Tarea 4

Matrices con R, Python y Octave

Daniel Eduardo Macias Estrada

9/8/2020

Resuelve todas estas preguntas haciendo uso de todos los lenguajes.

Pregunta 1

Consideremos las matrices

$$A = \begin{pmatrix} 0 & 1 & -2 \\ 2 & 3 & -1 \\ 1 & -1 & 5 \end{pmatrix} \qquad B = \begin{pmatrix} 1 & -1 & 2 & 1 \\ 2 & -2 & 2 & -2 \\ -1 & 2 & 1 & 2 \end{pmatrix} \qquad C = \begin{pmatrix} 2 \\ 0 \\ 1 \\ -4 \end{pmatrix}$$

Realizar las operaciones siguientes:

- $A \cdot B$
- $B \cdot C$
- B^t
- $B^t \cdot A$
- $C^t \cdot B^t$

Resultados

$A \cdot B$

R

```
A = rbind(c(0,1,-2), c(2,3,-1), c(1,-1,5))
B = rbind(c(1,-1,2,1), c(2,-2,2,-2), c(-1,2,1,2))
A%*%B
```

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

Python

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

```
## [[  4  -6   0  -6]
##  [  9 -10   9  -6]
##  [ -6  11   5  13]]
```

Octave

```
A = [0 1 -2; 2 3 -1; 1 -1 5]
B = [1 -1 2 1; 2 -2 2 -2; -1 2 1 2]
A*B
```

```
## A =
##
##    0   1  -2
##    2   3  -1
##    1  -1   5
##
## B =
##
##    1  -1   2   1
##    2  -2   2  -2
##   -1   2   1   2
##
## ans =
##
##     4   -6    0   -6
##     9  -10    9   -6
##    -6   11    5   13
```

$B \cdot C$

R

```
C = rbind(c(2),c(0),c(1),c(-4))
B%*%C
```

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

Python

```
C = np.array([[2],[0],[1],[-4]])
print(B.dot(C))
```

```
## [[ 0]
##  [14]
##  [-9]]
```

Octave

```
B = [1 -1 2 1; 2 -2 2 -2; -1 2 1 2]
C = [2; 0; 1; -4]
B*C
```

```
## B =
##
##    1  -1   2   1
##    2  -2   2  -2
##   -1   2   1   2
##
## C =
##
```

```
##    2
##    0
##    1
##   -4
##
## ans =
##
##     0
##    14
##    -9
```

B^t

R

```r
t(B)
```

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

Python

```python
print(B.transpose())
```

```
## [[ 1  2 -1]
##  [-1 -2  2]
##  [ 2  2  1]
##  [ 1 -2  2]]
```

Octave

```octave
B = [1 -1 2 1; 2 -2 2 -2; -1 2 1 2]
B'
```

```
## B =
##
##    1  -1   2   1
##    2  -2   2  -2
##   -1   2   1   2
##
## ans =
##
##    1   2  -1
##   -1  -2   2
##    2   2   1
##    1  -2   2
```

$B^t \cdot A$

R

```r
t(B)%*%A
```

```
##      [,1] [,2] [,3]
## [1,]    3    8   -9
## [2,]   -2   -9   14
```

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

Python

```
B.transpose().dot(A)
```

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

Octave

```
A = [0 1 -2; 2 3 -1; 1 -1 5]
B = [1 -1 2 1; 2 -2 2 -2; -1 2 1 2]
B'*A
```

```
## A =
##
##    0   1  -2
##    2   3  -1
##    1  -1   5
##
## B =
##
##    1  -1   2   1
##    2  -2   2  -2
##   -1   2   1   2
##
## ans =
##
##     3    8   -9
##    -2   -9   14
##     5    7   -1
##    -2   -7   10
```

$C^t \cdot B^t$

R

```
t(C)%*%t(B)
```

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

Python

```
C.transpose().dot(B.transpose())
```

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

Octave

```
B = [1 -1 2 1; 2 -2 2 -2; -1 2 1 2]
C = [2; 0; 1; -4]
C'*B'
```

```
## B =
##
```

```
##    1  -1   2   1
##    2  -2   2  -2
##   -1   2   1   2
##
## C =
##
##    2
##    0
##    1
##   -4
##
## ans =
##
##     0   14   -9
```

Pregunta 2

Considera las matrices

$$A = \begin{pmatrix} 0 & 1 \\ 0 & 1 \end{pmatrix} \qquad B = \begin{pmatrix} -1 & -1 \\ 0 & 0 \end{pmatrix}$$

Demostrar que

$$(A+B)^2 \neq A^2 + 2AB + B^2$$

pero que en cambio

$$(A+B)^3 = A^3 + 3A^2B + 3AB^2 + B^3$$

Resultados

$(A+B)^2 \neq A^2 + 2AB + B^2$

R

```r
library(Biodem)
A = rbind(c(0,1),c(0,1))
B = rbind(c(-1,-1),c(0,0))

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

```
##      [,1] [,2]
## [1,]    1    0
## [2,]    0    1
```

```r
D
```

```
##      [,1] [,2]
## [1,]    1    2
## [2,]    0    1
```

```r
C == D
```

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

Python

```python
A = np.array([[0,1],[0,1]])
B = np.array([[-1,-1],[0,0]])

C = np.linalg.matrix_power(A+B,2)
D = np.linalg.matrix_power(A,2) + 2*(A.dot(B)) + np.linalg.matrix_power(B,2)
print(C)
```

```
## [[1 0]
##  [0 1]]
```

```python
print(D)
```

```
## [[1 2]
##  [0 1]]
```

```python
C == D
```

```
## array([[ True, False],
##        [ True,  True]])
```

Octave

```octave
A = [0 1; 0 1]
B = [-1 -1; 0 0]

C = (A+B)^2
D = A^2 + 2*A*B + B^2

C == D
```

```
## A =
##
##    0   1
##    0   1
##
## B =
##
##   -1  -1
##    0   0
##
## C =
##
##    1   0
##    0   1
##
## D =
##
##    1   2
##    0   1
##
```

```
## ans =
##
##    1  0
##    1  1
```

$(A+B)^3 = A^3 + 3A^2B + 3AB^2 + B^3$

R

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

```
##      [,1] [,2]
## [1,]   -1    0
## [2,]    0    1
```

```r
D
```

```
##      [,1] [,2]
## [1,]   -1    0
## [2,]    0    1
```

```r
C == D
```

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

Python

```python
C = np.linalg.matrix_power(A+B,3)
D = (np.linalg.matrix_power(A,3)
    + 3*(np.linalg.matrix_power(A,2).dot(B))
    + 3*(A.dot(np.linalg.matrix_power(B,2)))
    + np.linalg.matrix_power(B,3))

print(C)
```

```
## [[-1  0]
##  [ 0  1]]
```

```python
print(D)
```

```
## [[-1  0]
##  [ 0  1]]
```

```python
C == D
```

```
## array([[ True,  True],
##        [ True,  True]])
```

Octave

```
A = [0 1; 0 1]
B = [-1 -1; 0 0]

C = (A+B)^3
D = A^3 + 3*A^2*B + 3*A*B^2 + B^3

C == D
```

```
## A =
##
##    0   1
##    0   1
##
## B =
##
##   -1  -1
##    0   0
##
## C =
##
##   -1   0
##    0   1
##
## D =
##
##   -1   0
##    0   1
##
## ans =
##
##   1  1
##   1  1
```

Pregunta 3

Calcula el rango de las siguientes matrices

$$A = \begin{pmatrix} -1 & 1 & -2 \\ 1 & 1 & 0 \\ 2 & 1 & 1 \end{pmatrix} \qquad B = \begin{pmatrix} -1 & 2 & 3 & 4 & 5 \\ 1 & 2 & 1 & 3 & 2 \\ 0 & 4 & 4 & 7 & 7 \end{pmatrix}$$

Resultados

R

```r
A = rbind(c(-1,1,-2),c(1,1,0),c(2,1,1))
B = rbind(c(-1,2,3,4,5),c(1,2,1,3,2),c(0,4,4,7,7))

arank = qr(A)$rank
brank = qr(B)$rank

sprintf("El rango de la matriz A es: %s", arank)
```

```
## [1] "El rango de la matriz A es: 2"
```

```r
sprintf("El rango de la matriz B es: %s", brank)
```

```
## [1] "El rango de la matriz B es: 2"
```

Python

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

print("El rango de la matriz A es: " + str(np.linalg.matrix_rank(A)))
```

```
## El rango de la matriz A es: 2
```

```
print("El rango de la matriz B es: " + str(np.linalg.matrix_rank(B)))
```

```
## El rango de la matriz B es: 3
```

Octave

```
A = [-1 1 -2; 1 1 0; 2 1 1]
B = [-1 2 3 4 5; 1 2 1 2 3; 0 4 4 7 7]

disp("El rango de la matriz A es:")
disp(rank(A))

disp("El rango de la matriz B es:")
disp(rank(B))
```

```
## A =
##
##   -1   1  -2
##    1   1   0
##    2   1   1
##
## B =
##
##   -1   2   3   4   5
##    1   2   1   2   3
##    0   4   4   7   7
##
## El rango de la matriz A es:
##  2
## El rango de la matriz B es:
##  3
```

Pregunta 4

Calcula la inversa de las siguientes matrices

$$A = \begin{pmatrix} 1 & 2 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 2 \end{pmatrix} \qquad B = \begin{pmatrix} i & 0 & 0 \\ 0 & 0 & 1 \\ 1 & i & 0 \end{pmatrix}$$

Resultados

R

```
A = rbind(c(1,2,0),c(0,1,-1),c(0,0,2))
B = rbind(c(0+1i, 0+0i, 0+0i), c(0+0i, 0+0i, 1+0i), c(1+0i, 0+1i, 0+0i))

sprintf("La inversa de la matriz A: %s", solve(A))
```

```
## [1] "La inversa de la matriz A: 1"   "La inversa de la matriz A: 0"
## [3] "La inversa de la matriz A: 0"   "La inversa de la matriz A: -2"
## [5] "La inversa de la matriz A: 1"   "La inversa de la matriz A: 0"
## [7] "La inversa de la matriz A: -1"  "La inversa de la matriz A: 0.5"
## [9] "La inversa de la matriz A: 0.5"
```

```r
sprintf("La inversa de la matriz B: %s", solve(B))
```

```
## [1] "La inversa de la matriz B: 0-1i" "La inversa de la matriz B: 1+0i"
## [3] "La inversa de la matriz B: 0+0i" "La inversa de la matriz B: 0+0i"
## [5] "La inversa de la matriz B: 0+0i" "La inversa de la matriz B: 1+0i"
## [7] "La inversa de la matriz B: 0+0i" "La inversa de la matriz B: 0-1i"
## [9] "La inversa de la matriz B: 0+0i"
```

Python

```python
A = np.array([[1,2,0],[0,1,-1],[0,0,2]])
B = np.array([[complex(0,1), 0, 0],[0, 0, 1],[1, complex(0,1), 0]], dtype = complex)

print("La inversa de la matriz A: ")
```

```
## La inversa de la matriz A:
```

```python
np.linalg.inv(A)
```

```
## array([[ 1. , -2. , -1. ],
##        [ 0. ,  1. ,  0.5],
##        [ 0. ,  0. ,  0.5]])
```

```python
print("La inversa de la matriz B: ")
```

```
## La inversa de la matriz B:
```

```python
np.linalg.inv(B)
```

```
## array([[0.-1.j, 0.+0.j, 0.+0.j],
##        [1.+0.j, 0.+0.j, 0.-1.j],
##        [0.+0.j, 1.+0.j, 0.+0.j]])
```