

Matrices

- Matrices can be created with the `matrix()` function, specifying all elements (column-by-column) as well as the number of rows and number of columns.

```
> A <- matrix(1:12, nr = 3, nc = 4)
> A
```

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

You may also specify the rows (or columns) as vectors, and then combine them into a matrix using the `rbind()` or `cbind()` functions.

```
> a <- c(1,2,3)
> a
```

```
[1] 1 2 3
```

```
> b <- c(10, 20, 30)
> b
```

```
[1] 10 20 30
```

```
> c <- c(100, 200, 300)
> c
```

```
[1] 100 200 300
```

```
> d <- c(1000, 2000, 3000)
> d
```

```
[1] 1000 2000 3000
```

```
> B <- rbind(a, b, c, d) # Create a matrix with vectors a, b, c, d as rows.
> B
```

```
      [,1] [,2] [,3]
a       1    2    3
b      10   20   30
c     100  200  300
d    1000 2000 3000
```

The name of the function `rbind()` stands for “row bind”; indeed, it binds rows together.

```
> C <- cbind(a, b, c, d) # Create a matrix with vectors a, b, c, d as
      columns.
> C
```

```
      a  b  c  d
[1,] 1 10 100 1000
[2,] 2 20 200 2000
[3,] 3 30 300 3000
```

The name of the function `cbind()` stands for “column bind”; indeed, it binds columns together.

- Use the function `t()` to transpose a matrix:

```
> TC <- t(C) # Transpose matrix C
> TC == B # The resulting matrix should be precisely B.
```

```
  [,1] [,2] [,3]
a TRUE TRUE TRUE
b TRUE TRUE TRUE
c TRUE TRUE TRUE
d TRUE TRUE TRUE
```

- To select a subset of a matrix, use the square brackets and specify rows before the comma, and columns after the comma.

```
> C[1:2,] # Select rows 1 and 2.
```

```
  a b c d
[1,] 1 10 100 1000
[2,] 2 20 200 2000
```

```
> C[,c(1,3)] # Select columns 1 and 3.
```

```
  a c
[1,] 1 100
[2,] 2 200
[3,] 3 300
```

```
> C[1:2,c(1,3)] # Select column 1 and 3 from rows 1 and 2.
```

```
  a c
[1,] 1 100
[2,] 2 200
```

- Note what happens if you multiply two matrices:

```
> B*t(C)
```

```
  [,1] [,2] [,3]
a 1e+00 4e+00 9e+00
b 1e+02 4e+02 9e+02
c 1e+04 4e+04 9e+04
d 1e+06 4e+06 9e+06
```

The matrices are multiplied element by element.

- Maybe you've learned about *matrix multiplication* in high school. In R matrix multiplication is performed with the operator `%*%`. Remember that, in matrix multiplication, the order matters!

```
> B%*%C
```

```
  a b c d
a 14 140 1400 1.4e+04
b 140 1400 14000 1.4e+05
c 1400 14000 140000 1.4e+06
d 14000 140000 1400000 1.4e+07
```

```
> C%*%B
```

```
  [,1] [,2] [,3]
[1,] 1010101 2020202 3030303
[2,] 2020202 4040404 6060606
[3,] 3030303 6060606 9090909
```

- You may apply a function to the rows or columns of a matrix using the `apply()` function.

```
> C
      a  b  c  d
[1,] 1 10 100 1000
[2,] 2 20 200 2000
[3,] 3 30 300 3000
```

```
> sum(C)
[1] 6666
```

```
> apply(C, 1, sum) # Sums of the rows.
[1] 1111 2222 3333
```

```
> apply(C, 2, sum) # Sums of the columns.
      a  b  c  d
[1,] 6 60 600 6000
```

If it is not clear to you what the “1” and “2” stand for in the above example, simply type “`?apply`” to look it up in the documentation.