Tutorial and Laboratories
Week4

The purpose of this week's tutorial and lab exercises is to introduce you to other types of data structures in R programming language, which are essential to perform data analysis. These are Matrices, Arrays and Data Frames.

The "objectives" of this tutorial and lab are to:

1. Get yourself familiar with creating, accessing and operating on 'matrices/arrays'
2. Get yourself familiar with creating, accessing and operating on 'Data Frames'

Matrices and Arrays

Matrices and arrays are data structures that store elements with similar data types in 2-D and in n-D respectively. Matrices are special case of arrays with only two dimensions. In R language, arrays and matrices are nothing more than vectors with an additional attribute called 'dimension'.

- To create a matrix or an array, we specify the data to be ordered in the matrix or in the array, as well as the potential dimension of the order (e.g., number of rows, number of columns, etc.). Practice the following:

```
> mat <- matrix(11: 80, 8, 10)
> arr <- array(1:60, dim = c(3,4,5)
```

Exercise 1 (matrices and arrays) 

suppose you have the following two vectors:

```
v1 <- c(3, 9, -1, 4, 2, 6)
v2 <- c(5, 2, 0, 9, 3, 4)
```

Perform the following:

- Create a matrix **m1**, from the first vector where number of rows are 2,
- Create a matrix **m2**, from the second vector where number of rows are 2,
- Add the two matrices and display the result,
- Subtract the second matrix from the first one and display the result,
- Multiply the two matrices and print the results,
- Transpose the second matrix and assign the results to a third matrix called **m3,**
- Do matrix multiplication between m1 and m3 and print the results,
- Display the dimension of the output matrix from the previous step,
- Add another row to the matrix m2,
- Display the elements of the first row and the elements of first column of m1.

Data Frames

Data frames are the most common used data structures in R. A Data Frame is a series of records that are represented by rows (observations), and columns (variables). They are like

matrices in structure as they are also bi-dimensional. However, contrary to matrices, data frames may include data of different types.

- Creating Data Frame:
- Practice the following example:

```
> books <- data.frame(author=c("Reda", "John", "Edward", "Ben"),
+ year=c(2104, 2016, 2005, 2019),
+ publisher=c("Wiley", "Springer", "Sage", "International Books"))

> books

> books$author

> books["author"]

> books <- data.frame(author=c("Reda", "John", "Edward", "Ben"),
+ year=c(2104, 2016, 2005, 2019),
+ publisher=c("Wiley", "Springer", "Sage", "International Books"),
+ stringsAsFactors = FALSE)

> books$author
```

- What is the effect of using "`stringsAsFactors = FALSE`" in the last example?

Exercise 2 (Data Frames) 

- Enter the following into a vector named **Boxes.colour**. Remember to surround each piece of text with quotes.

  {purple, red, yellow, brown}
- Display the 2nd element in the vector.
- Enter the following into a vector with the name **boxes.weight**:

  {40, 30, 18, 23}
- Join the two vectors together using the ***data.frame()*** function to make a data frame named **boxes.info** with two columns and four rows. Call the first column 'colour' and the second one 'weight'.
  1. Print the data frame
  2. Print the contents of the third row
  3. Print the contents of the first column
  4. Print the content of the cell in in 4th row and 1st column.

Exercise 3 (Data Frames)

5. Create a data frame representing a tiny database of *movies*. It should contain the fields title, director, year, country, and at least three movies.
6. Create a second data frame of the same format as above but containing just one new movie.
7. Merge the two data frames using *rbind()* function.
8. Extract the titles of the merged data frame using the accessor operator, **$**
9. Try sorting the year using ***sort()***, what happens?
10. Replace the sort with ***order()***, what happens?

Please use the remaining time to discuss with your tutor any question you have from the previous tutorials or labs.