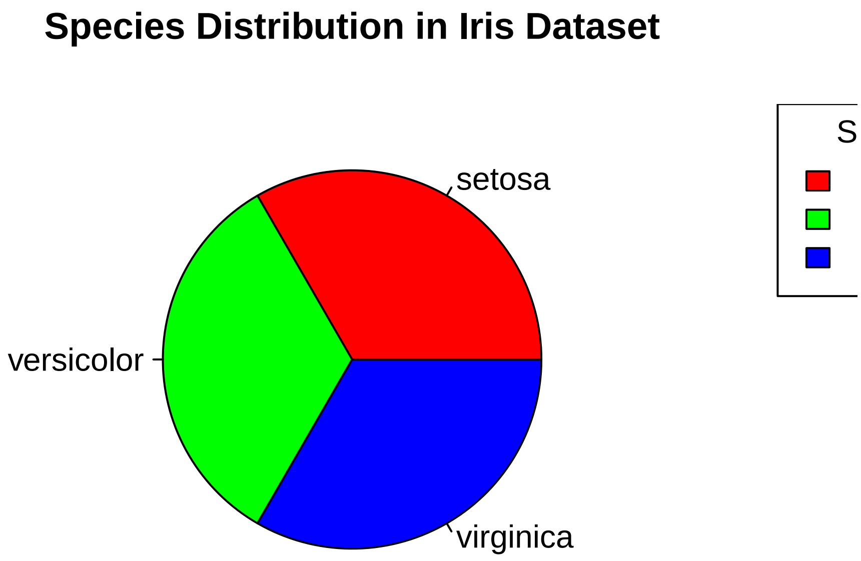
b. Create an R object that will contain the mean of the sepal.length, sepal.width,petal.length,and petal.width. What is the R script and its result?

```
means <- aggregate(iris[, 1:4], by=list(Species=iris$Species), FUN=mean)

means
```

```
##      Species Sepal.Length Sepal.Width Petal.Length Petal.Width
## 1     setosa        5.006       3.428        1.462       0.246
## 2 versicolor        5.936       2.770        4.260       1.326
## 3  virginica        6.588       2.974        5.552       2.026
```

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

```r
pie_chart <- pie(table(iris$Species), col=rainbow(3), main="Species Distribution in Iris Dataset")

legend("topright", inset=c(-0.15, 0), legend=levels(iris$Species), fill=rainbow(3), 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.

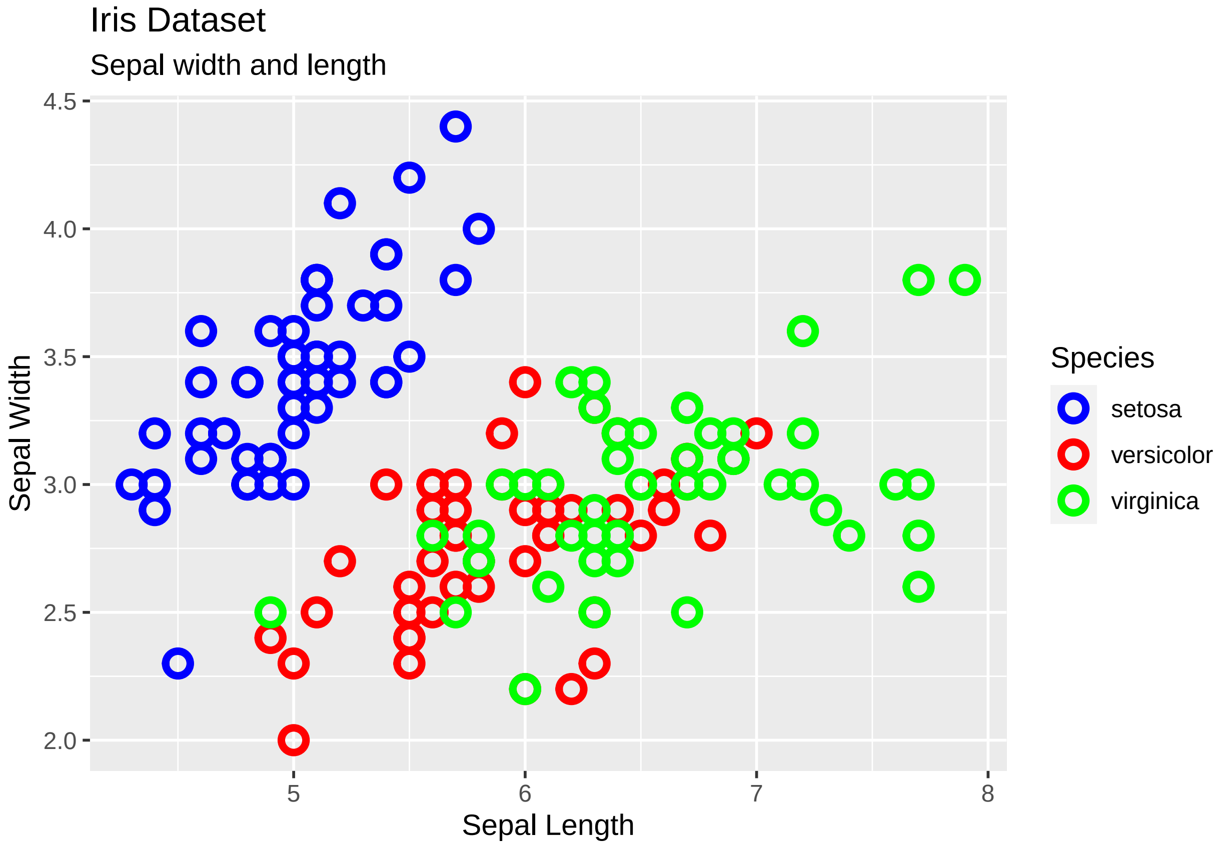
```r
setosa <- subset(iris, Species == "setosa")
versicolor <- subset(iris, Species == "versicolor")
virginica <- subset(iris, Species == "virginica")

species <- c(setosa,versicolor,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.

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.3.2
```

```
ggplot(data = iris, aes(x = Sepal.Length, y = Sepal.Width, color = Species)) +
 geom_point(size = 3, shape = 21, stroke = 2) +
 scale_color_manual(values = c("setosa" = "blue", "versicolor" = "red", "virginica" = "green")) +
 labs(title = "Iris Dataset", subtitle = "Sepal width and length",
       x = "Sepal Length", y = "Sepal Width")
```



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

```
species_factors <- factor(iris$Species)
```

f. Interpret the result.

```
#You can now use the species_factors variable in your model instead of the original species variable. T
```