| | |
|---|---|
| **Batch: H3-1** | **Roll No.: 16010122323** |
| **Experiment No. 1** | |

**Title :** Exploring R for Data Science

**Aim:** To understand basics of R - Operators, built-in functions, Data types, Data manipulation in R, R packages for Data Science

**Expected Outcome of Experiment:**
CO : Students should write

**Books/ Journals/ Websites referred:**
1. https://cran.r-project.org/
2. Students should write
3. Students should write

### What is R?

- R is a scripting/programming language and environment for statistical computing, data science and graphics.
- R is a successor of the proprietary statistical computing programming language S.
- It is an important tool for computational statistics, visualization and data science.

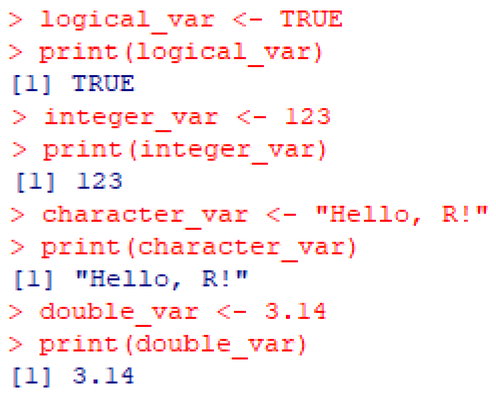
### Why R?

It provides techniques for various statistical analyses like classical tests and classification, time-series analysis, clustering, linear and non-linear modelling and graphical operations.

It has superior support for graphics.

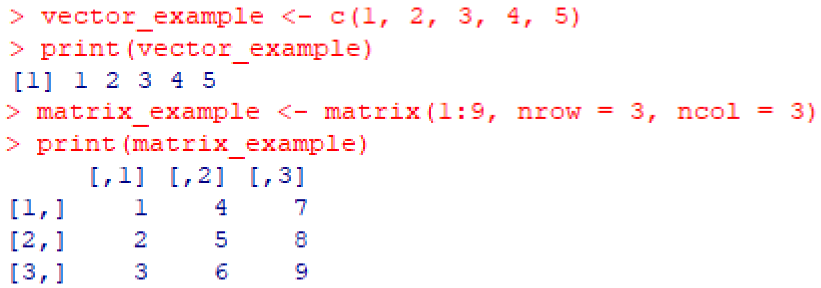
Reasons for learning R:
- Free, Open source
- Great visualization
- Cross-platform compatibility
- Advanced statistics
- Integration with other programming languages
- Supportive open source community
- Easy extensibility via packages

## 1. Exploring the atomic datatypes supported by R-Logical, Numeric-integer, Character, Double, Complex, Raw

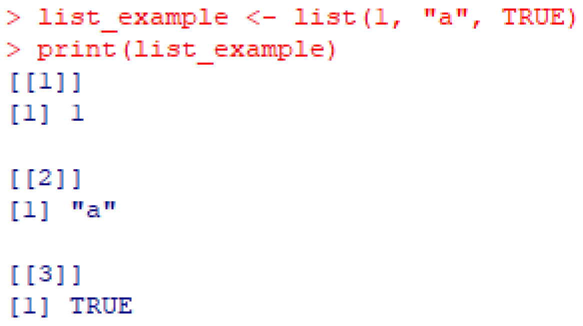


```
> logical_var <- TRUE
> print(logical_var)
[1] TRUE
> integer_var <- 123
> print(integer_var)
[1] 123
> character_var <- "Hello, R!"
> print(character_var)
[1] "Hello, R!"
> double_var <- 3.14
> print(double_var)
[1] 3.14
```

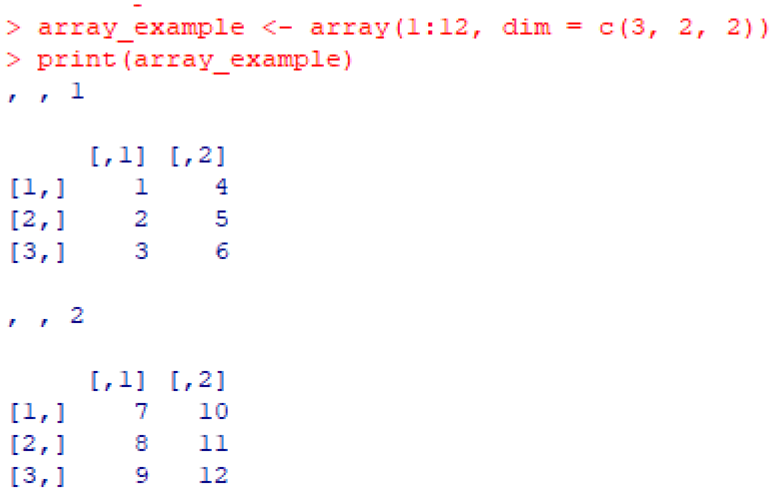
## 2. Exploring data manipulation of different data objects of R- Vectors- Matrices, Factors, List, Array, Data Frames



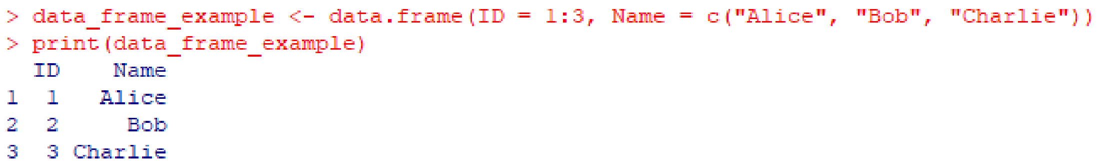
```
> vector_example <- c(1, 2, 3, 4, 5)
> print(vector_example)
[1] 1 2 3 4 5
> matrix_example <- matrix(1:9, nrow = 3, ncol = 3)
> print(matrix_example)
     [,1] [,2] [,3]
[1,]    1    4    7
[2,]    2    5    8
[3,]    3    6    9
```



```
> list_example <- list(1, "a", TRUE)
> print(list_example)
[[1]]
[1] 1

[[2]]
[1] "a"

[[3]]
[1] TRUE
```

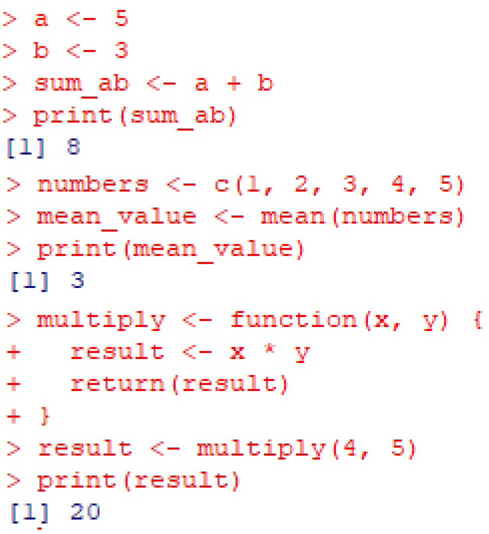


```
> array_example <- array(1:12, dim = c(3, 2, 2))
> print(array_example)
, , 1

     [,1] [,2]
[1,]    1    4
[2,]    2    5
[3,]    3    6

, , 2

     [,1] [,2]
[1,]    7   10
[2,]    8   11
[3,]    9   12
```



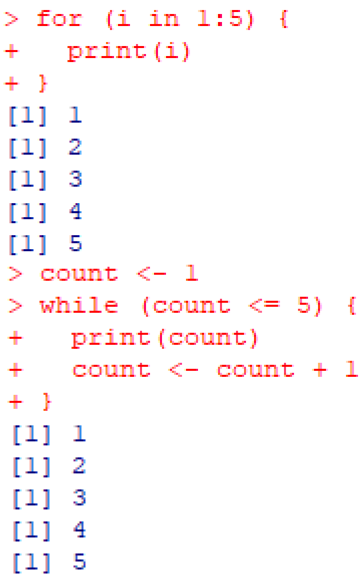
```
> data_frame_example <- data.frame(ID = 1:3, Name = c("Alice", "Bob", "Charlie"))
> print(data_frame_example)
  ID    Name
1  1   Alice
2  2     Bob
3  3 Charlie
```

## 3. Exploring Operators and built-in functions and writing user-defined functions in R



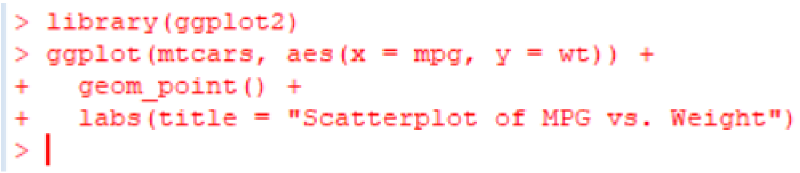
```
> a <- 5
> b <- 3
> sum_ab <- a + b
> print(sum_ab)
[1] 8
> numbers <- c(1, 2, 3, 4, 5)
> mean_value <- mean(numbers)
> print(mean_value)
[1] 3
> multiply <- function(x, y) {
+   result <- x * y
+   return(result)
+ }
> result <- multiply(4, 5)
> print(result)
[1] 20
```

## 4. Using Looping constructs in R

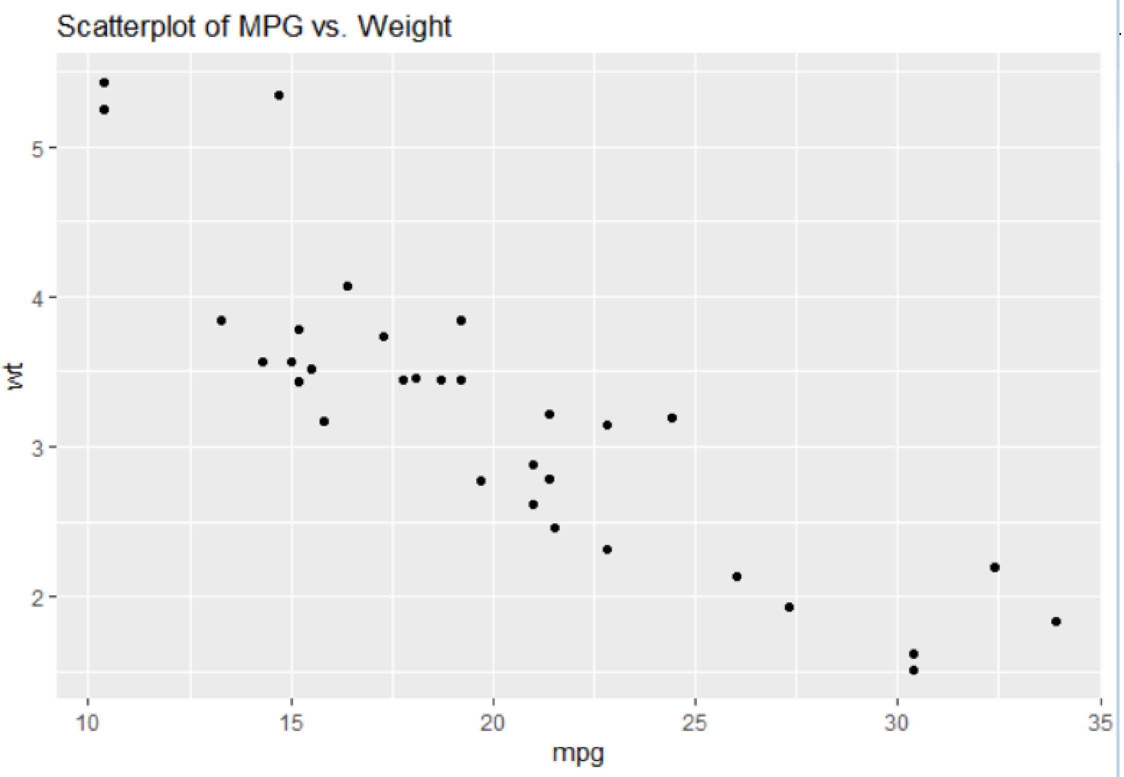


```
> for (i in 1:5) {
+   print(i)
+ }
[1] 1
[1] 2
[1] 3
[1] 4
[1] 5
> count <- 1
> while (count <= 5) {
+   print(count)
+   count <- count + 1
+ }
[1] 1
[1] 2
[1] 3
[1] 4
[1] 5
```

## 5. Exploring any Packages in R (any graphic package)



```
> library(ggplot2)
> ggplot(mtcars, aes(x = mpg, y = wt)) +
+   geom_point() +
+   labs(title = "Scatterplot of MPG vs. Weight")
> 
```



## Post Lab questions

### Write R commands for the following

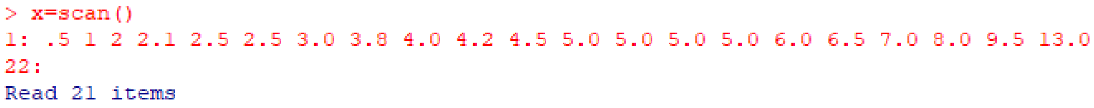
**1.** In an article in American Journal of Pathology, Pitts et al (2001) have taken the measurements on diameters in centimetres of the neoplasm removed from the breasts of 20 subjects with pure sarcoma. Following is the dataset: 0.5, 1, 2, 2.1, 2.5, 2.5, 3.0, 3.8, 4.0, 4.2, 4.5, 5.0, 5.0, 5.0, 5.0, 6.0, 6.5, 7.0, 8.0, 9.5, 13.0

I) Enter the dataset using scan function and store in the variable X

II) Find the mean, median, variance and standard deviation of x

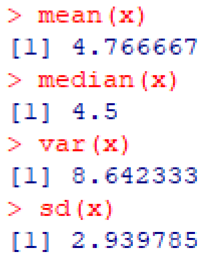
III) Create the boxplot

i)



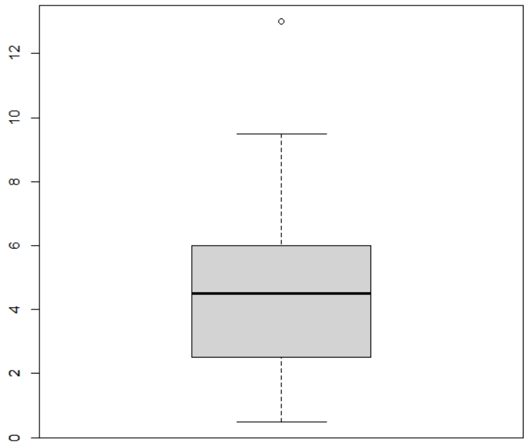
```
> x=scan()
1: .5 1 2 2.1 2.5 2.5 3.0 3.8 4.0 4.2 4.5 5.0 5.0 5.0 5.0 6.0 6.5 7.0 8.0 9.5 13.0
22: 
Read 21 items
```

ii)



```
> mean(x)
[1] 4.766667
> median(x)
[1] 4.5
> var(x)
[1] 8.642333
> sd(x)
[1] 2.939785
```

iii)

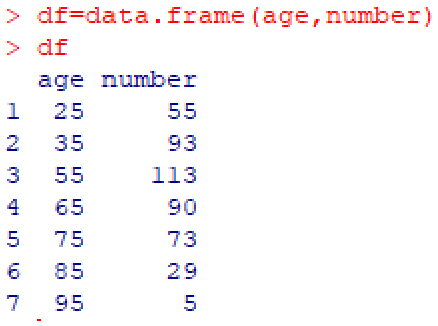


3. American Journal of psychiatry conducted a study of the presence of significant psychiatric illness in heterozygous carriers of the gene for the Wolfram syndrome. Among the subject studied were 543 blood relatives of patients of Wolfram syndrome. Following is the frequency distribution of ages of these blood relatives:

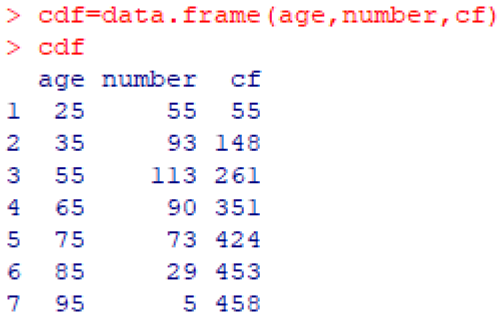
| Age(Mid-point) | 25 | 35 | 55 | 65 | 75 | 85 | 95 |
|---|---|---|---|---|---|---|---|
| Number(Frequency) | 55 | 93 | 113 | 90 | 73 | 29 | 5 |

I) Enter the dataset using dataframe command
II) Add a column cumulative frequency
III) Add a column of relative frequency (frequency/total frequency)
IV) Add a column of relative cumulative frequency (cumulative frequency/total frequency)
V) Plot cumulative frequency vs mid points

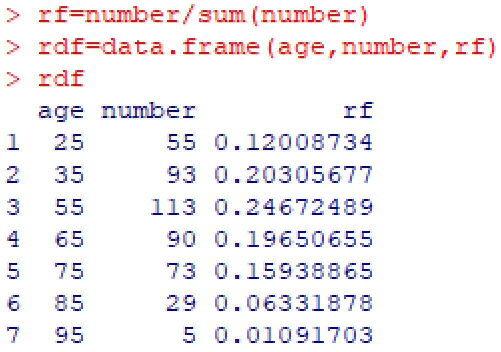
```
age=c(25,35,55,65,75,85,95)
number=c(55,93,113,90,73,29,5)
```



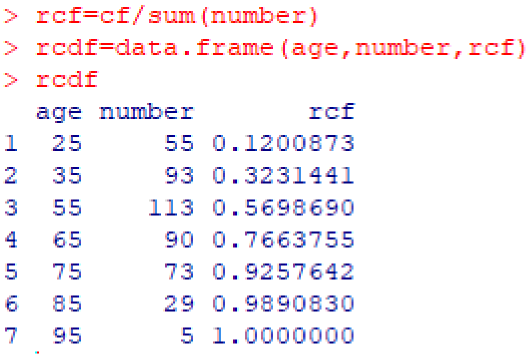
```
> df=data.frame(age,number)
> df
  age number
1  25     55
2  35     93
3  55    113
4  65     90
5  75     73
6  85     29
7  95      5
```



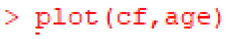
```
> cdf=data.frame(age,number,cf)
> cdf
  age number  cf
1  25     55  55
2  35     93 148
3  55    113 261
4  65     90 351
5  75     73 424
6  85     29 453
7  95      5 458
```



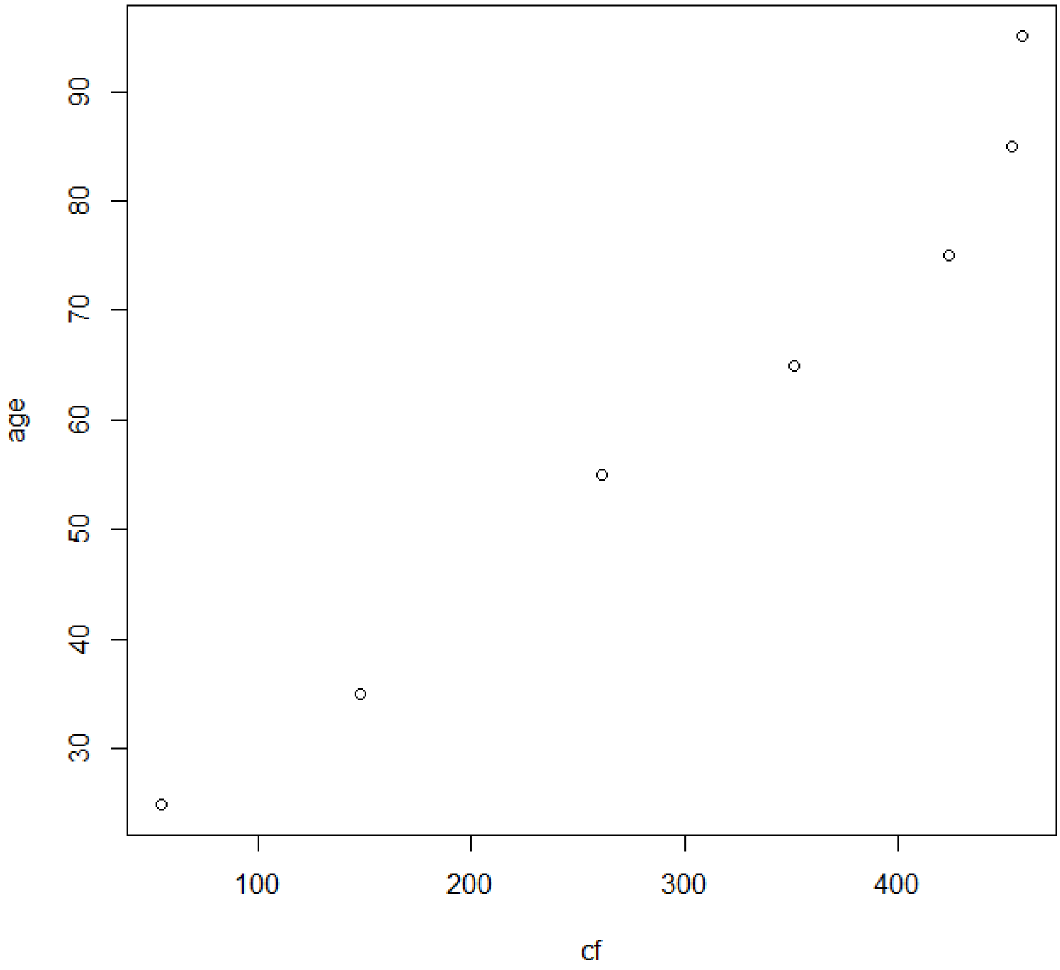
```
> rf=number/sum(number)
> rdf=data.frame(age,number,rf)
> rdf
  age number         rf
1  25     55 0.12008734
2  35     93 0.20305677
3  55    113 0.24672489
4  65     90 0.19650655
5  75     73 0.15938865
6  85     29 0.06331878
7  95      5 0.01091703
```



```
> rcf=cf/sum(number)
> rcdf=data.frame(age,number,rcf)
> rcdf
  age number       rcf
1  25     55 0.1200873
2  35     93 0.3231441
3  55    113 0.5698690
4  65     90 0.7663755
5  75     73 0.9257642
6  85     29 0.9890830
7  95      5 1.0000000
```



```
> plot(cf,age)
```



## Conclusion:

We have understood the basics of R such as Operators, built-in functions, Data types, Data manipulation in R and R packages for Data Science. We have also learnt about plotting functions