

APUNTES 2ª PRÁCTICA FUNDAMENTOS DE PROGRAMACIÓN (1ºBIOTECNOLOGÍA)

VECTORES Y MATRICES

La segunda práctica del primer curso del grado de Biotecnología, tiene como objetivo el aprendizaje y manejo de funciones básicas de la programación con el lenguaje R, relacionadas con vectores, matrices y sus respectivas operaciones.

- Vectores: Los vectores sirven para almacenar valores en una sola fila. La estructura es `c()`. Y los valores que contiene el vector pueden ser números o no numéricos.

EJ:

`vv= c(8,9,10,-1/3,sin(7)) (Vector)`

`suma_comp= vv[1]+vv[4]`

Los corchetes se utilizan para las componentes del vector y los paréntesis para asignaciones (posiciones dentro del vector)

`print(suma_comp)`

Significa lo mismo que poner directamente `suma_comp` para que ejecute la suma

`w= c(10.3/2,"pepe") (Nuevo vector)`

Tenemos 2 componentes numéricos y una no numérica

`w[1]+w[2]`

Da error al ejecutarlo ya que al ser la tercera componente no numérica todas las toma como no numéricas y no deja operarlas

`z= c(1.6,7) (Nuevo vector)`

`z[1]+z[2]`

Aquí si deja ejecutarlo pues todas sus componentes son numéricas.

`za= c(1.3,42.5/3.log(7)) (Nuevo vector)`

El `log()` es logaritmo neperiano NO logaritmo en base 10 en el caso de querer poner el logaritmo en base 10 es `log10()`.

`za+vv` Suma de vectores

`za*vv` Multiplicación de los vectores por componentes (NO producto escalar)

`za*%*%vv` Producto escalar

- Matrices: Las matrices sirven para almacenar muchos valores simultáneamente en filas y columnas dependiendo de su dimensión. La estructura es `matrix()` .

EJ:

`A= matrix(c(3,4,5,6),nrow=2, ncol=2) (Matriz)`

Definimos la matriz A, escribe el vector `c(3,4,5,6)` por columnas, es decir:

`3=A11`

`4=A21`

`5=A12`

`6=A22`

Los comandos `nrow` y `ncol` definen el número de filas y columnas de la matriz.

`B=matrix(c(sin(8),2*pi,sqrt(57),log10(33),1/3,0.9),nrow=3,ncol=2) (Nueva matriz)`

El comando `sqrt()` significa=raíz cuadrada

`B=matrix(c(sin(8),2*pi,sqrt(57),log10(33),1/3,0.9),nrow=2,ncol=3) (Nueva matriz)`

Cambio filas y columnas, en vez de ser una matriz 3x2 pasa a ser una matriz 2x3

En el caso de que se pongan más filas o columnas de los valores que tiene el vector, se repiten valores.

`B=matrix(c(sin(8),2*pi,sqrt(57),log10(33),1/3,0.9),nrow=3,ncol=2) (Nueva matriz)`

Volvemos a definir la matriz de 3x2 ya que es con la que nos interesa trabajar.

Para definir la matriz traspuesta de B la almacenaremos en una nueva matriz C:

`C=t(B)`

`B*C` Multiplicación elemento a elemento de la matriz. En este caso nos da error ya que las dimensiones no son compatibles (tienen que tener la misma dimensión)

`D=B%%C` Producto de matrices (filas*columnas) D va a ser una matriz de 3x3

`det(D)` Comando para calcular el determinante de una matriz (Si la solución sale como `e-13` significa :10 elevado al número en este caso -13)

`diag(D)` Comando que calcula la diagonal de una matriz.

Podemos almacenar el valor de la diagonal en una variable como `pepin`.

`pepin=diag(D)`

TODA OPERACIÓN O VALOR SE PUEDE ALMACENAR EN UNA VARIABLE

COMANDOS : `rbind` y `cbind`

Sirven para construir matrices a partir de vectores.

`v1= c(1,4,5) (vector 1)`

`v2= c(0.8,97,13/4) (vector 2)`

Formamos la matriz AA:

`AA=rbind(v1,v2)` Por filas (`rbind`) → MATRIZ DE 2*3

Formamos la matriz BB:

`BB=cbind(v1,v2)` Por columnas (`cbind`) → MATRIZ DE 3*2

Otras operaciones con matrices:

- **t(A)** : transpuesta de la matriz A.
- **det(A)** : determinante 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.
- **eigen(A)** : valores y vectores propios.
- **diag(b)** : matriz diagonal (b es un vector).
- **diag(A)** : matriz diagonal (A es una matriz).

EJERCICIO A REALIZAR POR EL ALUMNO:

Dados los siguientes vectores y matrices:

$u=c(-10, \exp(2))$

$v= c(3, 25)$

$A= \text{matrix}(c(5/3, 10^2, \exp(2), 0.5), \text{nrow}=2, \text{ncol}=2)$

$B= \text{matrix}(c(-6, -3/8, 12, 1), \text{nrow}=2, \text{ncol}=2)$

*El número e se representa como $\exp()$

REALICE:

1. suma los vectores u y v
2. resta los vectores u y v
3. calcula el producto escalar de los vectores
4. Construye una matriz C cuyas columnas sean los vectores u y v
5. Construye una matriz C cuyas filas sean los vectores u y v
6. Sma las matrices A y B
7. Multiplica las matrices A y B
8. Multiplica elemento a elemento las matrices A y B
9. Resuelve el sistema de ecuaciones $Az=b$ siendo $b=c(2, 1)$
10. Obtén la inversa de A y almacenarla en la matriz A1
11. Calcula la transpuesta de B y almacenarla en BT
12. Producto de las matrices A y BT
13. Valores propios de la matriz BT

SOLUCIÓN:

1. Suma los vectores u y v

$u+v$

2. Resta los vectores u y v

$u-v$

3. Calcula el producto escalar de los vectores

$u \% \% v$

4. Construir una matriz C cuyas columnas sean los vectores u y v

$C = \text{cbind}(u, v)$

C

5. Construir una matriz R cuyas filas sean los v y u

$R = \text{rbind}(u, v)$

R

6. Sumar las matrices A y B

$A+B$

7. Multiplicar las matrices A y B

$A \% \% B$

8. Multiplicar elemento a elemento las matrices A y B

$A * B$

9. Resolver el sistema de ecuaciones $Az=b$ siendo $b= c(2,1)$

$\text{solve}(A, b)$

10. Obtener la inversa de A y almacenarla en la matriz A1

$A1 = \text{solve}(A)$

A1

11. Traspuesta de B y almacenarla en BT

$BT = t(B)$

BT

12. Producto de las matrices A y BT

$A \% \% BT$

13. Valores propios de la matriz BT

$\text{eigen}(BT)$