Dr. A. Jha
M.Sc. A. Khan

MA637 Numerical Analysis & Computing

Winter Semester 24/25

24.02.2025

Assigment 6

Assignment and programming exercises must be completed in groups of 4 members and submitted via Google Classroom by 24.03.2025 at 23:59. If the assignment is not prepared using LaTeX, a clear scanned copy of the handwritten work must be uploaded. Ensure that the names and enrollment numbers of all group members are clearly written on the submission. Late submissions will not be accepted.

Question 1: [Eigenvalues and Eigenvectors]

a) Let λ be an eigenvalue of a $n \times n$ matrix $\mathbf{A}$ and $\mathbf{x} \neq \mathbf{0}$ be the associated eigenvector

- **i)** Show that λ is an eigenvalue of $\mathbf{A}^\top$.
- **ii)** For any integer $k \geq 1$, λ^k is an eigenvalue of $\mathbf{A}^k$ with eigenvector $\mathbf{x}$.
- **iii)** If $\mathbf{A}^{-1}$ exists, then $1/\lambda$ is an eigenvalue of $\mathbf{A}^{-1}$ with eigenvector $\mathbf{x}$.

b) Let $\rho(\mathbf{A})$ denote the spectral radius of $\mathbf{A}$. Prove or disprove that $\rho(\,\cdot\,)$ is a matrix norm.

Question 2: [Gauss-Seidel Method]

Let $\mathbf{A}$ be a symmetric positive definite matrix of size $n \times n$.

a) Show that we can write $\mathbf{A} = \mathbf{D} - \mathbf{L} - \mathbf{L}^\top$ where $d_{ii} > 0$ for $i = 1, 2, \ldots, n$, and $\mathbf{L}$ is lower triangular matrix. Further show that $\mathbf{D} - \mathbf{L}$ is non-singular.

b) Let $\mathbf{T}_{\text{GS}} = (\mathbf{D} - \mathbf{L})^{-1} \mathbf{L}^\top$ and $\mathbf{P} = \mathbf{A} - \mathbf{T}_{\text{GS}}^\top \mathbf{A} \mathbf{T}_{\text{GS}}$. Show that $\mathbf{P}$ is symmetric.

c) Show that $\mathbf{T}_{\text{GS}}$ can also be written as $\mathbf{T}_{\text{GS}} = \mathbf{I} - (\mathbf{D} - \mathbf{L})^{-1} \mathbf{A}$.

d) Let λ be an eigenvalue of $\mathbf{T}_{\text{GS}}$ with eigenvector $\mathbf{x} \neq \mathbf{0}$. Use **b)** to show that $\mathbf{x}^\top \mathbf{P} \mathbf{x} > 0$ implies $|\lambda| < 1$.

Question 3: [Condition Number]

Show that if $\mathbf{B}$ is singular, then

$$\frac{1}{\kappa(\mathbf{A})} = \frac{\|\mathbf{A} - \mathbf{B}\|}{\|\mathbf{A}\|},$$

where $\kappa(\mathbf{A})$ denotes the condition number of $\mathbf{A}$. Using this estimator find the condition number of:

a) $\begin{bmatrix} 1 & 2 \\ 1.0001 & 2 \end{bmatrix}$

b) $\begin{bmatrix} 3.9 & 1.6 \\ 6.8 & 2.9 \end{bmatrix}$.

Hint: Remember that $\mathbf{B}$ is singular hence you can find a vector $\mathbf{x} \neq \mathbf{0}$ such that $\mathbf{B}\mathbf{x} = \mathbf{0}$ and $\|\mathbf{x}\| = 1$.

Question 4: [**Programming Exercise**]

The goal of this exercise is to understand iterative methods and analyze its accuracy for different system of equations. Consider the system of equation $\mathbf{A}\mathbf{x} = \mathbf{b}$ where

$$\mathbf{A} = \begin{bmatrix} 4 & 0 & 2 & 1 \\ 0 & 3 & -1 & 1 \\ 2 & -1 & 6 & 3 \\ 1 & 1 & 3 & 8 \end{bmatrix}, \qquad \text{and} \qquad \mathbf{b} = \begin{bmatrix} -8 \\ -20 \\ -2 \\ 4 \end{bmatrix}.$$

a) Write a code that computes $\mathbf{x}$ for the above system using the Jacobi, Gauss-Seidel, and SOR iteration. Use $\mathbf{x}^{(0)} = \mathbf{0}$, `tolerance` $= 10^{-4}$ and maximum number of iterations as 100.

b) In **a)** for the SOR iteration take $\omega \in \{0.1, 0.3, 0.5, \dots, 1.9\}$. Which ω has the least number of iterations. What happens if you take $\omega = 2.2$?

c) Consider the system of equation

$$\begin{bmatrix} 4 & 3 & 0 \\ 3 & 4 & -1 \\ 0 & -1 & 4 \end{bmatrix} \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \mathbf{x}_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 0 \end{bmatrix}.$$

Compute the optimal ω using the formula

$$\omega_{\text{opt}} = \frac{2}{1 + \sqrt{1 - [\rho(\mathbf{T}_{\text{J}})]^2}},$$

where $\rho(\mathbf{T}_{\text{J}})$ is the spectral radius of the Jacobi iteration matrix. Using this ω_{opt} compute the solution of the above system of equation. Take a value of ω other than ω_{opt}, what do you observe?

Hint: In this programming exercise, you would require the following programming concepts:

a) `for` loops.

b) `if-else` structure.

c) Function declaration using `def`.

d) `numpy` library:

   i) `array`: For creating arrays and matrices.

   ii) `size`: To get the number of elements in the numpy array.

iii) `zeros`: To create an array with entries as zero.

iv) `linalg.eigvalsh`: To compute the eigenvalues of **A**.

v) `max`: To compute the maximum number.

vi) `sqrt`: To compute the square root of a number.

vii) `dot`: To compute the product of two matrices.

viii) `copy`: To copy a vector.