

Vectores y matrices en R

1. Vectores

- Los vectores y matrices permiten almacenar un conjunto de elementos. Los vectores están compuestos por una fila y n columnas. Los elementos almacenados pueden ser numéricos o no numéricos (en este caso irán entre comillas). Para nombrar un vector:

```
> a=c(8,67,exp(7),sqrt(33),8.97)
> Nombres=c('Pepito','Luisito','Jaimito')
> B=0 #También B=c(0)
```

- La longitud del vector es el número de elementos que contiene. Puede ser importante en caso de realizar bucles. El comando length() almacena la longitud de un vector.

```
> J=c(1,2,3,4,5)
> n=length(J)
> n
[1] 5
```

- Para tomar un determinado elemento del vector:

```
> D=c(89,76,5678,33,2)
> D[3]
[1] 5678
```

Los corchetes indican la posición dentro del vector. En este caso, se quiere tomar el tercer valor del vector D, que corresponde con 5678.

- Para operar con vectores, deben tener la misma longitud. Solo se puede operar con vectores que contengan valores numéricos.

```
> a=c(1,2,3,4)
> b=c(5,6,7,8)
> c=c(9,10,'hola',12)
> a+b
[1] 6 8 10 12
> a-b
[1] -4 -4 -4 -4
> a+c
Error in a + c : argumento no-numérico para operador binario
> b*a
[1] 5 12 21 32
> b%*%a
      [,1]
[1,]    70
> a/b
[1] 0.2000000 0.3333333 0.4285714 0.5000000
> a^b
[1] 1 64 2187 65536
```

Es importante diferenciar el producto de vectores elemento a elemento del producto escalar. Para el producto de vectores elemento a elemento se utiliza el apóstrofe (A*B), y para el producto escalar (suma de los productos de cada elemento de la misma posición) se utiliza %*%

- Los vectores también pueden sumarse, restarse, multiplicarse, dividirse o elevarse exponencialmente a un escalar (componente a componente).

```
> a=c(1,2,3,4)
> a+1
[1] 2 3 4 5
> 2-a
[1] 1 0 -1 -2
> 3*a
[1] 3 6 9 12
> a/7
[1] 0.1428571 0.2857143 0.4285714 0.5714286
> a^10
[1] 1 1024 59049 1048576
```

2. Matrices

- Las matrices permiten almacenar valores en n filas y m columnas. Para generar una matriz, los dos métodos más usados son:
 - Utilizando el comando matrix():

```
> M=matrix(c(1,2,3,4,5,6),nrow=2,ncol=3)
> M
      [,1] [,2] [,3]
[1,]  1    3    5
[2,]  2    4    6
```

Con c() indicamos los valores que queremos almacenar, con nrow el número de filas y, con ncol, el número de columnas. Los valores se van almacenando de forma que primero se rellena la primera columna, luego la segunda, y así sucesivamente. También se pueden almacenar elementos no numéricos (tienen que ir entre comillas).

- Utilizando los comandos rbind o cbind:

```
> A=c(1,2,3,4)
> B=c(5,6,7,8)
> C=c(45,76,90,22)
> M1=rbind(A,B,C)
> M2=cbind(A,B,C)
> M1
      [,1] [,2] [,3] [,4]
A      1    2    3    4
B      5    6    7    8
C     45   76   90   22
> M2
      A B C
[1,] 1 5 45
[2,] 2 6 76
[3,] 3 7 90
[4,] 4 8 22
```

rbind ordena los vectores por filas y cbind ordena los vectores por columnas. Es necesario que todos los vectores tengan la misma longitud.

- Operaciones con matrices:
 - Para sumar, restar, multiplicar elemento a elemento, dividir elemento a elemento y elevar exponencialmente elemento a elemento dos matrices, éstas deben tener el mismo número de filas y columnas:

```

> A=matrix(c(1,2,3,4,5,6),nrow=2,ncol=3)
> B=matrix(c(7,8,9,10,11,12),nrow=2,ncol=3)
> A+B
      [,1] [,2] [,3]
[1,]   8  12  16
[2,]  10  14  18
> B-A
      [,1] [,2] [,3]
[1,]   6   6   6
[2,]   6   6   6
> A*B
      [,1] [,2] [,3]
[1,]   7  27  55
[2,]  16  40  72
> A/B
      [,1] [,2] [,3]
[1,] 0.1428571 0.3333333 0.4545455
[2,] 0.2500000 0.4000000 0.5000000
> B^A
      [,1] [,2] [,3]
[1,]   7  729 161051
[2,]  64 10000 2985984

```

- Para el producto matemático de dos matrices, el número de columnas de la primera matriz debe coincidir con el número de filas de la segunda. Se realiza con %*%:

```

> C=matrix(c(1,2,3,8,9,78),nrow=3,ncol=2)
> D=matrix(c(65,7,987,1,4.54,4,87,3),nrow=2,ncol=4)
> C%*%D
      [,1] [,2] [,3] [,4]
[1,] 121  995  36.54 111
[2,] 193 1983  45.08 201
[3,] 741 3039 325.62 495

```

- Otras operaciones con matrices son la matriz traspuesta, el determinante de una matriz y la matriz inversa. Para realizar el determinante y la matriz inversa, la matriz inicial tiene que ser cuadrada.

```

> Z=matrix(c(1,2,3,8),nrow=2,ncol=2)
> t(Z)
      [,1] [,2]
[1,]   1   2
[2,]   3   8
> det(Z)
[1] 2
> solve(Z)
      [,1] [,2]
[1,]   4 -1.5
[2,]  -1  0.5

```