

Fundamentos de programación: Laboratorio en R

(<https://programacion-en-r.webnode.es/>)

2. Vectores y matrices

Los **vectores** y las **matrices** **almacenan grupos de valores** y la diferencia entre ambos es que en el **vector** estos valores se encuentran todos en **una única fila**. Para escribir los valores que compondrán el vector o la matriz utilizaremos `c()`.

1. Vectores

Los valores de los vectores se introducen en R con el comando `c(valor 1, valor 2, ... , valor n)`.

Algunos ejemplos de vectores pueden ser:

- `u=c(3,25,10,7,115,48)`
- `v=c("Alemania","Argentina","China")`
- `w=c(7,9,"Madrid","Roma")`

Vemos que las **componentes** de los vectores pueden ser **números** (es el caso de `u`), cadenas de **caracteres** (`v`) o **ambos** (`w`).

Además, en el vector `u` vemos un vector con 6 componentes, en `v` uno de 3 y en `w` uno de 4; para **buscar** una **componente dentro de un vector** se escribe el comando `vector[componente buscada]`, por **ejemplo**, la cuarta componente de `u` escribiríamos: `u [4]`.

OPERACIONES

1. Operaciones con las componentes de los vectores.

En primer lugar, la **suma** de dos componentes de un vector, como la suma de la segunda y tercera componente del vector `u`, almacenando el resultado en la variable `a`:

```
a=u[2]+u[3]
```

Del mismo modo, realizamos las operaciones explicadas anteriormente, como son la **resta**, la **multiplicación**, el **producto**, el **cociente**, etc.

2. Operaciones con los propios vectores

Tomando dos vectores exclusivamente numéricos, por ejemplo `t` y `s`, de 4 componentes cada uno:

- `t+s` $\Rightarrow$ suma de vectores
- `t-s` $\Rightarrow$ resta de vectores
- `2*t+s` $\Rightarrow$ multiplicación de escalar por vector y suma de vectores
- `t*s` $\Rightarrow$ multiplicación de vectores

Todas las operaciones se refieren a la primera componente de un vector con la primera componente del otro, la segunda componente de uno menos la segunda del otro, etc.

Es decir, la suma `t+s` da **4 valores**, siendo el primero la suma de `t[1]` y `s[1]`. De esta manera, `t*s` **no** es el **producto escalar**, dará 4 resultados, siendo el primero de ellos el producto de `t[1]` y `s[1]`, y así sucesivamente.

Para el **producto escalar** escribiremos: `t%*%s`

2. Matrices

Las **matrices** son objetos en los que, al igual ocurre con los que los vectores, se pueden **almacenar** muchos **datos**, con la diferencia de que éstos lo hacen distribuyéndose **en filas** y en **columnas**.

En R, podemos **definir una matriz** de dos maneras distintas:

1. Utilizando el comando **matrix(c(datos), nrow= i, ncol= j)**, siendo "i" el número de filas y "j" el número de columnas. Algunos **ejemplos** serían:

```
A=matrix(c(2,5,3,8,11,7),nrow=3,ncol=2)
A      #Tras ejecutar el programa, obtenemos
      [,1] [,2]
[1,]  2   8
[2,]  5   1
[3,]  3   7
B=matrix(c(14,23,27,19),nrow=2,ncol=2)
B      #Tras ejecutar el programa obtenemos
      [,1] [,2]
[1,] 14 27
[2,] 23 19
```

Primero, se **enumeran** los **valores** que componen la matriz utilizando el comando **c(datos)** y, después, se escribe el **número de filas (nrow)** y el **número de columnas (ncol)**. Una vez llevados a cabo estos pasos, se engloban los comandos anteriores en **matrix()**, tal que nos queda: **matrix(c(datos), nrow= i, ncol= j)**

2. Utilizando los comandos **rbind** o **cbind**, que forman matrices a partir de la unión de vectores combinando sus **filas** o **columnas** respectivamente. Por **ejemplo**:

```
u=c(5,25,18,7)
v=c(16,2,23,4)
A=rbind(u,v)
A      #Tras ejecutar este programa en R, obtenemos:
      [,1] [,2] [,3] [,4]
u    5   25  18   7
v   16    2  23   4

t=c(2,5,3,8)
s=c(11,7,24,5)
B=cbind(t,s)
B      #Tras ejecutar este programa en R, obtenemos:
      t s
[1,] 2 11
[2,] 5  7
[3,] 3 24
[4,] 8  5
```

OPERACIONES

Podemos realizar operaciones con las matrices:

- **Suma.** A+B
- **Resta.** A-B

- **Producto.** $A \% * \% B$

Para **multiplicar** matrices **elemento a elemento** se utilizaría $*$.

- Otras **operaciones**:

- **t(A):** transpuesta de la matriz A.
- **solve(A,b):** solución del sistema de ecuaciones $AX=B$.
- **solve(A):** inversa de la matriz A.
- **svd(A):** descomposición en valores singulares.
- **qr(A):** descomposición QR.
- **eigen(A):** valores y vectores propios.
- **diag(b):** matriz diagonal (b es un vector).
- **diag(A):** matriz diagonal (A es una matriz).