Gabarito da Lista de Exercícios 2 de Analise Numérica

Prof.: Fabrício Murai

Informações importantes:

- Data de entrega: até 23:59 do dia 05/04/2018.
- Questões podem ser discutidas entre até três alunos. Nomes dos colegas precisam ser listados. Contudo, a escrita das soluções e submissão deve ser feita individualmente.
- Submissão deve ser feita em formato PDF através do Moodle, mesmo que tenham sido resolvidas a mão e escaneadas.
- Todas as soluções devem ser justificadas.
- Se puder, peço por favor que marque o tempo gasto para resolver a lista, para que o tamanho da lista de exercícios seja ajustado em semestres futuros.

1. Considere o sistema linear $Ax = b$, onde a matriz $A = \begin{bmatrix} 9 & 18 \\ 18 & 52 \end{bmatrix}$ é simétrica e definida positiva. Sendo $b = \begin{bmatrix} 3 \\ 4 \end{bmatrix}$, encontre a solução usando a Decomposição de Cholesky.

   A matriz L pode ser obtida utilizando as fórmulas:

   $l_{jj} = \sqrt{a_{jj} - \sum_{k=1}^{j-1} l_{jk}^2},\ j = 1, 2, \ldots, n.$

   $l_{ij} = \frac{a_{ij} - \sum_{k=1}^{j-1} l_{ik} l_{jk}}{l_{jj}},\ j = 1, 2, \ldots, n$ e $i = j+1, j+2, \ldots, n.$

   Temos então

   Coluna 1: $l_{11} = \sqrt{a_{11}} = \sqrt{9} = 3,\ l_{21} = \frac{a_{21}}{l_{11}} = \frac{18}{3} = 6$

   Coluna 2: $l_{22} = \sqrt{a_{22} - l21^2} = \sqrt{52 - 6^2} = \sqrt{16} = 4$

   Esses resultados podem ser observados no dispositivo prático abaixo

A			L		
i\j	1	2	i\j	1	2
1	9	18	1		
2	18	52	2		

$\Downarrow$

A			L		
i\j	1	2	i\j	1	2
1	9	18	1	3	
2	18	52	2	6	

$\Downarrow$

A			L		
i\j	1	2	i\j	1	2
1	9	18	1	3	
2	18	52	2	6	4

Resolvendo o sistema $Ly = b$ pelas substituições sucessivas
$$\begin{bmatrix} 3 & 0 \\ 6 & 4 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 3 \\ 4 \end{bmatrix} \rightarrow y = \begin{bmatrix} 1 \\ -\frac{1}{2} \end{bmatrix}$$

Resolvendo o sistema $L^T x = y$ pelas substituições retroativas
$$\begin{bmatrix} 3 & 6 \\ 0 & 4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 1 \\ -\frac{1}{2} \end{bmatrix} \rightarrow x = \begin{bmatrix} \frac{7}{12} \\ -\frac{1}{8} \end{bmatrix}$$

2. Refinamento de solução baseado na matriz A anterior.

   (a) Para $b = \begin{bmatrix} 3 \\ 4 \end{bmatrix}$, calcule o erro residual ($r = b - Ax$) da solução aproximada dada por $x^{(0)} = \begin{bmatrix} 0,5 \\ -0,1 \end{bmatrix}$.
   $$r^{(0)} = \begin{pmatrix} 3 \\ 4 \end{pmatrix} - \begin{pmatrix} 9 & 18 \\ 18 & 52 \end{pmatrix} \begin{pmatrix} 0.5 \\ -0.1 \end{pmatrix} = \begin{pmatrix} 0.3 \\ 0.2 \end{pmatrix}.$$

   (b) Use r para executar um refinamento da solução $x^{(0)}$.
   Para encontrar a nova aproximação $x^{(1)} = x^{(0)} + c^{(0)}$, basta resolver $Ac^{(0)} = r^{(0)}$. Usando a fatoração Cholesky obtida na questão anterior, temos
   $$Ly = r_0 \rightarrow \begin{bmatrix} 3 & 0 \\ 6 & 4 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} 0.3 \\ 0.2 \end{bmatrix}$$
   Resolvendo pelas substituições sucessivas, $y = \begin{bmatrix} 0.1 \\ -0.1 \end{bmatrix}$. Em seguida, resolvemos
   $$L^\top c^{(0)} = y \rightarrow \begin{bmatrix} 3 & 6 \\ 0 & 4 \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} 0.1 \\ -0.1 \end{bmatrix},$$
   através das substituições retroativas, obtendo $c^{(0)} = \begin{bmatrix} 0,0833 \\ -0,025 \end{bmatrix}$.

   Logo, $x^{(1)} = x^{(0)} + c^{(0)} = \begin{bmatrix} 0,5833 \\ -0,125 \end{bmatrix}$.

3. Assinale **V** para verdadeiro ou **F** para falso **e justifique**:

   ( ) A decomposição de Cholesky pode ser aplicada a qualquer matriz $A^\top A$ tal que as entradas de A são números reais.

   Verdadeiro. Toda matriz $A^\top A$ é semidefinida positiva e, portanto, simétrica. Não é necessário provar, mas a demonstração pode ser obtida pela forma quadrática $v^\top A^\top Av = (Av)^\top Av = (Av)\dot{(}Av) \geq 0$.

   ( ) A fatoração LU de uma matriz quadrada A de posto n pode resultar em uma matriz U com linhas nulas.

   Falso. A obtenção de uma linha nula em U só pode acontecer se esta for uma combinação linear de outras linhas. No entanto, como o posto$(A) = n$, sabemos que A possui n linhas linearmente independentes.

   ( ) A fatoração LU é aproximadamente duas vezes mais demorada que a fatoração Cholesky, mas ambas requerem o mesmo espaço em memória.

   Falso. A primeira parte da sentença é verdadeira, mas o custo de memória da fatoração Cholesky é aproximadamente metade do custo para armazenar as matrizes LU.

4. Dada a matriz $A = \begin{bmatrix} 3 & 0 & 2 \\ 9 & 1 & 7 \\ 1 & 0 & 1 \end{bmatrix}$ e sua inversa $A^{-1} = \begin{bmatrix} 1 & 0 & -2 \\ -2 & 1 & -3 \\ -1 & 0 & 3 \end{bmatrix}$, calcule o número de condicionamento segundos à

(a) Norma-1. É bem condicionado?

$||A||_1 = \underset{1\leq j\leq 3}{max} \sum_{i=1}^{3} |a_{ij}| = max(13, 1, 10) = 13$
$||A^{-1}||_1 = \underset{1\leq j\leq 3}{max} \sum_{i=1}^{3} |a_{ij}| = max(4, 1, 8) = 8$
$\kappa_1(A) = ||A||_1||A^{-1}||_1 = 13 \times 8 = 104$
$\kappa_1(A) \gg 1$ portanto é mal condicionado.

(b) Norma-infinito. É bem condicionado?

$||A||_\infty = \underset{1\leq i\leq 3}{max} \sum_{j=1}^{3} |a_{ij}| = max(5, 17, 2) = 17$
$||A^{-1}||_\infty = \underset{1\leq i\leq 3}{max} \sum_{j=1}^{3} |a_{ij}| = max(3, 6, 4) = 6$
$\kappa_\infty(A) = ||A||_1||A^{-1}||_1 = 17 \times 6 = 102$
$\kappa_\infty(A) \gg 1$ portanto é mal condicionado.
OBS: é válido lembrar que o número de condicionamento varia com a norma utilizada. Isso pôde ser observado nas duas letras anteriores, obtivemos 2 números de condicionamento diferentes para a mesma matriz A.

(c) Considere um erro na medição do vetor $b = \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix}$, resultando no vetor $b' = \begin{bmatrix} 1,1 \\ 1,9 \\ 3,0 \end{bmatrix}$. Usando o número de condição calculado a partir da norma-infinito, determine o limite superior do erro da aproximação encontrada ao se resolver $Ax = b'$.

Temos que $\delta b = b' - b = \begin{bmatrix} 1,1 \\ 1,9 \\ 3 \end{bmatrix} - \begin{bmatrix} 1 \\ 2 \\ 3 \end{bmatrix} = \begin{bmatrix} 0,1 \\ -0,1 \\ 0 \end{bmatrix}$
A fórmula utilizada para calcular o limite superior do erro é dada por:
$\frac{||\delta x||}{||x||} \leq \kappa(A)\frac{||\delta b||}{||b||}|$, e usaremos $\kappa_\infty(A)$ calculado no item anterior.
$\kappa_\infty = 102$, $||b||_\infty = 3$, $||\delta b||_\infty = 0,1$
Substituindo os valores, $\frac{||\delta x||}{||x||} \leq 102 \times \frac{0,1}{3} \Rightarrow \frac{||\delta x||}{||x||} \leq 3.4$
Uma explição mais detalhada desse limite superior do erro pode ser encontrada no tópico 2.9.3 - Sensibilidade da solução- (páginas 115 e 116) do livro Algorimos Númericos do professor Frederico. É aconselhável estudar antes o tópico 2.1.4 -Normas- (páginas 39, 40, 41 e 42) para entender as diferentes normas e quando uma norma é dita consistente.

5. Assinale **V** para verdadeiro ou **F** para falso **e justifique**:

( ) Toda matriz $A \in \mathbb{R}^{m\times n}$ possui um SVD.

Verdadeiro. Os valores e vetores singulares vem da decomposição espectral de $A^\top A$, que sempre existe.

( ) Na decomposicao SVD $A = U\Sigma V^\top$, as colunas de U sao ortogonais às linhas de V.

Falso. As colunas de U são ortogonais entre si e as colunhas de V são ortogonais entre si.

( ) Se A é singular, pelo menos um dos autovalores de A é nulo.

Verdadeiro. Como $\det(A) = 0$ é igual ao produto dos seus autovalores, pelo menos um deles deve ser nulo.

6. Seja a matriz $C = \begin{bmatrix} 1 & -1 \\ 0 & 1 \\ 1 & 0 \\ -1 & 1 \end{bmatrix}$. Mostre que

$$U = \begin{bmatrix} -0.632 & 0.000 \\ 0.316 & -0.707 \\ -0.316 & -0.707 \\ 0.632 & 0.000 \end{bmatrix} \quad \Sigma = \begin{bmatrix} 2.236 & 0.000 \\ 0.000 & 1.000 \end{bmatrix} \quad V^\top = \begin{bmatrix} -0.707 & 0.707 \\ -0.707 & -0.707 \end{bmatrix}$$

é o SVD de C. Para isso, você pode usar:

- a decomposição espectral $C^\top C = V\Lambda V^\top$ e encontrar U usando $Cv_i = \sigma_i u_i \Rightarrow u_i = \frac{Cv_i}{\sigma_i}$, **OU**
- o fato de que o SVD é único quando todos os valores singulares de uma matriz são não-degenerados (não aparecem com multiplicidade maior que 1) e não-nulos.

Solução 1. As linhas da matriz $V^\top$ (vetores singulares direitos de C) são os autovetores de $C^\top C$ associados a autovalores não-nulos. Decompondo $C^\top C = V\Lambda V^\top$, temos

$$V = \begin{bmatrix} 0.707 & 0.707 \\ -0.707 & 0.707 \end{bmatrix} \quad \sigma_1 = \sqrt{5} = 2.236 \quad \sigma_2 = \sqrt{1} = 1.000.$$

Em seguida, calculamos $u_i = \frac{Cv_i}{\sigma_i}$, obtendo

$$U = \begin{bmatrix} 0.632 & 0.000 \\ -0.316 & 0.707 \\ 0.316 & 0.707 \\ -0.632 & 0.000 \end{bmatrix}.$$

Embora os sinais apareçam trocados, este SVD é equivalente.

```
C = np.array([[1,-1],[0,1],[1,0],[-1,1]])
Lambda, V = np.linalg.eig(C.T @ C)
print(Lambda)
print(V)
Sigma = np.sqrt(Lambda)
U = np.zeros((4,2))
for i in range(2):
    U[:,i] = (C @ V[:,i])/Sigma[i]
```

Solução 2. Como todos os valores singulares são não-nulos e tem multiplicidade 1, basta mostrar que U e V são matrizes ortornomais. Para isso, basta fazer $UU^\top = \mathbb{I}_4$ e $VV^\top = \mathbb{I}_2$. Logo, esta decomposição tem de ser o SVD.

7. Suponha que a matriz C acima denota a opinião de quatro usuários sobre dois sites na Internet (-1: negativa, 0: neutra, 1: positiva). Vamos identificar os pares de usuários mais semelhantes. Uma forma de medir a semelhança entre dois usuários representados pelos vetores u e v é calcular o cosseno do ângulo θ formado entre eles:
$$\cos(\theta) = \frac{u \cdot v}{\|u\|\|v\|}.$$
Identifique para cada vetor, o outro vetor com que ele mais se assemelha. Refaça as contas utilizando a matriz U. O que você observou?

Calculando a similaridade de cosseno para a matriz C, obtemos a matriz

$$\begin{pmatrix} 1.000 & -0.707 & 0.707 & -1.000 \\ -0.707 & 1.000 & 0.000 & 0.707 \\ 0.707 & 0.000 & 1.000 & -0.707 \\ -1.000 & 0.707 & -0.707 & 1.000 \end{pmatrix},$$

logo o vetor que mais se assemelha ao 1o. é o 3o., ao 2o. é o 4o., ao 3o. é o 1o., e ao 4o. é o 2o.

Calculando a similaridade de cosseno para a matriz C, obtemos a matriz

$$\begin{pmatrix} 1.000 & -0.408 & 0.408 & -1.000 \\ -0.408 & 1.000 & 0.667 & 0.408 \\ 0.408 & 0.667 & 1.000 & -0.408 \\ -1.000 & 0.408 & -0.408 & 1.000 \end{pmatrix},$$

logo o vetor que mais se assemelha ao 1o. é o 3o., ao 2o. é o 3o., ao 3o. é o 2o., e ao 4o. é o 2o.

Observamos que o ranking das similaridades em C não é preservado na matriz U.

8. Suponha que queremos adicionar informação sobre um novo usuário que tem opinião positiva sobre os dois sites $w = (1, 1)$ à matriz U, sem recalcular o SVD. Como seria a representação deste usuário?

   Para isso teríamos que adicionar uma linha $x^\top$ à U de forma que $x^\top \Sigma V^\top = (1, 1)$. Transpondo ambos os lados temos:

   $$\begin{pmatrix} 0.707 & 0.707 \\ -0.707 & 0.707 \end{pmatrix} \begin{pmatrix} 2.236 & 0.000 \\ 0.000 & 1.000 \end{pmatrix} x = \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

   Resolvendo este sistema encontramos $x^\top = [0.000, -1.414]$.

9. Seja A uma matriz $m \times n$. O SVD reduzido de A retorna $U_{m\times k}$, $\Sigma_{k\times k}$ e $V^\top_{k\times n}$. Já o SVD truncado de posto $r < \min(m, n)$ retorna apenas as r primeiras colunas de U, os r maiores valores singulares de Σ e as r primeiras linhas de $V^\top$.

   Uma propriedade importante do SVD truncado é que ele retorna a melhor aproximação A_r para uma matriz A dentre todas as matrizes de posto r, onde a qualidade da aproximação é medida por $\|A - A_r\|_F$, sendo $\|B\|_F = \sqrt{\sum_i \sum_j B_{i,j}^2}$ a norma de Frobenius de uma matriz B.

   Nesta questão, vamos ver como a qualidade da aproximação aumenta com r. Primeiramente, vamos gerar uma matriz aleatória $A_{10\times 8}$. Em seguida, vamos encontrar a decomposição SVD reduzida. Depois disso, vamos variar o número r de valores singulares considerados para encontrar aproximações A_r para, finalmente, calcular $\|A - A_r\|_F$. Você pode criar uma tabela com os erros ou um gráfico, se preferir.

   Para facilitar a resolução deste problema, o código com algumas lacunas é fornecido a seguir.

```
import numpy as np
m = 10; n = 8

A = ???             # criar uma matriz aleatoria 10 x 8
U,Sigma,Vt = np.linalg.svd(A)

erro = np.zeros(n)
A_r = np.zeros((m,n))
for r in range(min(m,n)):
    # calcular a aproximacao A_r de posto r para A
    A_r += np.outer(??? * Sigma[r],???)
    erro[r] = np.linalg.norm(???)

print(erro)
```

```
import numpy as np
m = 10; n = 8

A = np.random.rand(m,n)             # criar uma matriz aleatoria 10 x 8
U,Sigma,Vt = np.linalg.svd(A)

erro = np.zeros(n)
```

```
A_r = np.zeros((m,n))
for r in range(min(m,n)):
    # calcular a aproximacao A_r de posto r para A
    A_r += np.outer(U[:,r] * Sigma[r],Vt[r,:])
    erro[r] = np.linalg.norm(A-A_r)

print(erro)
```

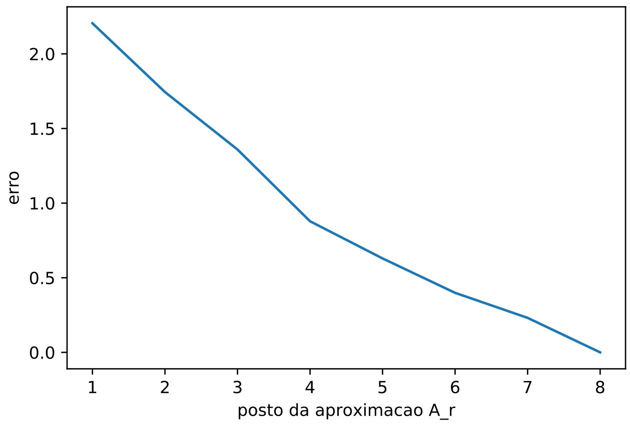

Figura 1: Solução da Questão 9: variação do erro conforme o posto da aproximação SVD.

10. Use a decomposição espectral de $A = \begin{pmatrix} 2 & -12 \\ 1 & -5 \end{pmatrix}$ em $A = Q\Lambda Q^{-1}$, onde $Q = \begin{pmatrix} 4 & 3 \\ 1 & 1 \end{pmatrix}$, $\Lambda = \begin{pmatrix} -1 & 0 \\ 0 & -2 \end{pmatrix}$ e $Q^{-1} = \begin{pmatrix} 1 & -3 \\ -1 & 4 \end{pmatrix}$, para calcular:

    (a) a inversa da matriz A, e
    Temos que $A^{-1} = (Q\Lambda Q^{-1})^{-1} = Q\Lambda^{-1}Q^{-1}$. Como Λ é uma matriz diagonal, sua inversa é obtida invertendo-se os elementos da diagonal. Logo $A^{-1} = \begin{pmatrix} -2.5 & 6.0 \\ -0.5 & 1.0 \end{pmatrix}$

    (b) a potência A^{10}.
    Temos que $A^{10} = Q\Lambda^{10}Q^{-1} = \begin{pmatrix} -3068 & 12276 \\ -1023 & 4093 \end{pmatrix}$.

    (c) o produto $A^{10}w$, por um vetor w de sua escolha e normalize o resultado para obter um vetor unitário. Qual a relação entre o resultado obtido e os autovetores de A?

Observação: Se você quiser calcular a potência A^{10} para conferir sua solução do item b, você deve multiplicar a matriz 10 vezes ou usar `np.linalg.matrix_power(A,10)`. É errado fazer `A**10`, pois esta operação na verdade retorna uma matriz onde cada elemento de A está elevado a 10.

Vamos escolher $w = [1, 1]^\top$, mas qualquer escolha onde uma coordenada não seja muito maior que a outra deve funcionar também. Nesse caso, o resultado obtido é $A^{10}w = [92083070]^\top$. Note que isso é aproximadamente 3070 q_2, onde q_2 é o autovetor relativo ao maior autovalor em módulo (i.e., -2). A explicação teórica é a seguinte. Sejam c_1 e c_2 os coeficientes tais queasily $w = c_1q_1 + c_2q_2$. Logo, tempoes

$$
\begin{aligned}
A^k w &= A^k(c_1q_1 + c_2q_2) = c_1A^kq_1 + c_2A^kq_2 \\
&= c_1\lambda_1^k q_1 + c_2\lambda_2^k q_2.
\end{aligned}
$$

Como $|\lambda_2^k| = 2^{10} \gg 1^{10} = |\lambda_1^k|$, A^kw é aproximadamente um múltiplo de q_2.