UNIVERSIDADE FEDERAL DE MINAS GERAIS
INSTITUTO DE CIÊNCIAS EXATAS
DEPARTAMENTO CIÊNCIA DA COMPUTAÇÃO

BRENO DE CASTRO PIMENTA
RA: 2017114809

Trabalho: Lista 02
Disciplina: ALC
Turma: TZ

Belo Horizonte
2019

1) Já que a matriz possui posto = 1, suas colunas são múltiplas uma das outras, Logo:

$$\underset{\times 2}{\begin{bmatrix} 3 & ? & ? \\ ? & 4 & ? \\ ? & 12 & 42 \\ ? & ? & 28 \\ 1 & 2 & ? \end{bmatrix}} \Rightarrow \underset{\times 7/2}{\begin{bmatrix} 3 & 6 & ? \\ 2 & 4 & ? \\ 6 & 12 & 42 \\ ? & ? & 28 \\ 1 & 2 & ? \end{bmatrix}} \Rightarrow \begin{bmatrix} 3 & 6 & 21 \\ 2 & 4 & 14 \\ 6 & 12 & 42 \\ 4 & 8 & 28 \\ 1 & 2 & 7 \end{bmatrix}$$

2) a) $$B = \begin{bmatrix} 1,2 & 1 & -1 & -1,4 & \cdots \\ -0,8 & -1 & 1 & 0,6 & \cdots \\ 0,2 & ? & ? & 1,6 & \cdots \\ -0,8 & ? & ? & -1,4 & \cdots \\ 0,2 & 1 & 1 & 0,6 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

b) $$\begin{bmatrix} 1,25 & 1,05 & -0,95 & -1,35 & \cdots \\ -0,75 & -0,95 & 1,05 & 0,65 & \cdots \\ -0,25 & ? & ? & 1,15 & \cdots \\ 0,25 & ? & ? & -0,35 & \cdots \\ -0,5 & 0,3 & 0,3 & -0,1 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

c)

$$\begin{array}{c|cc} & \text{Filme 1} & \text{Filme 2} \\ \hline \text{Usu2} & x & y \\ \text{Usu3} & w & z \end{array} = \begin{array}{c} 0 \\ 1 \\ \text{Us2} \\ \text{Us3} \\ \vdots \end{array} \begin{bmatrix} & \\ & \\ 0{,}5 & -0{,}6 \\ -0{,}38 & 0{,}19 \\ \vdots & \end{bmatrix}_{\text{Usuário} \times K} \begin{bmatrix} 3{,}63 & \\ & 1{,}05 \end{bmatrix}_{K \times K} \begin{array}{cccc} 0 & 1\ \text{Filme} & 2\ \text{Filme} & 3 \quad 4 \\ \end{array} \begin{bmatrix} -0{,}61 & 0{,}48 & \\ 0{,}59 & 0{,}28 & \end{bmatrix}_{K \times \text{Filme}}$$

$$\therefore \underbrace{\begin{bmatrix} 0{,}5 & -0{,}6 \\ -0{,}38 & 0{,}19 \end{bmatrix}}_{U_{[\,]}} \times \underbrace{\begin{bmatrix} 3{,}63 & 0 \\ 0 & 1{,}05 \end{bmatrix}}_{\Sigma} \times \underbrace{\begin{bmatrix} -0{,}61 & 0{,}48 \\ 0{,}59 & 0{,}28 \end{bmatrix}}_{V^T_{[\,]}} = \begin{bmatrix} x & y \\ w & z \end{bmatrix}$$

$$\underbrace{\begin{bmatrix} 1{,}815 & -0{,}63 \\ -1{,}3794 & 0{,}1995 \end{bmatrix}}_{(U_{[\,]} \cdot \Sigma)} \cdot \underbrace{\begin{bmatrix} -0{,}61 & 0{,}48 \\ 0{,}59 & 0{,}28 \end{bmatrix}}_{V^T_{[\,]}} = \begin{bmatrix} x & y \\ w & z \end{bmatrix}$$

$$\begin{bmatrix} -1{,}47702 & 0{,}69336 \\ 0{,}959139 & -0{,}604812 \end{bmatrix} = \begin{bmatrix} x & y \\ w & z \end{bmatrix}$$

• Aproximando:

$$+\mu_i \rightarrow \begin{bmatrix} -1{,}02702 & 1{,}14336 \\ -0{,}090861 & -1{,}654812 \end{bmatrix}$$

$$+m_j \rightarrow \begin{bmatrix} 1{,}97298 & 5{,}14336 \\ 2{,}909139 & 2{,}345188 \end{bmatrix}$$

$$\text{Notas} \Rightarrow \begin{array}{c} \\ \text{Us. 2} \\ \text{Us. 3} \end{array} \begin{array}{cc} \text{Filme 1} & \text{Filme 2} \\ \end{array} \begin{bmatrix} 2 & 5 \\ 3 & 2 \end{bmatrix}$$

3) 3.1 (F). Primeiro que a fatoração de matrizes não-negativas (NMF) não é possível alcançar valores exatos em tempo hábil.
E segundo que esse método parte do princípio da inexistência de números negativos para a fatoração, caso haja se torna uma Semi-NMF.

3.2 (V)

$$A = \begin{bmatrix} & \end{bmatrix}_{m \times n} \rightarrow \begin{cases} A_K = \begin{bmatrix} \end{bmatrix}_{m \times K} \cdot \begin{bmatrix} \end{bmatrix}_{K \times K} \cdot \begin{bmatrix} \end{bmatrix}_{K \times n} \\ N_K = \begin{bmatrix} \end{bmatrix}_{m \times K} \times \begin{bmatrix} \end{bmatrix}_{K \times n} \end{cases}$$

O teorema do SVD pressupõe que sua decomposição que gera A_K, é a que possui menor erro através de uma aproximação de A através da Norma de Frobenius.

4) $\begin{pmatrix} 3 & 0 & 2 \\ 9 & 1 & 7 \\ 1 & 0 & 1 \end{pmatrix}$

a) Norma-1: $(13, 1, 10) \therefore \|A\|_1 = 13$

b) Norma-infinita: $(5, 17, 2) \therefore \|A\|_\infty = 17$

c) Norma-2:

```
In [1]: import numpy as np

In [2]: A = np.array([[3,0,2],
                      [9,1,7],
                      [1,0,1]])
        np.linalg.norm(A,2)

Out[2]: 12.074814532664146
```

d) Norma Frobenius: $\sqrt{|3|^2 + |2|^2 + |9|^2 + |1|^2 + |7|^2 + |1|^2 + |1|^2}$

$\therefore \|A\|_F = \sqrt{9 + 4 + 81 + 1 + 49 + 1 + 1}$

$\|A\|_F = \sqrt{146} \cong 12{,}0830459736$

5) a) Para representar 1 pixel: $2^8 = 256 \therefore$ 1 pixel = 1 byte.

$$[U]_{1024 \times K} \; [\Sigma]_{K \times K} \; [V^T]_{K \times 768} \Rightarrow \begin{cases} U = 1024 \times K \text{ bytes} \\ \Sigma \Rightarrow \text{pode ser armazenado apenas com K valores: K bytes.} \\ V^T \Rightarrow 768 \times K \text{ bytes.} \end{cases}$$

$$\left.\begin{aligned} TOTAL &= 1024K + K + 768K = \\ TOTAL &= K \cdot 1793 \text{ bytes} \end{aligned}\right\}$$

↳ 1 imagem armazenada usando SVD com posto K para compressão

$1024 \times 768 = 786432$ bytes → caso se armazene direto

$786432 > K \cdot 1793$

$\underbrace{452{,}23 > K}$

↳ Para valer a pena compressão, K deve ser igual a no máximo 452.

b) $[\quad]_{786432 \times z}$ → sendo z o número de imagens.

$$[U]_{786432 \times K} \; [\Sigma]_{K \times K} \; [V^T]_{K \times z} \Rightarrow (786432) \cdot K + K + (z \cdot K)$$

→ 10 imagens: $K \cdot 786443$ bytes

→ 1000 imagens: $K \cdot 787433$ bytes.