Unit-3

Matrices, Arrays and Data frames

A matrix is a two-dimensional data set that collects rows and columns. The matrix stores the data in rows and columns format. It is possible to access the data in the matrix easily.

How to create the matrix
R language offers various methods to create a matrix efficiently. By using these methods provided by R, it is easy to create the matrix. Some of the methods to create the matrix are:

Creating a matrix by using the function ‘matrix ()'

These is the one of the common method used by the users to create the matrices in R language. These function ‘matrix ()' can allows us to access the data elements and can able to specify the dimension of matrix.

Syntax

The basic syntax for creating a matrix in R is −

matrix(data, nrow, ncol, byrow, dimnames)

Following is the description of the parameters used −

- **data** is the input vector which becomes the data elements of the matrix.
- **nrow** is the number of rows to be created.
- **ncol** is the number of columns to be created.
- **byrow** is a logical clue. If TRUE then the input vector elements are arranged by row.
- **dimname** is the names assigned to the rows and columns.

Ex.

```
# Elements are arranged sequentially by row.
M <- matrix(c(3:14), nrow = 4, byrow = TRUE)
print(M)

# Elements are arranged sequentially by column.
N <- matrix(c(3:14), nrow = 4, byrow = FALSE)
print(N)

# Define the column and row names.
rownames = c("row1", "row2", "row3", "row4")
colnames = c("col1", "col2", "col3")

P <- matrix(c(3:14), nrow = 4, byrow = TRUE, dimnames =
list(rownames, colnames))
print(P)
```

Output

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

```
[1,]    3    4    5
[2,]    6    7    8
[3,]    9   10   11
[4,]   12   13   14
     [,1] [,2] [,3]
[1,]    3    7   11
[2,]    4    8   12
[3,]    5    9   13
[4,]    6   10   14
     col1 col2 col3
row1    3    4    5
row2    6    7    8
row3    9   10   11
row4   12   13   14
```

Accessing Specific Elements

Elements of a matrix can be accessed by using the column and row index of the element. We consider the matrix P above to find the specific elements below.

```
# Define the column and row names.
rownames = c("row1", "row2", "row3", "row4")
colnames = c("col1", "col2", "col3")

# Create the matrix.
P <- matrix(c(3:14), nrow = 4, byrow = TRUE, dimnames =
list(rownames, colnames))

# Access the element at 3rd column and 1st row.
print(P[1,3])

# Access the element at 2nd column and 4th row.
print(P[4,2])

# Access only the  2nd row.
print(P[2,])

# Access only the 3rd column.
print(P[,3])
```

Output

```
[1] 5
[1] 13
col1 col2 col3
```

```
  6   7   8
row1 row2 row3 row4
  5   8  11  14
```

Matrix Computations

Various mathematical operations are performed on the matrices using the R operators. The result of the operation is also a matrix.

The dimensions (number of rows and columns) should be same for the matrices involved in the operation.

Matrix Addition & Subtraction

```
# Create two 2x3 matrices.
matrix1 <- matrix(c(3, 9, -1, 4, 2, 6), nrow = 2)
print(matrix1)

matrix2 <- matrix(c(5, 2, 0, 9, 3, 4), nrow = 2)
print(matrix2)

# Add the matrices.
result <- matrix1 + matrix2
cat("Result of addition","\n")
print(result)

# Subtract the matrices
result <- matrix1 - matrix2
cat("Result of subtraction","\n")
print(result)
```

Output

```
[,1] [,2] [,3]
[1,]    3   -1    2
[2,]    9    4    6
     [,1] [,2] [,3]
[1,]    5    0    3
[2,]    2    9    4
Result of addition
     [,1] [,2] [,3]
[1,]    8   -1    5
[2,]   11   13   10
Result of subtraction
```

```
     [,1] [,2] [,3]
[1,]   -2   -1   -1
[2,]    7   -5    2
```

Matrix Multiplication & Division

```
# Create two 2x3 matrices.

matrix1 <- matrix(c(3, 9, -1, 4, 2, 6), nrow = 2)

print(matrix1)

matrix2 <- matrix(c(5, 2, 0, 9, 3, 4), nrow = 2)

print(matrix2)

# Multiply the matrices.

result <- matrix1 * matrix2

cat("Result of multiplication","\n")

print(result)

# Divide the matrices

result <- matrix1 / matrix2

cat("Result of division","\n")

print(result)
```

Ex.

```
[,1] [,2] [,3]
[1,]    3   -1    2
[2,]    9    4    6
     [,1] [,2] [,3]
[1,]    5    0    3
[2,]    2    9    4
Result of multiplication
     [,1] [,2] [,3]
[1,]   15    0    6
[2,]   18   36   24
```

```
Result of division
    [,1]      [,2]      [,3]
[1,]  0.6      -Inf 0.6666667
[2,]  4.5 0.4444444 1.5000000
```

Transpose of Matrix

The matrix transposition in R involves flipping a matrix over its diagonal, swapping the row and column indices of each elements. This can be done using the t() function.

Ex.

```
#craete matrix
M<-matrix (1:6,nrow=2,ncol=3)
print (“Original Matrix”)
print (M)
#Transpose of matrix
Transposed_mat<-t(M)
print (“Transposed matrix”)
print (Transposed_mat)
```

output-

```
Original Matrix
        [,1]    [,2]      [,3]
[1,]     1        3        5
[2,]     2        4        6
Transposed matrix
        [,1]     [,2]
[1,]     1        2
[2,]     3        4
[3,]     5        6
```

Where, t (M) returns the transpose of matrix, switching rows and columns.
If the original matrix is m*n the transposed matrix will have n*m.