

# Explicación de la segunda práctica de R

## 1. Matrices:

- **Comando de una matriz:** Para escribir matrices, se asigna un nombre a la matriz seguido de un =, y después se escribe la matriz como tal, con el formato: `matrix(c(x,y,z,t))`; siendo x,y,z,t los elementos de la matriz separados por comas. Después, separado por comas, se escribe `nrow=n`, siendo n el número de filas de la matriz y `ncol=m`, siendo m el número de columnas de la matriz. Por último, se cierra el paréntesis.

El número de filas y columnas deben ser coherentes con el número de elementos asignados a la matriz. El número de elementos dividido entre el número de filas debe dar el número de columnas y viceversa.

Es decir que, si llamamos al número de elementos asignados a la matriz "d",  $m=d/n$  y  $n=d/m$

Por ejemplo:

```
A=matrix(c(1,2,3,4,5,6,7,8,9), nrow = 3, ncol = 3)
```

En este caso, los 9 elementos entre las 3 filas dan 3, que es el número de columnas. Esos 9 elementos entre las 3 columnas también dan 3, que son el número de filas.

- **Comandos rbind y cbind:** Combinan vectores en matrices por filas y por columnas, respectivamente. Primero creas unos vectores, con el mismo número de elementos y luego los combinas de la forma que gustes, dando nombre a las matrices.

Por ejemplo:

```
v=c(2,3,7,9) ; w=c(4,8,12,24)
```

- Combinación por filas:

```
P=rbind(v,w)
```

- Combinación por columnas:

```
R=cbind(v,w)
```

- **Iniciación de una matriz a 0:** En lugar de añadir elementos a la matriz, se asigna un 0 que será el valor de todos los elementos de esa matriz

Por ejemplo:

```
B=matrix(c(0),nrow=5,ncol=7)
```

## 2. Data.frame:

El comando data.frame sirve para almacenar datos de diferentes tipos, como pueden ser elementos numéricos, letras, texto, en una única matriz. Para ello, damos un nombre a esta matriz, y seguido del = ponemos data.frame y entre paréntesis los elementos que queramos incluir.

Por ejemplo:

Creamos 3 vectores con elementos de distintos tipos

```
Paises=c("España", "Suiza", "Austria")
```

```
Extension=c(54e4,12e3,15e3)
```

```
lobos=c(2e4,3e4,5e4)
```

Ahora creamos nuestra matriz data.frame con estos 3 vectores

```
Pa=data.frame(Paises,Extension,lobos)
```

Si queremos ver el resultado, escribimos el nombre de nuestra matriz en la consola

```
Pa
```

## 3. Ejercicio 1:

Además de hablar sobre matrices, en esta práctica realizamos un ejercicio sobre bucles. A continuación, el enunciado del ejercicio con su resolución y explicación.

Enunciado:

### Ejercicios con bucles

Dados los vectores:  $v=(12, -3, 5, 18.7)$  y  $w=(12, 0.25, 77, \exp(2))$

- 1) Obtener la suma de los dos vectores mediante bucles y comprobar empleando  $v+w$ .
- 2) Obtener la suma de las componentes del vector  $v$  y almacenarlos en SumaC.
- 3) Realizar el producto escalar de ambos vectores **mediante bucles** y DESPUÉS comprobar empleando  $\%*\%$ .
- 4) Multiplicar ambos vectores componente a componente **mediante bucles**
- 5) Realizar la operación:  $z_j=v_j+2w_j$ ,  $j = 1, \dots, \text{length}(v)$  **mediante bucles**
- 6) Construir una tabla, data.frame que contenga:

'Suma'	<i>Resultado del apartado 2)</i>
'Producto escalar'	<i>Resultado del apartado 3)</i>

- 1) Podemos definir los vectores  $v$  y  $w$  y sumarlos tal que  $v+w$ , o también podemos crear otro vector  $u[i]$  que vaya sumando componente a componente gracias a un bucle for.

```
v=c(12, -3,5,18.7)
w=c(12,0.25,77,exp(2))
u=0
for(i in 1:length(v)){
u[i]=v[i]+w[i]}
u
v+w
```

- 2) Las componentes del vector  $v$  se van sumando a  $\text{SumaC}$  que inicialmente vale 0, hasta que se han sumado todas las componentes y  $\text{SumaC}$  resulta con la suma total.

```
SumaC=0
for(i in 1:length(v)){
  SumaC=SumaC+v[i]}
SumaC
```

- 3) El producto escalar consiste en la suma de los valores obtenidos al multiplicar componente a componente ambos vectores, por lo que hacemos algo similar al apartado anterior, donde a una variable “ $p$ ” que inicialmente vale 0 le vamos sumando lo que obtenemos tras multiplicar componente a componente los vectores, por lo que resultaría en un número escalar (ej: 7). Esto se consigue mediante el bucle for.  
→ `%%` sirve para calcular directamente el producto escalar, por lo que nos tendría que dar lo mismo.

```
p=0
for(i in 1:length(v)){
p=p+v[i]*w[i]}
p
v%%w
```

- 4) Hacemos como anteriormente, pero esta vez queremos obtener un vector resultado de la multiplicación componente a componente de los vectores  $v$  y  $w$ . Así, creamos un vector  $m[i]$  cuya primera componente sea la multiplicación de las primeras componentes de  $v$  y  $w$ , la segunda componente la multiplicación de las segundas componentes de  $v$  y  $w$ , etc.

```
m=0
for(i in 1:length(v)){
m[i]=v[i]*w[i]}
m
```

- 5) Sustituimos la [i] que estábamos utilizando anteriormente por una [j] (lo pide el enunciado) para designar a las componentes de los vectores y realizamos la operación. El vector z[j] almacena vectorialmente el resultado de sumar v con 2w componente a componente.

```
z=0
for(j in 1:length(v)){
z[j]=v[j]+2*w[j]}
z
```

- 6) Realizamos una tabla con data.frame donde una columna sean los nombres y otro columna los resultados correspondientes tal y como se muestra debajo.

```
nombres=c("Suma","Producto escalar")
resultado=c(SumaC,p)
data.frame(nombres,resultado)
```

**DEBERÍA TODO QUEDAR ASÍ:**

```
> # PRÁCTICA 2
> # EJERCICIO 1: ejercicios con bucles
> # 1)
> v=c(12,-3,5,18.7)
> w=c(12,0.25,77,exp(2))
> u=0
> for(i in 1:length(v)){
+ u[i]=v[i]+w[i]}
> u
[1] 24.00000 -2.75000 82.00000 26.08906
> v+w
[1] 24.00000 -2.75000 82.00000 26.08906
> # 2)
> SumaC=0
> for(i in 1:length(v)){
+ SumaC=SumaC+v[i]}
> SumaC
[1] 32.7
> # 3)
> p=0
> for(i in 1:length(v)){
+ p=p+v[i]*w[i]}
> p
[1] 666.4253
> v%*%w
      [,1]
[1,] 666.4253
> # 4)
> m=0
> for(i in 1:length(v)){
+ m[i]=v[i]*w[i]}
> m
[1] 144.0000 -0.7500 385.0000 138.1753
> # 5)
> z=0
> for(j in 1:length(v)){
+ z[j]=v[j]+2*w[j]}
> z
[1] 36.00000 -2.50000 159.00000 33.47811
> # 6)
> nombres=c("Suma","Producto escalar")
> resultado=c(SumaC,p)
> data.frame(nombres,resultado)
      nombres resultado
1      Suma    32.7000
2 Producto escalar 666.4253
```

## 4. Ejercicio 2:

El último ejercicio de esta práctica se basó en los bucles anidados. Adjunto imagen del enunciado, así como su resolución y explicación. Se emplearán los vectores del ejercicio anterior.

Enunciado:

1. Construir una matriz A1 de manera que los vectores v, w sean sus filas.
2. Construir una matriz A2 de manera que los vectores v, w sean sus columnas.
3. Multiplicar, empleando bucles, ambas matrices, obteniendo una matriz C
4. Verificar el resultado obtenido previamente empleando %\*%
5. Inventar una matriz 2x2 y llamarla D.
6. Sumar, mediante bucles, las matrices C y D.
7. Multiplicar las matrices C y D elemento a elemento.

```
Bucles  for (i in 1:nrow(A1)){
        for (j in 1:ncol(A2)){
```

1) Podemos hacer esto de dos maneras:

```
# Definir los vectores v y w
```

```
v=c(12,-3,5,18.7)
```

```
w=c(12,0.25,77,exp(2))
```

```
# Construir la matriz A1 con v y w como filas
```

```
A1=matrix(c(v,w), nrow=2, byrow=TRUE) # byrow=TRUE indica que la matriz
debe llenarse por filas
```

```
A1
```

```
# Otra forma
```

```
# Definir los vectores v y w
```

```
v=c(12,-3,5,18.7)
```

```
w=c(12,0.25,77,exp(2))
```

```
# Utilizar rbind para combinar v y w en filas y crear la matriz A1
```

```
A1=rbind(v,w)
```

```
A1
```

2) Hacemos exactamente lo mismo que en el anterior apartado, pero esta vez, en la opción (1), nrow pasa a ser ncol, y en la opción (2) rbind es cbind. No definimos los vectores porque ya lo hemos hecho anteriormente.

```
# Construir la matriz A2 con v y w como columnas
```

```
A2=matrix(c(v,w), ncol=2)
```

```
A2
```

```
# Otra forma
```

```
# Utilizar cbind para combinar v y w en filas y crear la matriz A2
```

```
A2=cbind(v,w)
```

```
A2
```

**3)**

```
# Las dimensiones de C deben corresponderse con las dimensiones de la
matriz producto de la multiplicación entre A1 y A2
```

```
C=matrix(0, nrow=nrow(A1), ncol=ncol(A2))
```

```
# Multiplicar A1 por A2 utilizando bucles
```

```
for (i in 1:nrow(A1)) {
  for (j in 1:ncol(A2)) {
    for (k in 1:ncol(A1)) {
      C[i, j]=C[i, j]+A1[i, k]*A2[k, j]}}}
C
```

```
# Otra forma
```

```
m=nrow(A1)
```

```
n=ncol(A2)
```

```
p=ncol(A1)
```

```
C=matrix(c(0),nrow=m,ncol=n)
```

```
for(i in 1:m){
  for(j in 1:n){
    for(k in 1:p){
      C[i,j]=C[ij]+A1[ik]*A2[kj]}}}
C
```

```
C
```

**4) Puedo verificar el anterior resultado empleando %\*\*%**

```
C=A1%**A2
```

```
C
```

**5) Me invento una matriz D de 2x2**

```
D=matrix(c(1,2,3,4), nrow=2, ncol=2)
```

```
D
```

**6) Sumamos C y D creando una matriz E de la siguiente manera:**

```
m=nrow(C)
```

```
n=ncol(C)
```

```
E=matrix(0,nrow=m,ncol=n)
```

```
for(i in 1:m){
```

```
for(j in 1:n){
```

```
E[i,j]=C[i,j]+D[i,j]}
```

```
E
```

- 7) En este caso basta con multiplicar las matrices así:  $C*D$ , ya que si quisiésemos obtener un número escalar producto de la suma de la multiplicación componente a componente sería repetir el apartado 3).**

```
F=C*D
```

```
F
```

## DEBERÍA QUEDAR ASÍ:

```
> # EJERCICIO 2
> # 1)
> # Definir los vectores v y w
> v=c(12,-3,5,18.7)
> w=c(12,0.25,77,exp(2))
> # Construir la matriz A1 con v y w como filas
> A1=matrix(c(v,w), nrow=2, byrow=TRUE) #byrow=TRUE indica que la matriz debe llenarse por filas
> A1
      [,1] [,2] [,3] [,4]
[1,]  12 -3.00  5 18.700000
[2,]  12  0.25 77  7.389056
> # Otra forma
> v=c(12,-3,5,18.7)
> w=c(12,0.25,77,exp(2))
> # Utilizar rbind para combinar v y w en filas y crear la matriz A1
> A1=rbind(v,w)
> A1
      [,1] [,2] [,3] [,4]
v  12 -3.00  5 18.700000
w  12  0.25 77  7.389056
> # 2)
> # Construir la matriz A1 con v y w como filas
> A2=matrix(c(v,w), ncol=2)
> A2
      [,1] [,2]
[1,] 12.0 12.000000
[2,] -3.0  0.250000
[3,]  5.0 77.000000
[4,] 18.7  7.389056
> # Otra forma
> # Utilizar cbind para combinar v y w en filas y crear la matriz A2
> A2=cbind(v,w)
> A2
      v      w
[1,] 12.0 12.000000
[2,] -3.0  0.250000
[3,]  5.0 77.000000
[4,] 18.7  7.389056
> # 3)
> # C debe tener las dimensiones adecuadas
> C=matrix(0, nrow=nrow(A1), ncol=ncol(A2))
> # Multiplicar A1 por A2 utilizando bucles
> for (i in 1:nrow(A1)) {
+   for (j in 1:ncol(A2)) {
+     for (k in 1:ncol(A1)) {
+       C[i, j]=C[i, j]+A1[i, k]*A2[k, j]}
+   }
+ }
> C
      [,1] [,2]
[1,] 527.6900 666.4253
[2,] 666.4253 6127.6607
> # Otra forma
> m=nrow(A1)
> n=ncol(A2)
> p=ncol(A1)
> C=matrix(0,nrow=m,ncol=n)
> for(i in 1:m){
+   for(j in 1:n){
+     for(k in 1:p){
+       C[i, j]=C[i, j]+A1[i, k]*A2[k, j]}
+   }
+ }
> C
      [,1] [,2]
[1,] 527.6900 666.4253
[2,] 666.4253 6127.6607
> # 4)
> C=A1%A2
> C
      v      w
v 527.6900 666.4253
w 666.4253 6127.6607
> # 5)
> D=matrix(c(1,2,3,4), nrow=2, ncol=2)
> D
      [,1] [,2]
[1,]  1  3
[2,]  2  4
> # 6)
> m=nrow(C)
> n=ncol(C)
> E=matrix(0,nrow=m,ncol=n)
> for(i in 1:m){
+   for(j in 1:n){
+     E[i, j]=C[i, j]+D[i, j]}
+ }
> E
      [,1] [,2]
[1,] 528.6900 669.4253
[2,] 668.4253 6131.6607
> # 7)
> F=C*D
> F
      v      w
v 527.690 1999.276
w 1332.851 24510.643
>
```