21 Jan 2022, Saturday

R Programming Lab
Assignment 3 with Concepts

Matrices

Dimension	Homogenous	Heterogenous
1	Atomic vectors	Lists
2	Matrix	Data frame
≥ 1	Array	

- Matrices can only contain data of **one type** (like vectors).
- A matrix is a special array with **two dimensions**.

The basic syntax to create a matrix is given below:

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

```
data = the input element of a matrix given as a vector.
nrow = the number of rows to be created.
ncol = the number of columns to be created.
byrow = the row-wise arrangement of the elements instead of column-wise
dimnames = the names of columns or rows to be created.
```

1. Try the usage of **?matrix** or **help("matrix")**

2. Create a vector x=(1,2,3,4,5,6,7,8,9)
    a. Create a 3X3 matrix m using vector x.
    b. Fill elements of x by row.
    c. Check dimension of matrix m (hint: dim())
    d. Find the number of rows and number of columns (hint: nrow(), ncol())
    e. Find the number of elements in m

3. Create four matrices based on vectors of different types (double, integer, character, and logical). Investigate the objects.
    a. x1 <- matrix(seq(0, 4.5, length.out = 9), nrow = 3)   # double
    b. x2 <- matrix(1:9, nrow = 3)   # integer
    c. x3 <- matrix(LETTERS[1:9], nrow = 3)   # character
    d. x4 <- matrix(TRUE, nrow = 3, ncol = 3)   # logical

    Use is.matrix(), is.double(), is.integer(), and is.numeric() to check the type of the data of the matrix.

4. Try to create the following matrices. To do so, we need to specify data, the number of rows, and the number of columns.
    a. A matrix of dimension 5X5 which contains 5L (integer) everywhere.

    b. A matrix of dimension 10X1 which contains -100(numeric) everywhere.
    c. Check that the class of your result is c("matrix", "array")

5. Generate 50 random numbers using rnorm() with mean=50, sd=10.
    a. Create a large matrix called mat (dimension 10X5) with generated random numbers.
    b. Call head(mat) and head(mat, n = 2) to get the first 6 (default) or 2 rows only.
    c. Call tail(mat) and tail(mat, n = 3) to get the last 6 or 3 rows.
    d. Call summary(mat) and try to interpret the output. Do you see what happens? (Hint: numerical summary for each column individually)
    e. Try to answer the following questions:
        i. Get the largest and smallest values in the matrix (minimum and maximum).
        ii. What is the arithmetic mean (average), and what is the standard deviation of the entire matrix?
        iii. What is the sum of the entire matrix mat?

6. Arithmetic using Matrices and scalars:
   x <- matrix(1:4, ncol = 2)
    a. x + 2
    b. x * 2
    c. x / 2
    d. x ^ 2

7. Arithmetic using Matrices and Vectors:
   x <- matrix(1:4, ncol = 2)
   y <- c(10, 100)
    a. Compute x*y

8. Arithmetic using Matrices and Matrices:
   x <- matrix(c(1, 2, 3, 4), ncol = 2, nrow = 2)
   y <- matrix(c(10, 20, 30, 40), ncol = 2, nrow = 2)
    a. x + y
    b. x - y
    c. x * y
    d. x / y

9. Create a matrix x
   x <- matrix(c(1, 2, 3, 4), ncol = 2, nrow = 2)
    a. Transpose of x using trans(x).
    b. Diagonal elements of x using diag(x).
    c. Matrix multiplication using x %*% x. Check how this is different from x*x.

10. Set dimension names:

```
x <- matrix(data = 1:9, nrow = 3, ncol = 3)
```

   a. Check rownames(x); colnames(x); dimnames(x)
   b. Run

```
rownames(x) <- c("Row 1", "Row 2", "Row 3")
colnames(x) <- c("Col A", "Col B", "Col C")
```

      Print the matrix x.
      Check rownames(x); colnames(x); dimnames(x)

11. Create named matrix m

```
m <- matrix(data = 1:9, nrow = 3, ncol = 3,
            dimnames = list( c("Row 1", "Row 2", "Row 3"),
                             c("Col A",   "Col B", "Col C"))
           )
```

      Print the matrix m.
      Check rownames(m); colnames(m); dimnames(m)

12. Create a matrix by combining vectors.

```
# Generate vectors
x <- c( 5,  5,  5)
y <- c(11, 22, 33)

# Combine
z <- cbind(x, y)
z <- rbind(x, y)
```

13. Let's see if we try to column-bind vectors of different lengths. For simplicity, only use integer vectors of form 1:2 (vector of length 2) or 1:5 (vector of length 5).
Try to row-bind and/or column-bind vectors of length:
   a. 4 and 4
   b. 8 and 4
   c. 4 and 8
   d. 4 and 1
   e. 5 and 3
   f. 9 and 10

14. Combine matrices:

```
x1 <- matrix(1:6,     ncol = 2)
x2 <- matrix(101:106, ncol = 2)
```

   a. cbind(x1, x2)
   b. rbind(x1, x2)

15. Hands-on matrix subsetting. Try to answer the questions based on the following numeric matrix mat.

```
mat <- matrix(c(270, 100, 330, 340, 260, 160, 10, 310, 80, 50, 60, 190, 150,
110, 290, 220, 10, 350, 100, 0), nrow = 5)
```

a. What is the value of element mat[3, 2]?
b. What is the value of element mat[2, 4]?
c. What is the value of element mat[7], and how can we extract the same element using row and column indices?
d. What is the value of element mat[15], and how can we extract the same element using row and column indices?
e. mat[c(4, 2), 3]
(result is a vector of length two which contains the two elements ‘row4, column3 ’ and ‘row2, column3 ’)
f. mat[2, 1:3]
(elements for columns 1−3 of row2)
g. mat[c(2, 7, 12)]
(object mat is of dimension 5×4– the elements returned by mat[2, 1:3] are the elements 2, 7, and 12)
h. mat[1,] (gives us the “first row”, “all columns”)
i. mat[, 3] (gives us “all rows” from the “third column”)
j. mat[1, c(3, 4)] (elements of “row one, column three and four”)
k. mat[c(2, 4), c(3, 1)]
(get a matrix of dimension 2×2 with all elements from rows 2 and 4, which lie in columns 3 and 1 (four elements))
l. mat[mat>100] to get all elements in the matrix that are larger than 100.

16. Create the following matrix medals, which contains the “medal table” of the Skeleton contest at the winter olympics (up to 2018). Each row contains the number of gold, silver, and bronze medals for the top five countries.

	Gold	Silver	Bronze
United States	3	4	1
Great Britain	3	1	5
Canada	2	1	1
Russia	1	0	2
Switzerland	1	0	2

a. Find the number of gold medals Canada got using dimension names.
b. Find the number of gold, silver, and bronze medals for Switzerland.

c. Find the data of all countries which got more than one gold medal (> 1) and want to get the entire row out of the matrix.

17. Create the following named matrix *Students* about the age, size, and current semester of some students.

```
##           age height semester
## Peter      24   1.67        5
## Elif       30   1.93        3
## Leo        53   1.73        7
## Marcus     24   1.65        2
## Rob        24   1.71        2
```

a. Try sort(students[, "age"]).
   This will only sort this specific column and the age will no longer match the actual age of the students. Solution: Instead, we make use of order().
b. Try students[order(students[, "age"]),]
c. Sort by multiple columns, e.g., first order by "age", and then (in case two students have the same age), order by "height".

18. Try Matrix plot using matplot()

matplot() is plotting data columnwise. Each column will get a different color and line type starting with color/line type 1 for the first column (black), 2 for the second (red) and so on.

```
matplot(m, main = "Matrix Plot")
```