Experiment:4
Solving systems using Direct methods

1. Row-Reduced Echelon Form

Create a code in MATLAB that can be called as a function (name it *r_r_e_f.m*) that gives row reduced echelon form of any given vector and also its rank. Find the rref of A and B using the above mentioned function:

$$A = \begin{bmatrix} 1 & 3 & 4 \\ 5 & -9 & -8 \\ 4 & 7 & 8 \end{bmatrix} \text{ and } B = \begin{bmatrix} 1 & 6 & 7 \\ -1 & -5 & 1 \\ 1 & 9 & 0 \end{bmatrix}$$

Algorithm

// Input: $m \times n$ matrix A

// Output:$m \times n$ matrix in reduced row echelon form and rank of matrix

(a) Create a function which take the matrix A as input

(b) Set i = 1, j = 1, rank = 0

(c) While $i \leq m$ and $j \leq n$ (m=number of rows and n=number of columns)

(d) Find position of maximum absolute value(Pivot element and position) from $A(i, j)$, $i \geq j$.

(e) If pivot is equal to zero(or less than some tolerance value)

   i. Set j = j + 1
      else

   ii. Perform $R_i \longleftrightarrow R_{pivot}$

   iii. Divide each element of row i by a_{ij}, thus making the pivot a_{ij} equal to one

   iv. For each row k from 1 to m, with $k \neq i$ subtract row i multiplied by a_{kj} from row k.

   v. Set $i = i + 1$, $j = j + 1$, $rank = rank + 1$;
      end if
      end while

(f) Return transformed matrix A and rank.

2. Solving system using Gauss-Elimination method

Create a code in MATLAB that can be called as a function (name it *Gauss_El.m*) that;

(a) gives the solution of a system of linear equation $Ax = b$ using Gauss-elimination method, where A is a matrix of order $m \times n$, x is the array of n unknowns and

b is a $m-$array, check whether the solution of the system $Ax = b$ is inconsistent, infinitely many or unique solution and, compute the solution in the case of unique solution.

(b) call this function in MATLAB to solve for following:

$$A = \begin{bmatrix} 1 & -1 & 1 & -1 \\ 2 & 1 & 1 & 1 \\ 7 & -2 & -1 & -1 \end{bmatrix}, b = \begin{bmatrix} -2 \\ 5 \\ 5 \end{bmatrix} \text{ and } A = \begin{bmatrix} 0 & 2 & 4 & 1 \\ 2 & 0 & 5 & 0 \\ 1 & 0 & 1 & 2 \\ 1 & 1 & 3 & 1 \end{bmatrix}, b = \begin{bmatrix} 4 \\ 3 \\ 2 \\ 1 \end{bmatrix}$$

Algorithm

// Input: $m \times n$ matrix A and $m \times 1$ matrix b

// Output: $n \times 1$ matrix x such that $Ax = b$

(a) Create a function which take the matrix A and b as input.

(b) Call the *r_e_f* function (discussed earlier in the class) for the matrix A and get the rank of matrix A.

(c) Call the *r_e_f* function for the augmented matrix $[A \mid b]$ and get the ref form and rank.

(d) Assign first n columns of the augmented matrix to a new matrix C and its last column to a new column vector d (to get equivalent system $Cx = d$).

(e) If ranks of A and $[A \mid b]$ are not equal:

   i. return *Inconsistent Solution - No Solution*

(f) If ranks are equal and rank of A is less than number of unknowns:

   i. return *Consistent Solution - Infinitely Many Solutions*

(g) Otherwise, there will be a unique solution:

   i. define solution vector as $n \times 1$ vector

   ii. apply back substitution method and assign the values in solution vector.

      A. get the value of last variable.

      B. get the value of second last variable by substituting the value of last variable.

      C. continue this till you get the value of first variable.

   iii. return the solution vector.

3. Solving system using Gauss-Jordan-Elimination

Create a code in MATLAB that can be called as a function (name it *Gauss_Jord.m*) that;

(a) gives the solution of a system of linear equation $Ax = b$ using Gauss-Jordan Elimination method, where A is a matrix of order $m \times n$, x is the array of n unknowns and b is a $m-$array, check whether the solution of the system $Ax = b$ is inconsistent, infinitely many or unique solution and, compute the solution in the case of unique solution.

(b) call this function in MATLAB to solve for following:

i. $A = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 1 & 12 \\ 13 & 14 & 15 & 16 \end{bmatrix}$ and $b = \begin{bmatrix} 7 \\ 8 \\ 9 \\ 11 \end{bmatrix}$

ii. $A = \begin{bmatrix} 2 & 1 & 1 & 3 \\ 2 & 2 & 0 & 3 \\ 1 & 1 & -2 & 0 \end{bmatrix}$ and $b = \begin{bmatrix} 10 \\ 11 \\ -1 \end{bmatrix}$

Algorithm

// Input: $m \times n$ matrix A and $m \times 1$ matrix b
// Output: $n \times 1$ matrix x such that $Ax = b$

(a) Create a function which take the matrix A and b as input.

(b) Call the *r_r_e_f* function (discussed earlier in the class) for the matrix A and get the rank of matrix A.

(c) Call the *r_r_e_f* function for the augmented matrix $[A \mid b]$ and get the rref form and rank.

(d) If ranks of A and $[A \mid b]$ are not equal:

   i. return *Inconsistent Solution - No Solution*

(e) If ranks are equal and rank of A is less than number of unknowns:

   i. return *Consistent Solution - Infinitely Many Solutions*

(f) Otherwise, there will be a unique solution:

   i. return the last column of rref of the augmented matrix.