Numerical Linear Algebra

Dr. Mriganka Mandal

Assistant Professor

Department of Mathematics

Indian Institute of Technology Jodhpur

NH 65 Nagaur Road, Karwar 342037, India

Pseudocode for LU Decomposition Method

Step – 1:

```
n = size(A)
L = create_identity_matrix(n)
U = create_empty_matrix(n, n)
```

Step – 2: Apply Decomposition on Matrix A:

```
for k from 1 to n:
    U[k][k] = A[k][k]
    for i from k+1 to n:
        L[i][k] = A[i][k] / U[k][k]
        U[k][i] = A[k][i]
    end
    for i from k+1 to n:
        for j from k+1 to n:
            A[i][j] = A[i][j] - L[i][k] * U[k][j]
        end
    end
end

return L, U
```

Step – 3:

```
n = size(L)
y = create_empty_vector(n)
x = create_empty_vector(n)
```

Step – 4: Forward Substitution to Solve Ly = b

```
for i from 1 to n:
    y[i] = b[i]
    for j from 1 to i-1:
        y[i] = y[i] - L[i][j] * y[j]
    end
    y[i] = y[i] / L[i][i]
end
```

Step – 5: Backward Substitution to Solve Ux = y:

```
for i from n to 1 step -1:
    x[i] = y[i]
    for j from i+1 to n:
        x[i] = x[i] - U[i][j] * x[j]
    end
    x[i] = x[i] / U[i][i]
end
```

Step – 6: Stop

Assignment – 2:

Q1. What is the worst-case running time of the LU Decomposition Method for an n-by-n matrix in terms of computational complexity notation?

Q2. Compare the running time between Gauss Elimination and LU Decomposition Methods.

Q3. Which method is better? Why?

Solving Ax = b Using LU Decomposition in Julia

 # Import the LinearAlgebra library

Command: using LinearAlgebra

 # Define the coefficient matrix A and the right-hand side vector b

Command: A = [2.0 1.0 -1.0;

```
-3.0 -1.0 2.0;
-2.0 1.0 2.0]
```

Command: b = [8.0, -11.0, -3.0]

Perform LU decomposition

Command: lu_result = lu(A)

Extract the L and U matrices from the result

Command: L = lu_result.L

Command: U = lu_result.U

Solve for y in Ly = b using forward substitution

Command: y = L \ b

Solve for x in Ux = y using backward substitution

Command: x = U \ y

Display the solution

Command: println("Solution x:")

Command: println(x)

Thank You