Experiment: 7
Power Method and LU Decomposition

1. Power Method

Implement this algorithm as MATLAB function script and find the given dominant eigen value and eigen vector of the following matrice.

(a) Using Power method find the dominant eigen value and eigen vector of the below with initial guess as $(1,1)^T$ and tolerance 0.0001.

$$\begin{bmatrix} 2 & -12 \\ 1 & -5 \end{bmatrix}$$

(b) Using Power method find the dominant eigen value and eigen vector of the below with initial guess as $(1,1)^T$ and tolerance 0.001.

$$\begin{bmatrix} -2 & -3 \\ 6 & 7 \end{bmatrix}$$

Algorithm

// Input: Matrix A, Tolerance, Initial guess, Maximum number of iterations
// Output: Dominant eigen value and eigen vector

- set counter=1.
- `While` $counter <= N$ % for running iterations
  (a) $x_1 = Ax_0$
  (b) $x_2 = Ax_1$
  (c) $[p1, k1] = max(abs(x1))$
  (d) $[p2, k2] = max(abs(x2))$
  (e) $a = x2./x2(k2,1) - x1./x1(k1,1)$
  (f) `if` $||a||_\infty$ is less than tol
  (g) `Print` the $x2./x2(k2,1)$ as eigen vector and $x2(k2,1)/x1(k1,1)$ as eigen value.
  (h) `return`;
  `end if`
  (i) Set `counter =counter+1`
  (j) Repeat the above steps.

  `end while`
- `Print` "Method fails to converge to given tol in given number of iterations"

2. **LU Decomposition**

Implement this algorithm as MATLAB function script and use it to attempt the following problems:

(a) Find the LU factorization of following matrix

$$A = \begin{bmatrix} 1 & -3 & 5 \\ 2 & -4 & 7 \\ -1 & -2 & 1 \end{bmatrix}$$

(b) Solve the system $Ax = b$, where A is provided above and $b = \begin{bmatrix} 1 \\ 1 \\ 3 \end{bmatrix}$.

Algorithm

// Input: Matrix A

// Output: Matrix L and U.

- Create a function which take the matrix A as input.
- Define dimension of matrix A(say $m \times n$)
- if A is not a square matrix
  Decomposition does not exists
  return to main script
  end if.
- for (j) 1 to n
  if determinant of j^{th} principal minor is non-zero (that is, matrix with rows and columns from 1 to i and 1 to j), then append 1 to a temporary array 1.
  end if.
  nested for rows(i) 1 to n
    find sum of absolute values of entries in each rows except the one on the principal diagonal.
    if the absolute value of entry on principal diagonal is strictly greater than the above sum, in that row, then append 1 to temporary array 2.
    end if.
    end nested for.
  end for.

- if sum of both temporary arrays are not equal to the order of the matrix.
  then decomposition does not exists
  return to main script
  end if.

- define L as identity matrix of same order and $U = A$.

- for column(j): 1 to $n-1$
  nested for rows(i): $j+1$ to n
  define $L(i,j)$ as $\frac{U(i,j)}{U(j,j)}$
  apply row operation on U:$R_i \rightarrow R_i - \frac{U(i,j)}{U(j,j)}R_j$
  end nested for.
  end for.