Worksheet-5 in R

Worksheet for R Programming

Instructions:

- Use RStudio or the RStudio Cloud accomplish this worksheet.

- Save the R script as *RWorksheet_lastname#5.R*.
- Create your own *GitHub repository* and push the R script as well as this pdf worksheet to your own repo. Do not forget to comment your Git repo

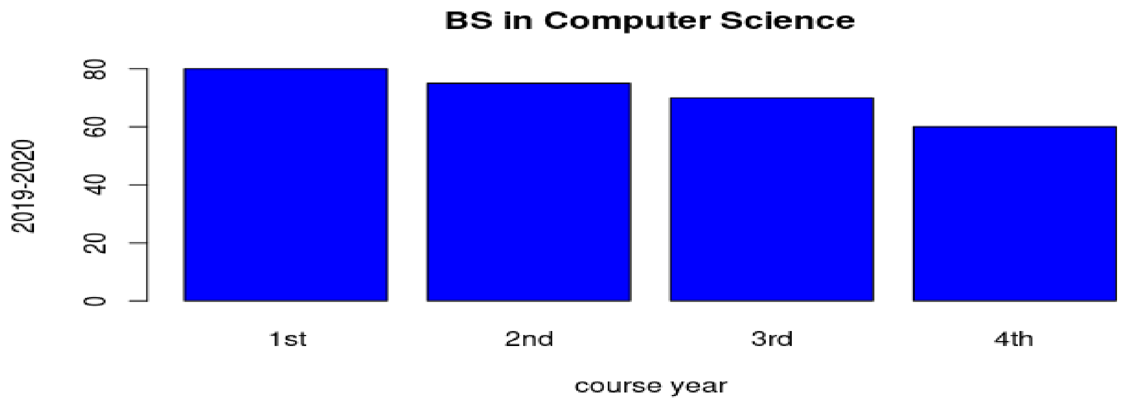
Accomplish this worksheet by answering the questions being asked and writing the code manually.

1. The table shows the enrollment of BS in Computer Science, SY 2010-2011.

Course Year	2019 - 2020
1st	80
2nd	75
3rd	70
4th	60

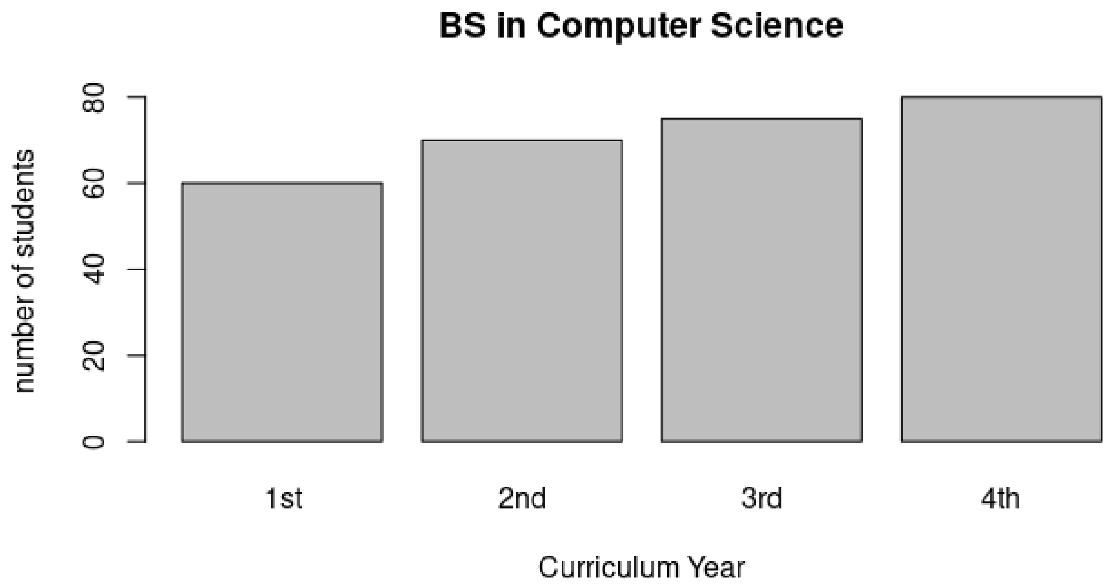
a. Plot the data using a bar graph. Write the codes and copy the result.

```
Enrollment <-c(80,75,70,60)
barplot(Enrollment,
        main="BS in Computer Science",
        xlab="course year",
        ylab="2019-2020",
        names.arg = c("1st", "2nd", "3rd", "4th"),
        col="blue",)
```



b. Using the same table, label the barchart with
   Title = ” Enrollment of BS Computer Science
   horizontal axis = “Curriculum Year” and vertical
   axis = “number of students”

```
Enrollment <-c(60,70,75,80)
barplot(Enrollment,
        main="BS in Computer Science",
        xlab="Curriculum Year",
        ylab="number of students",
        names.arg = c("1st", "2nd", "3rd", "4th"),)
```



2. The monthly income of De Jesus family was spent on the following:

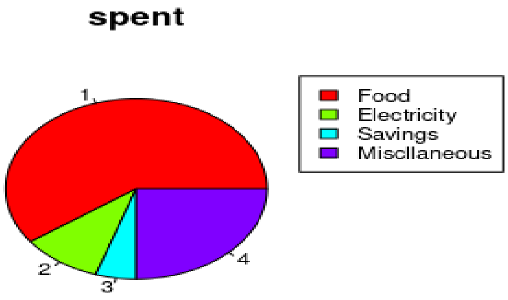
60% on Food, 10% on electricity, 5% for savings, and
25% for other miscellaneous expenses.

a. Create a table for the above scenario.
   Write the codes and its result.

```
spent <- c(60,10,5,25)
```

```
barplot(spent,names.arg = c("Food",
"Electricity", "Savings", "Miscllaneous"))
```

b. Plot the data using a pie chart. Add labels, colors and legend.
   Write the codes and its result.



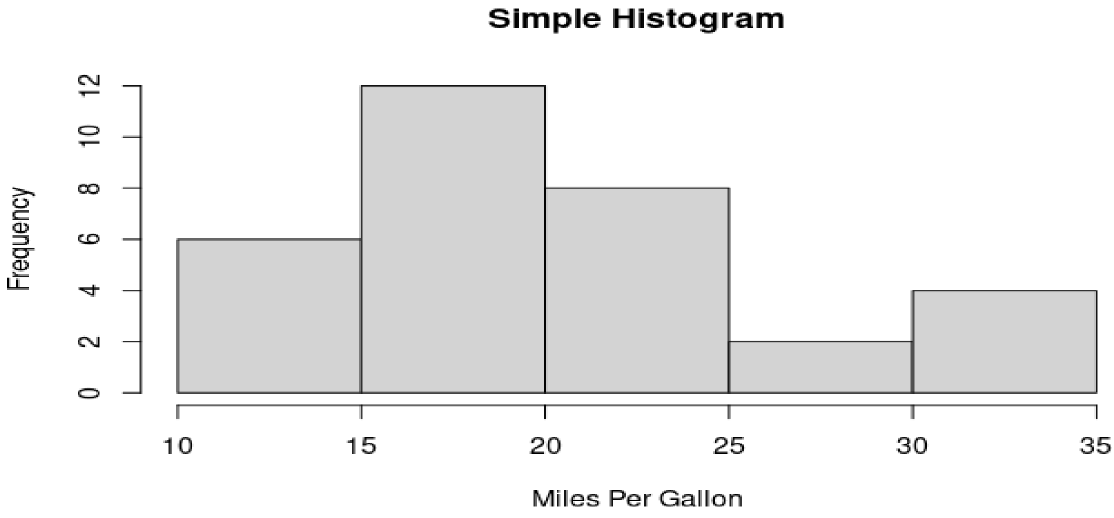
3. Open the mtcars dataset.

```
data("mtcars")

n1<-mtcars$mpg
n1
```

a. Create a simple histogram specifically for **mpg** (miles per gallon) variable. Use `$` to select the *mpg* only. Write the codes and its result.

```
n3 <-hist(n1, xlab="Miles Per Gallon",
          main="Simple Histogram")
```

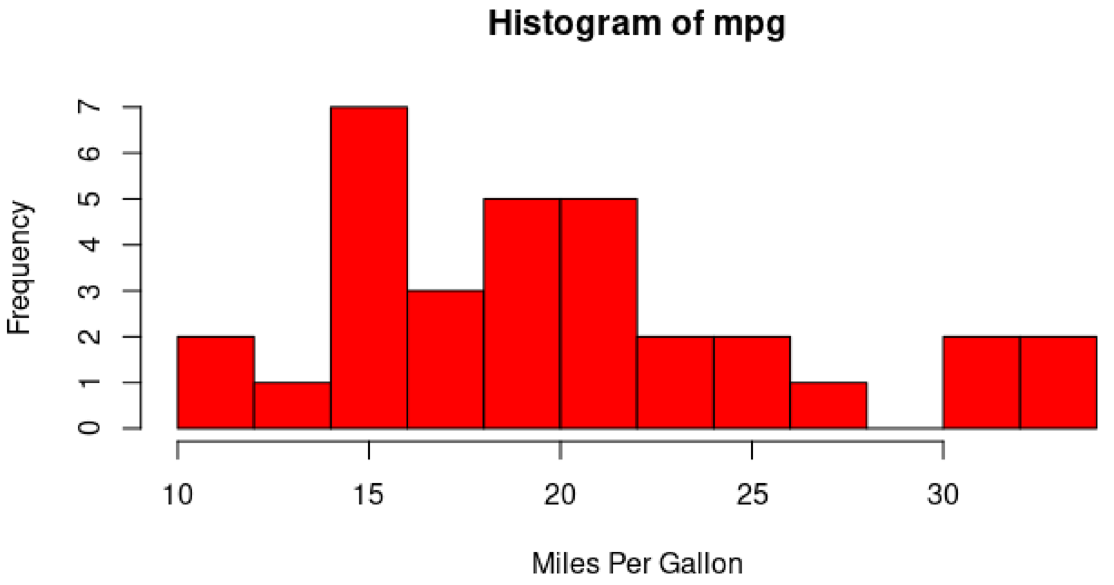


b. Colored histogram with different number of bins.

```
hist(mtcars$mpg, breaks=12, col=”red”)
```

Note: breaks= controls the number of bins

```
numb3b <-hist(n1, breaks=12, col="red", xlab="Miles Per Gallon",
          main="Histogram of mpg")
```

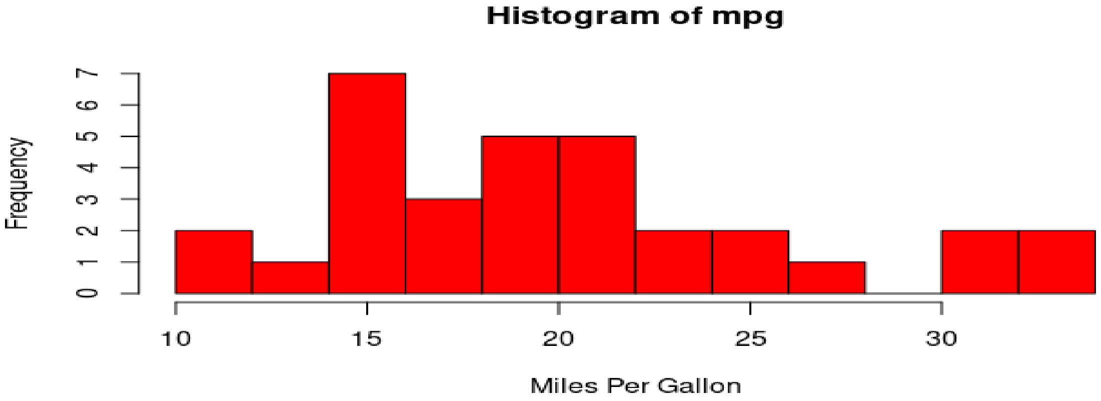


c. Add a Normal Curve

```
x <- mtcars$mpg
h<-hist(x, breaks=10, col="red", xlab="Miles Per Gallon",
     main="Histogram with Normal Curve")
xfit<-seq(min(x),max(x),length=40)
yfit<-dnorm(xfit,mean=mean(x),sd=sd(x))
yfit <- yfit*diff(h$mids[1:2])*length(x)
lines(xfit, yfit, col="blue", lwd=2)
```

Copy the result.

```
n3c <-hist(n1, breaks=12, col="red", xlab="Miles Per Gallon",
        main="Histogram with Normal Curve")
xfit<-seq(min(n1),max(n1),length=40)
yfit<-dnorm(xfit,mean=mean(n1),sd=sd(n1))
yfit <- yfit*diff(n3c$mids[1:2])*length(n1)
lines(xfit, yfit, col="blue", lwd=2)
```



4. Open the `iris` dataset. Create a subset for each species.

a. Write the codes and its result.

```
data("iris")
set1 <- subset(iris, Species == "setosa")
set2 <- subset(iris, Species == "versicolor")
set3 <- subset(iris, Species == "virginica")
```

b. Get the mean for every characteristics of each species usingcolMeans().

Write the codes and its result.
Example: setosa <- colMeans(setosa[sapply(setosaDF, is.numeric)])

```
set1 <- subset(iris, Species == ”setosa”)
setosa <- colMeans(set1[sapply(set1, is.numeric)])
setosa

set2 <- subset(iris, Species == ”versicolor”)
versi <- colMeans(set2[sapply(set2, is.numeric)])
versi

set3<- subset(iris, Species == ”virginica”)
virg<- colMeans(set3[sapply(set3, is.numeric)])
virg
```

```
> set1 <- subset(iris, Species == "setosa")
> setosa <- colMeans(set1[sapply(set1,is.numeric)])
> setosa
Sepal.Length  Sepal.Width Petal.Length  Petal.Width 
       5.006        3.428        1.462        0.246 
> 
> set2 <- subset(iris, Species == "versicolor")
> versi <- colMeans(set2[sapply(set2,is.numeric)])
> versi
Sepal.Length  Sepal.Width Petal.Length  Petal.Width 
       5.936        2.770        4.260        1.326 
> 
> set3<- subset(iris, Species == "virginica")
> virg<- colMeans(set3[sapply(set3,is.numeric)])
> virg
Sepal.Length  Sepal.Width Petal.Length  Petal.Width 
       6.588        2.974        5.552        2.026 


>
```

c. Combine all species by using `rbind()`

The table should be look like this:

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width
setosa			
versicolor			
virginia			

```
t1 <- rbind(setosa, versi, virg)
> t1
       Sepal.Length Sepal.Width Petal.Length Petal.Width
setosa        5.006       3.428        1.462       0.246
versi         5.936       2.770        4.260       1.326
virg          6.588       2.974        5.552       2.026


>
```

d. From the data in 4-c: Create the `barplot()`.
Write the codes and its result.
The barplot should be like this.

```
barplot(t1, beside = TRUE,
        main = "Iris Mean",
        xlab = "Characteristics",
        ylab = "Mean Scores",
        col = c("red","green","blue"))
```

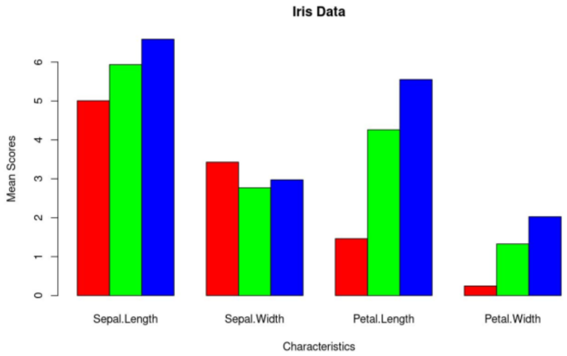


Figure 1: Iris Data using Barplot