**MTech CSE – 1st Semester**

**Student Name:** SIDDHARTH KUMAR
**Student ID:** A125021

# Question 4

Explain the LU decomposition of a matrix using Gaussian Elimination. Clearly describe each step involved in the process.

# Answer:

# Overview of LU Decomposition

LU decomposition is a matrix factorization technique in which a square matrix $A$ is expressed as the product of two triangular matrices:

$$A = LU,$$

where:

- $L$ is a lower triangular matrix with all diagonal entries equal to 1,
- $U$ is an upper triangular matrix.

This decomposition provides a systematic way to represent Gaussian elimination in matrix form and is widely used for solving systems of linear equations and other numerical computations.

# Connection with Gaussian Elimination

Gaussian elimination transforms a matrix into an upper triangular form by eliminating entries below the diagonal using elementary row operations.

LU decomposition records this process as follows:

- the elimination multipliers are stored in the matrix $L$,
- the resulting upper triangular matrix is stored in $U$.

Thus, LU decomposition is essentially Gaussian elimination written as a matrix factorization.

# Conditions for Existence

An LU decomposition without pivoting exists if Gaussian elimination can be performed without row exchanges. This requires all leading principal pivots of $A$ to be nonzero.

If row interchanges are necessary, the decomposition takes the form:

$$PA = LU,$$

where $P$ is a permutation matrix.

# Step-by-Step LU Decomposition Using Gaussian Elimination

## Step 1: Initial Matrix

Let:

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix}.$$

The goal is to eliminate all entries below the diagonal, column by column.

## Step 2: Elimination in the First Column

To eliminate entries below $a_{11}$, compute the multipliers:

$$m_{i1} = \frac{a_{i1}}{a_{11}}, \quad i = 2, 3, \ldots, n.$$

Apply row operations:

$$R_i \leftarrow R_i - m_{i1} R_1.$$

These operations create zeros below the first pivot and initiate the formation of the matrix $U$.

### Step 3: Forming the $L$ Matrix

The multipliers used in elimination are stored in the lower triangular matrix $L$:

$$L_{ii} = 1, \quad L_{i1} = m_{i1} \text{ for } i > 1.$$

The diagonal entries of $L$ are fixed as 1 to ensure uniqueness.

### Step 4: Repeating for Remaining Columns

For column $k$, where $k = 2, 3, \ldots, n-1$, compute:

$$m_{ik} = \frac{u_{ik}}{u_{kk}}, \quad i = k+1, \ldots, n.$$

Apply:

$$R_i \leftarrow R_i - m_{ik} R_k,$$

and store the multipliers in $L$ as $L_{ik} = m_{ik}$.

After completing these steps, the matrix becomes upper triangular and is denoted by $U$.

## Final LU Factorization

At the end of Gaussian elimination:

- $U$ contains the transformed upper triangular matrix,
- $L$ contains all elimination multipliers below the diagonal.

Hence,

$$A = LU.$$

## Solving Linear Systems Using LU Decomposition

Given a system:

$$Ax = b,$$

and $A = LU$, the solution is obtained in two stages:

- solve $Ly = b$ using forward substitution,

- solve $Ux = y$ using backward substitution.

This approach is computationally efficient, especially when solving multiple systems with the same coefficient matrix.

## Illustrative Numerical Example

Consider:

$$A = \begin{bmatrix} 2 & 3 \\ 4 & 7 \end{bmatrix}.$$

The multiplier is $m_{21} = 2$. Thus:

$$L = \begin{bmatrix} 1 & 0 \\ 2 & 1 \end{bmatrix}, \quad U = \begin{bmatrix} 2 & 3 \\ 0 & 1 \end{bmatrix}.$$

It is easily verified that $A = LU$.

## LU Decomposition with Pivoting

If a pivot element is zero or very small, Gaussian elimination requires row interchanges to proceed. These row swaps are captured by a permutation matrix $P$, leading to the factorization:

$$PA = LU.$$

Partial pivoting improves numerical stability and is used in most practical implementations of LU decomposition.

## Conclusion

LU decomposition provides a structured representation of Gaussian elimination by factoring a matrix into lower and upper triangular components. By separating the elimination process from the solution phase, it enables efficient and stable methods for solving linear systems and performing numerical computations.