

R Matrices

Armand Tossou

9/3/2021

Creating a Matrix

```
# sequential numeric vector  
1:10
```

```
## [1] 1 2 3 4 5 6 7 8 9 10
```

```
v <- 1:10  
v
```

```
## [1] 1 2 3 4 5 6 7 8 9 10
```

```
matrix(v)
```

```
##      [,1]  
## [1,] 1  
## [2,] 2  
## [3,] 3  
## [4,] 4  
## [5,] 5  
## [6,] 6  
## [7,] 7  
## [8,] 8  
## [9,] 9  
## [10,] 10
```

```
matrix(v, nrow = 2) # create a 2x5 matrix
```

```
##      [,1] [,2] [,3] [,4] [,5]  
## [1,] 1 3 5 7 9  
## [2,] 2 4 6 8 10
```

```
matrix(1:12, byrow = FALSE, nrow = 4)
```

```
##      [,1] [,2] [,3]  
## [1,] 1 5 9  
## [2,] 2 6 10  
## [3,] 3 7 11  
## [4,] 4 8 12
```

Google vs Microsoft stock prices

combine the vectors:

```
goog <- c(450,451,452,445,468)
msft <- c(230,231,232,233,220)

stocks <- c(goog,msft) # combine the 2 vectors

# build a matrix out of this
stock.matrix <- matrix(stocks, byrow = TRUE, nrow = 2)

days <- c('Mon','Tue','Wed','Thu','Fri')
st.names <- c('GOOG','MSFT')

colnames(stock.matrix) <- days
rownames(stock.matrix) <- st.names

print(stock.matrix)
```

```
##      Mon Tue Wed Thu Fri
## GOOG 450 451 452 445 468
## MSFT 230 231 232 233 220
```

Matrix Arithmetic

```
mat <- matrix(1:25, byrow=TRUE, nrow=5)
mat
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]    1    2    3    4    5
## [2,]    6    7    8    9   10
## [3,]   11   12   13   14   15
## [4,]   16   17   18   19   20
## [5,]   21   22   23   24   25
```

```
print('-----')
```

```
## [1] "-----"
```

```
mat * 2 # multiply everything by 2
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]    2    4    6    8   10
## [2,]   12   14   16   18   20
## [3,]   22   24   26   28   30
## [4,]   32   34   36   38   40
## [5,]   42   44   46   48   50
```

```
print('-----')
```

```
## [1] "-----"
```

```
mat / 2 # divide everything by 2
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]  0.5  1.0  1.5  2.0  2.5
## [2,]  3.0  3.5  4.0  4.5  5.0
## [3,]  5.5  6.0  6.5  7.0  7.5
## [4,]  8.0  8.5  9.0  9.5 10.0
## [5,] 10.5 11.0 11.5 12.0 12.5
```

```
print('-----')
```

```
## [1] "-----"
```

```
mat ^ 2 # everything to the power of 2
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]    1    4    9   16   25
## [2,]   36   49   64   81  100
## [3,]  121  144  169  196  225
## [4,]  256  289  324  361  400
## [5,]  441  484  529  576  625
```

```
print('-----')
```

```
## [1] "-----"
```

```
1 / mat # reciprocal of everything
```

```
##      [,1]      [,2]      [,3]      [,4]      [,5]
## [1,] 1.00000000 0.50000000 0.33333333 0.25000000 0.20000000
## [2,] 0.16666667 0.14285714 0.12500000 0.11111111 0.10000000
## [3,] 0.09090909 0.08333333 0.07692308 0.07142857 0.06666667
## [4,] 0.06250000 0.05882353 0.05555556 0.05263158 0.05000000
## [5,] 0.04761905 0.04545455 0.04347826 0.04166667 0.04000000
```

```
print('-----')
```

```
## [1] "-----"
```

```
mat > 15 # everywhere the matrix is greater than 15
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,] FALSE FALSE FALSE FALSE FALSE
## [2,] FALSE FALSE FALSE FALSE FALSE
## [3,] FALSE FALSE FALSE FALSE FALSE
## [4,]  TRUE  TRUE  TRUE  TRUE  TRUE
## [5,]  TRUE  TRUE  TRUE  TRUE  TRUE
```

```
mat[mat > 15]
```

```
## [1] 16 21 17 22 18 23 19 24 20 25
```

```
print('-----')
```

```
## [1] "-----"
```

```
mat * mat # element by element multiplication
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]    1    4    9   16   25
## [2,]   36   49   64   81  100
## [3,]  121  144  169  196  225
## [4,]  256  289  324  361  400
## [5,]  441  484  529  576  625
```

```
print('-----')
```

```
## [1] "-----"
```

```
mat %*% mat # matrix multiplication
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]  215  230  245  260  275
## [2,]  490  530  570  610  650
## [3,]  765  830  895  960 1025
## [4,] 1040 1130 1220 1310 1400
## [5,] 1315 1430 1545 1660 1775
```

Matrix operations

```
colSums(stock.matrix)
```

```
## Mon Tue Wed Thu Fri
## 680 682 684 678 688
```

```
rowSums(stock.matrix)
```

```
## GOOG MSFT
## 2266 1146
```

```
rowMeans(stock.matrix)
```

```
## GOOG MSFT
## 453.2 229.2
```

```
colMeans(stock.matrix)
```

```
## Mon Tue Wed Thu Fri  
## 340 341 342 339 344
```

adding rows and columns

```
# add new row to matrix  
FB <- c(111,112,113,120,145)  
tech.stocks <- rbind(stock.matrix, FB)  
tech.stocks
```

```
##      Mon Tue Wed Thu Fri  
## GOOG 450 451 452 445 468  
## MSFT 230 231 232 233 220  
## FB   111 112 113 120 145
```

```
# add new columns to matrix  
print("-----")
```

```
## [1] "-----"
```

```
avg <- rowMeans(tech.stocks)  
tech.stocks <- cbind(tech.stocks,avg)  
tech.stocks
```

```
##      Mon Tue Wed Thu Fri   avg  
## GOOG 450 451 452 445 468 453.2  
## MSFT 230 231 232 233 220 229.2  
## FB   111 112 113 120 145 120.2
```

Matrix Selection and Indexing

```
mat <- matrix(1:50, byrow = TRUE, nrow = 5)  
mat
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]  
## [1,]    1    2    3    4    5    6    7    8    9   10  
## [2,]   11   12   13   14   15   16   17   18   19   20  
## [3,]   21   22   23   24   25   26   27   28   29   30  
## [4,]   31   32   33   34   35   36   37   38   39   40  
## [5,]   41   42   43   44   45   46   47   48   49   50
```

```
mat[1,] # select 1st row
```

```
## [1] 1 2 3 4 5 6 7 8 9 10
```

```
mat[,1] # select 1st column
```

```
## [1] 1 11 21 31 41
```

```
mat[1:3,] # first 3 rows
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
## [1,] 1    2    3    4    5    6    7    8    9    10
## [2,] 11   12   13   14   15   16   17   18   19   20
## [3,] 21   22   23   24   25   26   27   28   29   30
```

```
mat[1:2,1:3] # top left rectangle
```

```
##      [,1] [,2] [,3]
## [1,] 1    2    3
## [2,] 11   12   13
```

```
mat[,9:10] # last 2 columns
```

```
##      [,1] [,2]
## [1,] 9    10
## [2,] 19   20
## [3,] 29   30
## [4,] 39   40
## [5,] 49   50
```

```
mat[2:3,5:6] # center square
```

```
##      [,1] [,2]
## [1,] 15   16
## [2,] 25   26
```

Factor and Categorical Matrices

```
animal <- c('d','c','d','c','c')
id <- c(1:5)

factor(animal)
```

```
## [1] d c d c c
## Levels: c d
```

```
fact.ani <- factor(animal)

# assign orders, for ordinal matrix
ord.cat <- c('cold','med','hot')
temps <- c('cold','med','hot','hot','hot','cold','med')
fact.temps <- factor(temps,ordered = TRUE, levels=c('cold','med','hot'))
fact.temps
```

```
## [1] cold med hot hot hot cold med
## Levels: cold < med < hot
```

```
summary(fact.temps)
```

```
## cold med hot
## 2 2 3
```

```
summary(temps)
```

```
## Length Class Mode
## 7 character character
```

Exercise

Ex 1: Create 2 vectors A and B, where A is (1,2,3) and B is (4,5,6). With these vectors, use the cbind() or rbind() function to create a 2 by 3 matrix from the vectors. You'll need to figure out which of these binding functions is the correct choice.

```
A <- c(1,2,3)
B <- c(4,5,6)

C <- rbind(A,B)
C
```

```
## [,1] [,2] [,3]
## A 1 2 3
## B 4 5 6
```

Ex 2: Create a 3 by 3 matrix consisting of the numbers 1-9. Create this matrix using the shortcut 1:9 and by specifying the nrow argument in the matrix() function call. Assign this matrix to the variable mat

```
mat <- matrix(1:9, byrow = FALSE, nrow = 3)

mat
```

```
## [,1] [,2] [,3]
## [1,] 1 4 7
## [2,] 2 5 8
## [3,] 3 6 9
```

Ex 3: Confirm that mat is a matrix using is.matrix()

```
is.matrix(mat)
```

```
## [1] TRUE
```

Ex 4: Create a 5 by 5 matrix consisting of the numbers 1-25 and assign it to the variable mat2. The top row should be the numbers 1-5.

```
mat2 <- matrix(1:25, byrow = TRUE, nrow = 5)
```

```
mat2
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]    1    2    3    4    5
## [2,]    6    7    8    9   10
## [3,]   11   12   13   14   15
## [4,]   16   17   18   19   20
## [5,]   21   22   23   24   25
```

Ex 5: Using indexing notation, grab a sub-section of mat2 from the previous exercise that looks like this:

```
mat2[2:3,2:3]
```

```
##      [,1] [,2]
## [1,]    7    8
## [2,]   12   13
```

Ex 6: Using indexing notation, grab a sub-section of mat2 from the previous exercise that looks like this:

```
mat2[4:5,4:5]
```

```
##      [,1] [,2]
## [1,]   19   20
## [2,]   24   25
```

Ex 7: What is the sum of all the elements in mat2?

```
sum(mat2)
```

```
## [1] 325
```

Ex 8: Ok time for our last exercise! Find out how to use runif() to create a 4 by 5 matrix consisting of 20 random numbers (4*5=20).

```
mat3 <- matrix(runif(20,0,1),nrow = 4)
```

```
mat3
```

```
##      [,1]      [,2]      [,3]      [,4]      [,5]
## [1,] 0.9410243 0.63293404 0.6862726 0.2630185 0.9171872
## [2,] 0.4811859 0.91988090 0.9455829 0.9404818 0.8097879
## [3,] 0.3812187 0.43272383 0.8314928 0.6979177 0.9089417
## [4,] 0.1189288 0.01540271 0.3959108 0.9433805 0.3002794
```

Note: Install this library to enable document knitting into PDF:

```
tinytex::install_tinytex()
```