

## ✓ LaTeX y RMarkdown

Esta tarea se ha trabajado en un entorno de vscode

### ✓ Pregunta 1

Realiza los siguientes productos de matrices siguiente en R:

Producto punto  $A \cdot B$

```
1 # Definir las matrices A y B
2 A <- matrix(c(4, 2, 3, 4, 3, 2, 2, 0, 1, 0, 2, 0, 1, 0, 4, 0), nrow = 4, ncol = 4, byrow = TRUE)
3 B <- matrix(c(4, 3, 2, 1, 3, 2, 0, 1, 2, 1, 3, 1, 1, 0, 1, 2), nrow = 4, ncol = 4, byrow = TRUE)
4
5 # A * B
6 A_B <- A %*% B
7 print('A * B')
8 print(A_B)
```

Producto punto  $B \cdot A$

```
1 # B * A
2 B_A <- B %*% A
3 print('B * A')
4 print(B_A)
```

Transpuesta  $t$  del resultado del producto punto  $A \cdot B$

```
1 # (A * B)^t
2 A_B_t <- t(A_B)
3 print('(A * B)^t')
4 print(A_B_t)
```

Resultado de  $B^t \cdot A$

```
1 # B^t * A
2 B_t_A <- t(B) %*% A
3 print('B^t * A')
4 print(B_t_A)
5
```

Resultado de  $(A \cdot B)^{-1}$

```
1 # (A * B)^-1 (si existe)
2 print('(A * B)^-1 (si existe)')
3 A_B_inv <- try(solve(A_B), silent = TRUE)
4 if (is.matrix(A_B_inv)) {
5   print(A_B_inv)
6 } else {
7   cat("La matriz (A * B) no es invertible.\n")
8 }
```

Resultado de  $(A^{-1}) \cdot B^t$

```

1
2 # A^-1 * B^t (si existe)
3 print('A^-1 * B^t (si existe)')
4 A_inv <- try(solve(A), silent = TRUE)
5 B_t <- t(B)
6 if (is.matrix(A_inv)) {
7   A_inv_B_t <- A_inv %*% B_t
8   print(A_inv_B_t)
9 } else {
10  cat("La matriz A no es invertible.\n")
11 }

```

## ▼ Pregunta 2

Definimos el vector *dni* con los números aleatorios **DNI**:

```
dni <- c(5, 4, 2, 0, 1, 5, 6, 7)
```

Ahora, para calcular el vector dni al cuadrado, usamos operaciones vectorizadas en *R*:

```
1 dni_cuadrado <- dni^2
```

Para calcular la raíz cuadrada del vector dni, utilizamos la función *sqrt*:

```
1 sqrt(dni)
```

Por último, para calcular la suma de todas las cifras del vector dni, podemos usar la función *sum*:

```
sum(dni)
```

## ▼ Pregunta 3

Primero, definimos el vector name con las letras de tu nombre y apellido:

```
1 name <- c("S", "E", "B", "A", "S", "T", "I", "A", "N", "M", "U", "Ñ", "O", "Z")
```

Ahora, para calcular el subvector que contenga solo tu nombre, podemos usar la indexación en *R* :

```
1 nombre <- name[1:9]
```

Para calcular el subvector que contenga solo tu apellido, también usamos indexación:

```
1 apellido <- name[10:14]
```

Ahora, para ordenar alfabéticamente el vector apellido, utilizamos la función *sort*:

```
1 apellido_ordenado <- sort(apellido)
2 print(apellido_ordenado)
```

Por último, para crear una matriz con el vector *name*, utilizamos la función *matrix*:

```
1 matriz_name <- matrix(name, nrow = 3, byrow = TRUE)
2 print(matriz_name)
```

