

Tarea - Matrices: R y Python

Ramon Ceballos

21/2/2021

Resolved todas estas preguntas haciendo uso de todos los lenguajes. No es necesario que repitáis cada pregunta 3 veces, simplemente con utilizar los 3 lenguajes a lo largo de toda la tarea ya os sirve para practicar.

Pregunta 1

Definimos las matrices del ejercicio con R y Python.

```
A = matrix(c(0,1,-2,2,3,-1,1,-1,5), nrow = 3, byrow = TRUE)
B = matrix(c(1,-1,2,1,2,-2,2,-2,-1,2,1,2), nrow = 3, byrow = TRUE)
C = matrix(c(2,0,1,-4), nrow = 4)
```

```
import numpy as np
A2 = np.array([[0,1,-2],[2,3,-1],[1,-1,5]])
B2 = np.array([[1,-1,2,1],[2,-2,2,-2],[-1,2,1,2]])
C2 = np.array([[2],[0],[1],[-4]])
```

Operaciones en R:

```
# A*B
A %*% B
```

```
##      [,1] [,2] [,3] [,4]
## [1,]    4   -6    0   -6
## [2,]    9  -10    9   -6
## [3,]   -6   11    5   13
```

```
# B*C
B %*% C
```

```
##      [,1]
## [1,]    0
## [2,]   14
## [3,]   -9
```

```
# Bt
t(B)
```

```
##      [,1] [,2] [,3]
## [1,]    1    2   -1
## [2,]   -1   -2    2
## [3,]    2    2    1
## [4,]    1   -2    2
```

```
# Bt * A
t(B) %*% A
```

```
##      [,1] [,2] [,3]
## [1,]    3    8   -9
## [2,]   -2   -9   14
## [3,]    5    7   -1
## [4,]   -2   -7   10
```

```
# Ct * Bt
t(C) %*% t(B)
```

```
##      [,1] [,2] [,3]
## [1,]    0   14   -9
```

Operaciones en Python:

```
# A*B
A2.dot(B2)
```

```
# B*C
```

```
## array([[ 4, -6,  0, -6],
##        [ 9, -10,  9, -6],
##        [-6, 11,  5, 13]])
```

```
B2.dot(C2)
```

```
# Bt
```

```
## array([[ 0],
##        [14],
##        [-9]])
```

```
B2.transpose()
```

```
# Bt * A
```

```
## array([[ 1,  2, -1],
##        [-1, -2,  2],
##        [ 2,  2,  1],
##        [ 1, -2,  2]])
```

```
B2.transpose().dot(A2)
```

```
# Ct * Bt
```

```
## array([[ 3,  8, -9],  
##        [-2, -9, 14],  
##        [ 5,  7, -1],  
##        [-2, -7, 10]])
```

```
C2.transpose().dot(B2.transpose())
```

```
## array([[ 0, 14, -9]])
```

Pregunta 2

Defino las matrices en R.

```
A = matrix(c(0,1,0,1),nrow = 2, byrow = TRUE)  
B = matrix(c(-1,-1,0,0), nrow = 2, byrow = TRUE)
```

Demostración en R de que $(A + B)^2 \neq A^2 + 2AB + B^2$.

```
library(Biodem)
```

```
mtx.exp((A+B),2) == mtx.exp(A,2) + 2*A %*% B+ mtx.exp(B,2)
```

```
##      [,1] [,2]  
## [1,] TRUE FALSE  
## [2,] TRUE  TRUE
```

Se verifica la proposición.

Demostarción en R de que $(A + B)^3 = A^3 + 3A^2B + 3AB^2 + B^3$.

```
mtx.exp((A+B),3) == mtx.exp(A,3) + 3 * mtx.exp(A,2) %*% B + 3 * A %*% mtx.exp(B,2) + mtx.exp(B,3)
```

```
##      [,1] [,2]  
## [1,] TRUE TRUE  
## [2,] TRUE TRUE
```

Se verifica la proposición.

Pregunta 3

Defino las matrices del ejercicio en Python.

```
import numpy as np  
A = np.array([[ -1, 1, -2], [1, 1, 0], [2, 1, 1]])  
B = np.array([[ -1, 2, 3, 4, 5], [1, 2, 1, 3, 2], [0, 4, 4, 7, 7]])
```

Calculo los rangos de las matrices definidas con Python.

```
np.linalg.matrix_rank(A)
```

```
## 2
```

```
np.linalg.matrix_rank(B)
```

```
## 2
```

Pregunta 4

Defino las matrices en R.

```
A = matrix(c(1,2,0,0,1,-1,0,0,2), nrow = 3, byrow = TRUE)
B = matrix(c(1i,0,0,0,0,1,1,1i,0), nrow = 3, byrow = TRUE)
```

Calculo la inversa de cada matriz definida en R.

```
solve(A)
```

```
##      [,1] [,2] [,3]
## [1,]    1   -2 -1.0
## [2,]    0    1  0.5
## [3,]    0    0  0.5
```

```
solve(B)
```

```
##      [,1] [,2] [,3]
## [1,] 0-1i 0+0i 0+0i
## [2,] 1+0i 0+0i 0-1i
## [3,] 0+0i 1+0i 0+0i
```