**Doing Math in ‘ R ‘ Part-2**

1. Consider A=matrix(c(2,0,1,3), ncol=2) and B=matrix(c(5,2,4,-1),ncol=2).

a) Find A+B

b) Find A-B

Solution: Matrix addition and Subtraction

a) To find A + B, We simply add the corresponding elements of the two matrices:

A <- matrix(c(2, 0, 1, 3), ncol = 2)

B <- matrix(c(5, 2, 4, -1), ncol = 2)

A + B

[,1] [,2]

[1,] 7 2

[2,] 5 2

b) To find A – B, We simply subtract the corresponding elements of the two matrices:

A <- matrix(c(2, 0, 1, 3), ncol = 2)

B <- matrix(c(5, 2, 4, -1), ncol = 2)

A - B

[,1] [,2]

[1,] -3 -2

[2,] -3 4

In RStudio, you can simply define the matrices A and B using the **‘matrix’** function, and then use

the **‘ - ’** and **‘ + ’** operators to perform the matrix operations. The resulting matrices will be

displayed in the console.

Graphical user interface, text, application

Description automatically generated

**2. Using the diag() function to build a matrix of size 4 with the following values in the diagonal 4,1,2,3.**

**Solution:** Matrix creation in R

In R, you can the ‘diag(0’ function to create a diagnol matrix with specific values in the diagonal. To create a matrix of size matrix of size 4 with the values 4, 1, 2, and 3 in the diagonal, you can use the following code:

matrix\_values <- c(4, 1, 2, 3)

diag\_matrix <- diag(matrix\_values)

Here, We first create a vector ‘matrix-values‘ with the value we want in the diagonal of the matrix. Then We use the ‘diag()’ function to create a matrix called ‘diag-matrix’ with the same dimensions as the length of the vector ‘matrix-values’ (Which is 4 in this case) and with the value of ‘matrix-values’ along the diagonal.

When this code run in R, the resulting matrix ‘diag-matrix’ will look like diag(c(4, 1, 2, 3))

[,1] [,2] [,3] [,4]

[1,] 4 0 0 0

[2,] 0 1 0 0

[3,] 0 0 2 0

[4,] 0 0 0 3

As We can see, the values 4, 1, 2, and 3 are placed along the diagnol of the matrix, all the non-diagonal elements are zero.

Graphical user interface, text, application, email

Description automatically generated

**3. Generate the following matrix:**

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

## [1,] 3 1 1 1 1

## [2,] 2 3 0 0 0

## [3,] 2 0 3 0 0

## [4,] 2 0 0 3 0

## [5,] 2 0 0 0 3

**Solution:** Diagonal Matrix in R

diag\_vec <- c(3, 3, 3, 3, 3)

m <- diag(diag\_vec)

m[1, 2:5] <- 1

m[2, 1] <- 2

m[2, 2] <- 3

m[3, 1] <- 2

m[3, 3] <- 3

m[4, 1] <- 2

m[4, 4] <- 3

m[5, 1] <- 2

m[5, 5] <- 3

m

This code generates a 5x5 matrix where the diagonal elements are all 3, and all other elements are initially 0.

The first line diag\_vec <- c(3, 3, 3, 3, 3) creates a vector of length 5 with all elements equal to 3.

The second line m <- diag(diag\_vec) creates a diagonal matrix with the diagonal values taken from diag\_vec. This means that m is initially a 5x5 matrix with all elements equal to 0 except for the diagonal, which is equal to 3.

The following lines modify the elements of m to produce the desired pattern

m[1, 2:5] <- 1 sets the elements in the first row, columns 2 to 5, equal to 1.

m[2, 1] <- 2 sets the element in the second row, first column, equal to 2.

m[2, 2] <- 3 sets the element in the second row, second column, equal to 3.

m[3, 1] <- 2 sets the element in the third row, first column, equal to 2.

m[3, 3] <- 3 sets the element in the third row, third column, equal to 3.

m[4, 1] <- 2 sets the element in the fourth row, first column, equal to 2.

m[4, 4] <- 3 sets the element in the fourth row, fourth column, equal to 3.

m[5, 1] <- 2 sets the element in the fifth row, first column, equal to 2.

m[5, 5] <- 3 sets the element in the fifth row, fifth column, equal to 3.

The resulting matrix m has the pattern described in the question prompt.

Graphical user interface, text, application

Description automatically generated