Matemática Computacional

Exercícios de Sistemas de Equações Lineares

maria irene falcão

Mestrado em Matemática e Computação

Exercícios

Exercício 1. Considere as matrizes $T_n = (t_{ij}),\ K_n = (k_{ij}),\ H_n = (h_{ij}),\ S_n = (s_{ij})$ definidas por

$$t_{ij} = \begin{cases} 4, & \text{se } i = j, \\ 1, & \text{se } |i-j| = 1, \\ 0, & \text{restantes valores de } i \text{ e } j \end{cases} \qquad k_{ij} = \begin{cases} 2, & \text{se } i = j, \\ -1, & \text{se } |i-j| = 1, \\ 0, & \text{restantes valores de } i \text{ e } j \end{cases}$$

$$h_{ij} = \frac{1}{i+j-1} \quad \text{(matrizes de Hilbert)} \qquad s_{ij} = \begin{cases} 1, & \text{se } i = j, \\ 4, & \text{se } i = j+1, \\ -4, & \text{se } i = j-1, \\ 0, & \text{restantes valores de } i \text{ e } j \end{cases}$$

Defina, no Matlab, de uma forma simples, as matrizes T_5, K_5, H_5 e S_5.

Nota: Para a matriz de Hilbert, faça uso da função pré-definida **hilb**.

Exercício 2. Resolva, usando uma função adequada cada um dos sistema de equações.

$$\begin{cases} 5\,x_1 = -10 \\ x_1 + 3\,x_2 = 4 \\ 3\,x_1 + 4\,x_2 + 2\,x_3 = 2 \\ -x_1 + 3\,x_2 - 6x_3 - x_4 = 5 \end{cases} \qquad \begin{cases} 3\,x_1 - 2\,x_2 + x_3 - x_4 = 8 \\ 4\,x_2 - x_3 + 2\,x_4 = -3 \\ 2\,x_3 + 3\,x_4 = 11 \\ 5\,x_4 = 15 \end{cases}$$

Exercício 3.

a) Descreva de que forma se pode obter a matriz inversa de uma dada matriz A, coluna a coluna, resolvendo n sistemas da forma $Ax = b$.

b) Considere a matriz $A = (a_{ij})$ de ordem 10, definida por

$$a_{ij} = \begin{cases} 1, & \text{se } i = j, \\ -1, & \text{se } i < j, \\ 0, & \text{se } i > j. \end{cases}$$

(i) Determine a inversa de A, fazendo uso da função **subInversa**.

(ii) Use a função pré-definida **inv** para calcular a inversa de A e compare o resultado com o resultado da alínea anterior.

Exercício 4. Considere o problema da resolução do seguinte sistema de equações lineares, numa máquina com sistema de numeração $F(10, 4, -99, 99)$:

$$\begin{cases} 0.0003\, x_1 + 1.566\, x_2 = 1.569 \\ \\ 0.3454\, x_1 - 2.436\, x_2 = 1.018 \end{cases}$$

a) Resolva o sistema, usando o método de Gauss:

(i) sem escolha de *pivot*;

(ii) com escolha parcial de *pivot*.

b) Comente os resultados obtidos, comparando com a solução dada pelo MATLAB.

Exercício 5. Utilize a função **metGauss** para resolver os sistemas $Ax = b$, onde:

a) $A = \begin{pmatrix} 4 & 1 & 1 & 0 & 1 \\ -1 & -3 & 1 & 1 & 0 \\ 2 & 1 & 5 & -1 & -1 \\ -1 & -1 & -1 & 4 & 0 \\ 0 & 2 & -1 & 1 & 4 \end{pmatrix}, \qquad b = \begin{pmatrix} 6 \\ 6 \\ 6 \\ 6 \\ 6 \end{pmatrix};$

b) $A = T_5, \quad b = (1, 6, 12, 18, 19)^{\mathsf{T}}$; (cf. Exercício 1)

c) $A = S_5, \quad b = (-4, -7, -6, -5, 16)^{\mathsf{T}}$;

Exercício 6. Considere a matriz

$$A = \begin{pmatrix} 2 & 2 & -2 & 4 \\ 1 & 5 & 7 & -10 \\ 0 & 1 & 5 & 3 \\ -1 & 1 & 6 & -5 \end{pmatrix}.$$

a) Obtenha a decomposição LU de A.

b) Resolva o sistema $Ax = b$, com $b = (1, 1, 1, 1)^{\mathsf{T}}$, usando a decomposição obtida em *a)* .

Exercício 7. Use a função pré-definida **lu** do MATLAB para:

a) obter a decomposição LU da matriz PA, onde

$$A = \begin{pmatrix} 17 & 24 & 1 & 8 \\ 23 & 5 & 7 & 14 \\ 4 & 6 & 13 & 29 \\ 10 & 12 & 19 & 21 \end{pmatrix}$$

e P é uma matriz de permutação adequada (correspondente às escolhas de *pivot* usadas no algoritmo). Resolva, então, o sistema $Ax = b$, onde $b = (16, 3, 44, 42)^{\mathsf{T}}$.

b) resolver o sistema $S_5 x = b$, com $b = (-4, -7, -6, -5, 16)^{\mathsf{T}}$.

Exercício 8. Dada uma matriz não singular $A \in \mathbb{R}^{n\times n}$ e supondo conhecida uma decomposição $A = LU$, como resolveria, de forma eficiente, cada um dos problemas seguintes?

a) Resolver a equação matricial $AX = B$, com B uma dada matriz de $\mathbb{R}^{n\times m}$.

b) Determinar a solução do sistema $A^k x = b$, com k um determinado inteiro positivo.

c) Calcular $\alpha = c^\mathsf{T} A^{-1} b$, sendo $c \in \mathbb{R}^n$ dado.

Exercício 9. Use a função **norm** do MATLAB para obter os seguintes valores:

$$||H_{100}||_\infty, \quad ||H_{100}||_1, \quad ||H_{100}||_2 \quad ||T_{100}||_\infty, \quad ||T_{100}||_1, \quad ||T_{100}||_2.$$

Exercício 10. Considere as matrizes $A_\epsilon = \begin{pmatrix} 2 & -1 \\ -1 & 0.5+\epsilon \end{pmatrix}$.

a) Calcule, recorrendo a funções pré-definidas, o número de condição de A_ϵ para $\epsilon = 1, 0.1, 0.01$ e 0.001 (relativamente à norma $\|\cdot\|_\infty$).

b) Resolva os sistemas

$$\begin{cases} 2x - y = 1 \\ -x + 0.499y = 1 \end{cases} \qquad \text{e} \qquad \begin{cases} 2x - y = 1 \\ -x + 0.501y = 1 \end{cases}$$

e comente os resultados obtidos.

Exercício 11. Considere o problema da resolução de um sistema $Ax = b$, onde $A = \begin{pmatrix} 0.5 & 0.4 \\ 0.3 & 0.25 \end{pmatrix}$ e onde se conhece apenas uma aproximação $\tilde{b} = \begin{pmatrix} 0.200 & 1.000 \end{pmatrix}^\mathsf{T}$ para o vetor b dos termos independentes. Sabendo que os elementos de $\tilde{b}$ são aproximações para os correspondentes elementos de b com 3 casas decimais correctas, indique um majorante para o erro relativo no vetor solução

$$\frac{\|x - \tilde{x}\|_\infty}{\|x\|_\infty}.$$

Exercício 12. Considere o sistema $Ax = B$, onde $A = \begin{pmatrix} 0.1 & 0.2 & 0.3 \\ 0.4 & 0.5 & 0.6 \\ 0.7 & 0.8 & 0.9 \end{pmatrix}$ e $b = (0.1, 0.3, 0.5)^\mathsf{T}$.

a) Verifique que A é uma matriz singular e descreva o conjunto de soluções do sistema dado.

b) Se usássemos o método de eliminação Gaussiana com escolha parcial de *pivot* (com aritmética exata) para resolver o sistema, em que passo falharia o processo?

c) Tente resolver esse sistema usando a função **metGauss**. Como justifica o resultado?

d) Tente novamente resolver o sistema, usando a forma $x = A\backslash b$ usual do MATLAB. Comente os resultados.

Exercício 13. Considere o sistema linear $Ax = b$, onde

$$A = \begin{pmatrix} 1.002 & 1 \\ 1 & 0.998 \end{pmatrix} \quad \text{e} \quad b = \begin{pmatrix} 0.002 \\ 0.002 \end{pmatrix},$$

o qual admite como solução exata $x = (1, -1)^{\mathsf{T}}$. Seja $\tilde{x} = (0.29360067817338, -0.29218646673249)^{\mathsf{T}}$ uma "aproximação" para a solução do sistema. Calcule o vetor residual $r = A\tilde{x} - b$, compare $\|r\|_\infty$ com $\|x - \tilde{x}\|_\infty$ e comente.

Exercício 14. **Matrizes esparsas no MATLAB**

O MATLAB tem alguns comandos que permitem trabalhar com matrizes esparsas, armazenando apenas os elementos não nulos da matriz. O uso de matrizes esparsas para armazenar dados que contêm um grande número de elementos com valor zero pode economizar uma quantidade significativa de memória e acelerar o processamento desses dados.

1. Considere a matriz

$$A = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 4 \end{pmatrix}$$

   a) Defina a matriz A no MATLAB, de forma simples.

   b) Execute as seguintes instruções e analise cuidadosamente o resultado de cada uma delas.

```
>> S=sparse(A)
>> nnz(A)
>> full(S)
>> spy(S)
```

2. Execute as seguintes instruções e analise cuidadosamente o resultado de cada uma delas:

```
>> S = sparse([1 2 3], [3 1 2], [4 5 6])
>> full(S)
>> S = spdiags([ones(6,6)], [-3,2], 6, 6);
>> A=full(S)
>> S = spdiags([ones(6,6)], [-1:1], 6, 6);
>> A=full(S)
```

3. Use a função `spdiags` para construir a matriz T_{10}. Obtenha a decomposição LU de T_{10}.

Exercício 15. Mostre que o passo k do método de Jacobi para a resolução de um sistema $Ax = b$ pode ser descrito como

$$Dx^{(k+1)} = -(M+N)x^{(k)} + b,$$

onde D, M e N são as matrizes que contêm a diagonal, a parte triangular inferior e a parte triangular superior de A, respetivamente, e diga como poderá obter facilmente a matriz $M+N$ no MATLAB.

Exercício 16. Use a função **metJacobi** para resolver os seguintes sistemas, usando diversos valores dos parâmetros `tol` e `maxit`.

(i) $\begin{cases} 10x_1 - x_2 + 2x_3 = 6 \\ -x_1 + 11x_2 - x_3 + 3x_4 = 25 \\ 2x_1 - x_2 + 10x_3 - x_4 = -11 \\ 3x_2 - x_3 + 8x_4 = 15 \end{cases}$ (ii) $A = T_{10}, \quad b = (3, 3, \ldots, 3)^\mathsf{T}$

(iii) $\begin{cases} 2x + 3y = 1 \\ 7x - 2y = 1 \end{cases}$ (iv) $A = T_{10}, \quad b = (1, 2, \ldots 1, 2)^\mathsf{T}$.

Exercício 17. Mostre que o passo k do método de Gauss-Seidel para a resolução de um sistema $Ax = b$ pode ser descrito como

$$(D+M)x^{(k+1)} = -Nx^{(k)} + b, \qquad (1)$$

onde D, M e N são as matrizes usuais. Tendo em conta que $D+M$ é uma matriz triangular inferior, sugira uma forma eficiente de obter o vetor $x^{(k+1)}$ a partir da equação (1).

Exercício 18. Repita o Exercício 16, usando agora a função **metGSeidel**.

Exercício 19.

a) Seja

$$A = \begin{pmatrix} 1 & 2 & -2 \\ 1 & 1 & 1 \\ 2 & 2 & 1 \end{pmatrix}.$$

Mostre que $\rho(B_J) < 1$ e $\rho(B_{GS}) > 1$. Que pode concluir quanto à convergência dos métodos de Jacobi e de Gauss-Seidel aplicados à resolução de um sistema que tenha A como matriz?

b) Seja $b = (1, 3, 5)^\mathsf{T}$. Efetue 5 iterações do método de Jacobi para resolver o sistema $Ax = b$, tomando como aproximação inicial o vetor $(0, 0, 0)^\mathsf{T}$. Repita com o método de Gauss-Seidel e verifique se os resultados confirmam a conclusão da alínea anterior.

c) Seja

$$A = \begin{pmatrix} 2 & -1 & 1 \\ 2 & 2 & 2 \\ -1 & -1 & 2 \end{pmatrix}.$$

Mostre que $\rho(B_J) > 1$ e $\rho(B_{GS}) < 1$. Neste caso, como se comportariam os métodos de Jacobi e de Gauss-Seidel quanto à convergência? Teste a sua conclusão, resolvendo o sistema com $Ax = b$, com b escolhido por forma que o sistema tenha $x = (1, 1, 1)^{\mathsf{T}}$ como vetor solução.

Exercício 20. Faça uso das funções **raioJacobi** e **raioGSeidel** para justificar o comportamento dos métodos de Jacobi e Gauss-Seidel na resolução dos sistemas considerados no Exercício 16.

Exercício 21. Discuta, em função dos valores do parâmetro $\alpha \in \mathbb{R}$, o comportamento dos métodos de Jacobi e de Gauss-Seidel aplicados a um sistema de equações cuja matriz seja

$$A_\alpha = \begin{pmatrix} \alpha & 0 & 1 \\ 0 & \alpha & 0 \\ 1 & 0 & \alpha \end{pmatrix}, \quad \alpha \neq 0.$$

Trabalhos

Trabalho 1.

a) Modifique a função **metGauss** de forma a que esta permita resolver vários sistemas de equações lineares em simultâneo e forneça também o determinante da matriz do sistema, caso o utilizador o deseje. Mais precisamente, a sua função deve: passar a aceitar como argumentos, duas matrizes $A \in \mathbb{R}^{n\times n}$ e $B \in \mathbb{R}^{n\times m}$; resolver a equação matricial $AX = B$, armazenando a matriz solução em B; calcular o determinante de A, caso a função seja invocada com dois argumentos de saída, $[B, deter] = \mathbf{metGaussMod}(A, B)$.

b) Considere a matriz

$$A = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 2 & 2 & 5 & 3 \\ 4 & 6 & 8 & 0 \\ 3 & 3 & 9 & 8 \end{pmatrix}.$$

Calcule o determinante de A e a matriz A^{-1}, usando a função da alínea anterior.

Trabalho 2. Escreva uma função $x = \textbf{resolveSistema}(A, b)$ destinada a resolver uma sistema de equações $Ax = b$, escolhendo um método adequado. Mais precisamente, a sua função deve

- começar por testar se A é triangular inferior (superior), resolvendo, nesse caso, o sistema por substituição directa (inversa);
- testar se A é tridiagonal, efetuando nesse caso a decomposição LU especialmente desenhada para esse caso, e resolvendo o sistema por substituição directa e inversa;
- verificar se A é simétrica, tentando, nesse caso, obter a sua decomposição de Cholesky; se tal decomposição for obtida, usá-la para resolver o sistema por substituição directa e inversa;
- em qualquer outro caso, usar o método de Gauss com escolha parcial de pivot.

Em cada caso, deverá ser dada uma mensagem indicando qual foi o método utilizado. Teste a sua função para resolver um sistema de cada um dos casos considerados.

Trabalho 3. Considere uma matriz tridiagonal

$$A = \begin{pmatrix} d_1 & f_1 & & & \\ e_1 & d_2 & f_2 & & \\ & \ddots & \ddots & \ddots & \\ & & e_{n-2} & d_{n-1} & f_{n-1} \\ & & & e_{n-1} & d_n \end{pmatrix},$$

a qual supomos admitir uma decomposição $A = LU$, onde

$$L = \begin{pmatrix} 1 & & & & \\ \ell_1 & 1 & & & \\ & \ddots & \ddots & & \\ & & \ell_{n-2} & 1 & \\ & & & \ell_{n-1} & 1 \end{pmatrix}, \; U = \begin{pmatrix} u_1 & f_1 & & & \\ & u_2 & f_2 & & \\ & & \ddots & \ddots & \\ & & & u_{n-1} & f_{n-1} \\ & & & & u_n \end{pmatrix}.$$

a) Escreva um algoritmo para determinar a decomposição LU de A, ou seja, para determinar os vetores $\ell = (\ell_1, \ldots, \ell_{n-1})$ e $(u_1, \ldots, u_n)$, em função dos elementos da diagonal, subdiagonal e sobrediagonal de A.

b) Efetue uma contagem do número de operações (multiplicações/divisões e adições/subtrações) envolvidas no algoritmo anterior.

c) Escreva uma função $[\ell, u] = \textbf{luTrid}(d, e, f)$ para implementar o algoritmo referido. A função deve

- aceitar como argumentos:
  - um vetor $d = (d_1, \ldots, d_n)$ (diagonal de uma matriz tridiagonal);

  - um vetor $e = (e_1, \ldots, e_{n-1})$ (subdiagonal da mesma matriz);
  - um vetor $f = (f_1, \ldots, f_{n-1})$ (sobrediagonal da mesma matriz);
- dar como resultados: dois vetores $\ell = (\ell_1, \ldots, \ell_{n-1})$ e $u = (u_1, \ldots, u_n)$ correspondentes à subdiagonal de L e diagonal de U da decomposição LU da matriz dada.

d) Obtenha a decomposição LU da matriz K_5, usando a função $\mathbf{luTrid}$ e use-a para calcular a segunda coluna da matriz K_5^{-1}.

Nota: Pode formar as matrizes L e U e usar as funções $\mathbf{subDireta}$ e $\mathbf{subInversa}$, ou modificar essas funções de modo a adaptá-las ao caso em questão.

Trabalho 4. **Método SR(ω)**

O *método de relaxação sucessiva* com parâmetro ω (método SR(ω)), para a resolução de um sistema $Ax = b$, é um método iterativo definido por:

Para $k = 0, 1, 2, \ldots,$

$$\left.\begin{array}{l} x_i^{(k+1)} = x_i^{(k)} + \frac{\omega}{a_{ii}}(b_i - \sum\limits_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum\limits_{j=i}^{n} a_{ij} x_j^{(k)}); i = 1, \ldots, n, \\ x_i^{(0)};\ i = 1, \ldots, n,\ \ \omega \in \mathbb{R},\ \ \text{dados.} \end{array}\right\} \tag{2}$$

a) Mostre que o passo k do método SR(ω) pode ser descrito matricialmente como

$$(D + \omega M)x^{(k+1)} = [(1 - \omega)D - \omega N]x^{(k)} + \omega b,$$

onde D, M e N são as matrizes usuais utilizadas na definição matricial dos métodos de Jacobi e Gauss-Seidel.

b) Escreva uma função $x = \mathbf{metSR}(A, b, \omega, tol, maxit)$ para implementar o método de relaxação sucessiva, com especificações idênticas às construídas para os métodos de Jacobi e Gauss-Seidel.

Nota: Tenha em atenção que, neste caso, a função deve aceitar um argumento adicional, valor do parâmetro de relaxação ω.

c) Pode provar-se que é condição necessária de convergência do método SR(ω) que $0 < \omega < 2$; além disso, para $\omega = 1$, o método SR(ω) é simplesmente o método de Gauss-Seidel.

Teste a sua função na resolução de diversos sistemas de equações por si escolhidos, fazendo variar o valor do parâmetro de relaxação e dando exemplos em que, para certos valores do parâmetro, o método convirja mais rapidamente que o método de Gauss-Seidel; ilustre também a afirmação quanto à condição imposta ao parâmetro para que o método possa convergir.