Exercícios 4

Prazo para entrega: segunda-feira 11/11 até as 23:59h.
Enviar relatório em formato PDF para `bernardomartinsrocha@gmail.com`

Problema 1

Implemente o algoritmo discutido em sala de aula para determinar a matriz inversa $\mathbf{A}^{-1}$ de uma matriz $\mathbf{A}$ que seja não-singular. Teste sua implementação com algumas matrizes de exemplo e comprove que a matriz obtida para cada caso é, de fato, a matriz inversa.

Você deve criar uma função `inversa` que recebe $\mathbf{A}$ como argumento e retorna a sua inversa. Sua implementação deve:

(a) verificar se $\mathbf{A}$ é invertível ou não; caso não seja, imprima uma mensagem de erro;

(b) utilizar a decomposição $\mathbf{A} = \mathbf{LU}$ de forma eficiente.

Problema 2

Implemente os métodos iterativos de Jacobi e Gauss-Seidel para a solução de sistemas de equações lineares $\mathbf{Ax} = \mathbf{b}$. As suas implementações devem:

(a) verificar se: para o método de Jacobi, o critério das linhas é satisfeito; ou para o método de Gauss-Seidel o critério de Sassenfeld é satisfeito antes de realizar a solução do sistema de equações lineares.

(b) estabelecer um critério de parada (erro absoluto ou relativo) e um número máximo de iterações;

(c) retornar a aproximação obtida para a solução e informar (ou retornar) o número de iterações realizadas.

Verifique sua implementação resolvendo alguns dos sistemas de equações lineares apresentados durante as aulas (quadro/slides).

Problema 3

Sistemas **mal condicionados** são aqueles onde pequenas modificações nos coeficientes ou constantes do sistema resultam em grandes modificações na solução do mesmo. Uma forma de se avaliar o mal condicionamento de um sistema $\mathbf{Ax} = \mathbf{b}$, é através do número de condição da matriz $\mathbf{A}$, dado por

$$\text{cond}(\mathbf{A}) = \|\mathbf{A}\|\|\mathbf{A}^{-1}\|$$

onde $\|\mathbf{A}\|$ representa uma norma matricial, que na linguagem de programação Python esta pode ser calculada usando `np.linalg.norm(A,p=2)`, onde `p=2` indica que a norma-2 será calculada.

Utilizando a função `inversa()` do Exercício 1, implementa uma função chamada `condMatriz()` que recebe uma matriz como argumento, calcula e retorne o seu número de condicionamento.

Teste sua implementação calculando o número de condição da matriz $M = [[5, -2],\ [-2, 4]]$ que é aproximadamente 2.69.

Problema 4

Seja o sistema linear $\mathbf{Hx} = \mathbf{b}$, onde a matriz de coeficientes e o vetor são dados por:

$$\mathbf{H} = H_{ij} = \frac{1}{i+j+1}, \quad \text{e} \quad \mathbf{b} = b_i = \frac{1}{i+n+1}$$

onde n denota a dimensão da matriz. Considere a matriz $\mathbf{H}$ com diferentes dimensões $n = 5, 10, 100, 1000$; e faça:

(a) Resolva o sistema de equações utilizando o método da eliminação de Gauss **sem** e **com** o uso da estratégia de pivotamento;

(b) Resolva o sistema de equações utilizando o método da decomposição LU;

(c) Determine o erro cometido, por cada um dos métodos utilizados, através do **resíduo** calculado na norma do máximo, o qual é dado por:

$$r = \|\mathbf{Hx} - \mathbf{b}\|_\infty = \max_{1 \leq i \leq n} |H_{ij}x_j - b_i|, \quad \text{para} \quad i = 1, \ldots, n.$$

Calcule também o número de condição $\text{cond}(\mathbf{H})$ da matriz. Discuta e apresente os resultados em forma de tabela.