Universidade Federal da Bahia
Programa de Pós-Graduação em Ciência de Dados e Big Data
Carlos Magno Santos Ribeiro de Brito

Prof Dra. Majela Penton Machado

Respostas da primeira lista de exercício

Questão 01) Em cada caso, encontra a matriz $A = [a_{ij}]$ de tamanho 4×4 cujas entradas satisfazem:

a) $a_{ij} = i + j$

$$A = \begin{bmatrix} 2 & 3 & 4 & 5 \\ 3 & 4 & 5 & 6 \\ 4 & 5 & 6 & 7 \\ 5 & 6 & 7 & 8 \end{bmatrix} \tag{1}$$

b) $a_{ij} = i^{j-i}$

$$A = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1/2 & 1 & 2 & 4 \\ 1/9 & 1/3 & 1 & 3 \\ 1/64 & 1/16 & 1/4 & 1 \end{bmatrix} \tag{2}$$

Questão 02) Considere as seguintes matrizes:

$$A = \begin{bmatrix} 2 & 0 \\ 6 & 7 \end{bmatrix}, \quad B = \begin{bmatrix} 0 & 4 \\ 2 & -8 \end{bmatrix}, \quad C = \begin{bmatrix} -6 & 9 & -7 \\ 7 & -3 & -2 \end{bmatrix}$$

$$D = \begin{bmatrix} -6 & 4 & 0 \\ 1 & 1 & 4 \\ -6 & 0 & 6 \end{bmatrix}, \quad E = \begin{bmatrix} 6 & 9 & -9 \\ -1 & 0 & -4 \\ -6 & 0 & -1 \end{bmatrix}$$

Efetue as seguintes operações ou justifique porque elas não podem ser realizadas:

a) $AB - BA$

$$AB = \begin{bmatrix} 0 & 8 \\ 14 & -32 \end{bmatrix}, \quad BA = \begin{bmatrix} 24 & 28 \\ -44 & -56 \end{bmatrix}$$

Assim sendo:

$$AB - BA = \begin{bmatrix} -24 & -20 \\ 58 & 24 \end{bmatrix} \tag{3}$$

b) $2C - D$

$$2C = 2 \cdot \begin{bmatrix} -6 & 9 & -7 \\ 7 & -3 & -2 \end{bmatrix} = \begin{bmatrix} -12 & 18 & -14 \\ 14 & -6 & -4 \end{bmatrix} - \begin{bmatrix} -6 & 4 & 0 \\ 1 & 1 & 4 \\ -6 & 0 & 6 \end{bmatrix}$$

Assim sendo, **<u>não</u>** é possível efetuar essa operação, já que uma matrix é de dimensão 2×3 ($2C = M_{i,j}$) e outra, 3×3 ($D_{i,j}$). A subtração só é possível com i e j iguais em ambas as matrizes.

c) $(2D^T - 3E^T)^T$

$$2D^T = 2 \cdot \begin{bmatrix} -6 & 4 & 0 \\ 1 & 1 & 4 \\ -6 & 0 & 6 \end{bmatrix} = \begin{bmatrix} -12 & 8 & 0 \\ 2 & 2 & 8 \\ -12 & 0 & 12 \end{bmatrix}^T = \begin{bmatrix} -12 & 2 & -12 \\ 8 & 2 & 0 \\ 0 & 8 & 12 \end{bmatrix}$$

$$3E^T = 3 \cdot \begin{bmatrix} 6 & 9 & -9 \\ -1 & 0 & -4 \\ -6 & 0 & -1 \end{bmatrix} = \begin{bmatrix} -18 & 27 & -27 \\ -3 & 0 & -12 \\ -18 & 0 & -3 \end{bmatrix}^T = \begin{bmatrix} 18 & -3 & -18 \\ 27 & 0 & 0 \\ -27 & -12 & -3 \end{bmatrix}$$

$$(2D^T - 3E^T)^T = \begin{bmatrix} -30 & -19 & 27 \\ 5 & 2 & 20 \\ 6 & 0 & 15 \end{bmatrix} \tag{4}$$

d) $D^2 - DE$

$$D^2 = \begin{bmatrix} -6 & 4 & 0 \\ 1 & 1 & 4 \\ -6 & 0 & 6 \end{bmatrix}^2 = \begin{bmatrix} -6 & 4 & 0 \\ 1 & 1 & 4 \\ -6 & 0 & 6 \end{bmatrix} \cdot \begin{bmatrix} -6 & 4 & 0 \\ 1 & 1 & 4 \\ -6 & 0 & 6 \end{bmatrix} = \begin{bmatrix} 40 & -20 & 16 \\ -29 & 5 & 28 \\ 0 & -24 & 36 \end{bmatrix}$$

$$DE = \begin{bmatrix} -6 & 4 & 0 \\ 1 & 1 & 4 \\ -6 & 0 & 6 \end{bmatrix} \cdot \begin{bmatrix} 6 & 9 & -9 \\ -1 & 0 & -4 \\ -6 & 0 & -1 \end{bmatrix} = \begin{bmatrix} -40 & -54 & -38 \\ 19 & 9 & -17 \\ -72 & -54 & -48 \end{bmatrix}$$

Assim sendo:

$$D^2 - DE = \begin{bmatrix} 80 & 34 & -22 \\ -10 & -4 & 45 \\ 72 & 30 & -12 \end{bmatrix} \tag{5}$$

d) $(DC)^T$

$$DC = \begin{bmatrix} -6 & 4 & 0 \\ 1 & 1 & 4 \\ -6 & 0 & 6 \end{bmatrix} \cdot \begin{bmatrix} -6 & 9 & -7 \\ 7 & -3 & -2 \end{bmatrix}$$

<u>Não</u> é possível efetuar essa operação, já que uma matrix é de dimensão 3×3 $(D_{i,j})$ e outra, 2×3 $(C_{i,j})$ e essa multiplicação não é válida porque em D, $j = 3$ e em C, $i = 2$. Para ser compatível esses valores devem ser iguais.

Questão 03) Mostre que se A e B são matrizes que comutam com a matriz $M = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$.

Temos a matriz $A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$ e a matriz $B = \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix}$. Assim sendo:

$$AM = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \cdot \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} = \begin{bmatrix} -a_{12} & a_{11} \\ -a_{22} & a_{21} \end{bmatrix}$$

$$MA = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \cdot \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} = \begin{bmatrix} a_{21} & a_{22} \\ -a_{11} & -a_{12} \end{bmatrix}$$

A partir disso, podemos encontrar:

$$\begin{cases} -a_{12} = a_{21} \\ a_{11} = a_{22} \\ -a_{22} = -a_{11} \\ a_{21} = -a_{12} \end{cases}$$

Pode-se simplificar para a condição de que $a_{11} = a_{22} = r$, $a_{12} = s$ e $a_{21} = -s$. Similarmente, para a matriz B, temos $b_{11} = b_{22} = x$, $b_{12} = y$ e $b_{21} = -y$:

$$A = \begin{bmatrix} r & s \\ -s & r \end{bmatrix}, \quad B = \begin{bmatrix} x & y \\ -y & x \end{bmatrix}$$

multiplicando-as, encontramos:

$$AB = \begin{bmatrix} rx - sy & ry + sx \\ -sx - ry & -sy + rx \end{bmatrix}, \quad BA = \begin{bmatrix} rx - sy & sx + ry \\ -ry - sx & -sy + rx \end{bmatrix} \tag{6}$$

A partir da Equação 6 é possivel chegar a conclusão de que $AB = BA = M$.

Questão 05) Exercícios com python.

a) Responda a questão 2 e a questão 4(b) usando Python. Especifique os comandos utilizados em cada uma e as respostas (ou os erros) obtidas[1].

2-a) $AB - BA$

```
import pandas as pd

# Cria as matrizes
A = pd.DataFrame([[2, 0], [6, 7]])
B = pd.DataFrame([[0, 4], [2, -8]])

# Calcula AB-BA
M=A.dot(B) - B.dot(A)
```

[1] Utilizou-se a *library* Pandas para cálculo de todas as operações

```
# Imprime valor
print(M)

#      0    1
# 0  -24  -20
# 1   58   24
```

2-b) $2C - D$. Efetua-se a subtração apenas das linhas que são possíveis retornando NaN (not a number) nas que não são possíveis.

```
import pandas as pd

C = pd.DataFrame([[-6, 9, -7], [7, -3, -2]])
D = pd.DataFrame([[-6, 4, 0], [1, 1, 4], [-6, 0, 6]])

# Calcula 2C-D (calculo pode ser feito diretamente)
M = 2*C-D

# Imprime valor
print(M)

#        0      1      2
# 0   -6.0   14.0  -14.0
# 1   13.0   -7.0   -8.0
# 2    NaN    NaN    NaN
```

2-c) $(2D^T - 3E^T)^T$.

```
import pandas as pd

# Cria as matrizes
D = pd.DataFrame([[-6, 4, 0], [1, 1, 4], [-6, 0, 6]])
E = pd.DataFrame([[6, 9, -9], [-1, 0, -4], [-6, 0, -1]])

# Calcula (2D^T-3E^T)
M = 2*D.T-3*E.T
# Calcula a transposta de M
MT = M.T
# Imprime valor
print(MT)

#      0    1    2
# 0  -30  -19   27
# 1    5    2   20
```

```
# 2   6   0  15
```

2-d) $D^2 - DE$.

```
import pandas as pd

# Cria as matrizes
D = pd.DataFrame([[-6, 4, 0], [1, 1, 4], [-6, 0, 6]])
E = pd.DataFrame([[6, 9, -9], [-1, 0, -4], [-6, 0, -1]])

# Calcula D^2-DE
M = D.dot(D)-D.dot(E)
# Imprime valor
print(M)

#      0   1   2
# 0   80  34 -22
# 1  -10  -4  45
# 2   72  30 -12
```

2-d) $(DC)^T$. Erro quando calcula a multiplicação entre D e C

```
import pandas as pd

# Cria as matrizes
A = pd.DataFrame([[2, 0], [6, 7]])
B = pd.DataFrame([[0, 4], [2, -8]])
C = pd.DataFrame([[-6, 9, -7], [7, -3, -2]])
D = pd.DataFrame([[-6, 4, 0], [1, 1, 4], [-6, 0, 6]])
E = pd.DataFrame([[6, 9, -9], [-1, 0, -4], [-6, 0, -1]])

# Calcula DC
M = D.dot(C)
# Calcula M^T
MT = M.T
# Imprime valor
print(MT)

# Error in M = D.dot(C)
# raise ValueError("matrices are not aligned")
# ValueError: matrices are not aligned
```

b) Realize o seguinte experimento para tentar ter uma ideia de quão comum é encontrar matrizes cujo produto comuta. Em Python, digite as seguintes linhas de comando:

```python
import numpy as np
c=0

for i in range(10000):
    A=np.random.randint(-5,5,size=(3,3))
    B=np.random.randint(-5,5,size=(3,3))
    C=A@B==B@A
    if C.all():
        c=c+1
print(c)
```

Na primeira linha é o *import* da *library*, na segunda atribuição de valor a uma variável inteira, na terceira o ínicio do loop até 10000, na quarta e quinta duas matrizes randômicas de tamanho 3×3 para as variáveis A e B, na sexta a comparação de igualdade entre os produtos A@B e B@A (verificando a comutação), na sétima a condição de que se ambos forem iguais a variável C é incrementada em 1, na oitava linha o valor de c é impresso.

A variável c poderá ter valores entre 0 e 10000, pois o loop é executado 10000 vezes, dependendo apenas se o produto A@B é igual a B@A ou não. As matrizes A e B são geradas randômicamente e podem ter valores entre -5 e 5 e a comutação pode não ocorrer em todos os casos.

c) Realize o seguinte experimento para tentar ter uma ideia de quão comum é encontrar matrizes invertíveis. Em Python, digite as seguintes linhas de comando:

```python
import numpy as np
c=0
for i in range(10000):
    A=np.random.randint(-5,5,size=(4,4))
    if np.linalg.det(A) != 0:
        c+=1

print(c)
```

Na primeira linha é o *import* da *library*, na segunda atribuição de valor a uma variável inteira, na terceira o ínicio do loop até 10000, na quarta e quinta uma matriz randômica de tamanho 4×4 para a variável A, na sexta a condição de que se o determinante for diferente de 0 a variável c é incrementada em 1, na sétima linha o valor de c é impresso.

A variável c poderá ter valores entre 0 e 10000, pois o loop é executado 10000 vezes, dependendo apenas se o determinante for diferente de 0 ou não. As matrizes A são geradas randômicamente e podem ter valores entre -5 e 5.