ME 542

Assignment -4

Stationary Iterative Methods for Linear System of Equations

- Students need to save all the programs in a zipped file. Name it with your roll number and submit it on MS TEAMS.
- The programs are to be compiled and checked before submitting.
- Results obtained by your code should be written (do not copy the image file of your run) in a pdf file, and keep this file also in the same zipped folder.
- Make one program that can solve all problems.

1. Solve the following systems of linear equations using Jacobi Method for different values of ε for terminating the algorithm. Use the same termination condition as discussed in the lecture.

   a.

$$0.2x_1 + 0.1x_2 + x_3 + x_4 = 1$$
$$0.1x_1 + 4x_2 - x_3 + x_4 - x_5 = 2$$
$$x_1 - x_2 + 60x_3 - 2x_5 = 3$$
$$x_1 + x_2 + 8x_4 + 4x_5 = 4$$
$$-x_2 - 2x_3 + 4x_4 + 700x_5 = 5$$

   b.

(1) $x_1 + x_2 + x_3 = 7$

(2) $x_1 + 2x_2 + 2x_3 = 13$

(3) $x_1 + 3x_2 + x_3 = 13$

   c. Are you able to solve (b)? Try exchanging equation (2) and equation (3) and solve. Check the diagonal dominant feature for both methods.

2. Solve the above questions using Gauss-Seidel Method.
3. Calculate the exact number of function evaluations for both the methods and compare for different values of ε. Include the results in a pdf file.

7.3 The Jacobi and Gauss-Seidel Iterative Methods

The Jacobi Method

***Two assumptions made on Jacobi Method*:**

1. The system given by

$$\begin{gathered} a_{11}x_1 + a_{12}x_2 + \cdots a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots a_{2n}x_n = b_2 \\ \vdots \\ a_{n1}x_1 + a_{n2}x_2 + \cdots a_{nn}x_n = b_n \end{gathered}$$

Has a unique solution.

2. The coefficient matrix A has no zeros on its main diagonal, namely, $a_{11}, a_{22}, \dots, a_{nn}$ are nonzeros.

<u>Main idea of Jacobi</u>

To begin, solve the 1st equation for x_1, the 2nd equation for x_2 and so on to obtain the rewritten equations:

$$x_1 = \frac{1}{a_{11}}(b_1 - a_{12}x_2 - a_{13}x_3 - \cdots a_{1n}x_n)$$

$$x_2 = \frac{1}{a_{22}}(b_2 - a_{21}x_1 - a_{23}x_3 - \cdots a_{2n}x_n)$$

$$\vdots$$

$$x_n = \frac{1}{a_{nn}}(b_n - a_{n1}x_1 - a_{n2}x_2 - \cdots a_{n,n-1}x_{n-1})$$

Then make an initial guess of the solution $\boldsymbol{x}^{(0)} = (x_1^{(0)}, x_2^{(0)}, x_3^{(0)}, \dots x_n^{(0)})$. Substitute these values into the right hand side the of the rewritten equations to obtain the *first approximation*, $\left(x_1^{(1)}, x_2^{(1)}, x_3^{(1)}, \dots x_n^{(1)}\right)$.

This accomplishes one **iteration**.

In the same way, the *second approximation* $\left(x_1^{(2)}, x_2^{(2)}, x_3^{(2)}, \dots x_n^{(2)}\right)$ is computed by substituting the first approximation's x-vales into the right hand side of the rewritten equations.

By repeated iterations, we form a sequence of approximations $\boldsymbol{x}^{(k)} = \left(x_1^{(k)}, x_2^{(k)}, x_3^{(k)}, \dots x_n^{(k)}\right)^t, \quad k = 1,2,3, \dots$

__The Jacobi Method.__ For each $k \geq 1$, generate the components $x_i^{(k)}$ of $\boldsymbol{x}^{(\boldsymbol{k})}$ from $\boldsymbol{x}^{(\boldsymbol{k-1})}$ by

$$x_i^{(k)} = \frac{1}{a_{ii}}\left[\sum_{\substack{j=1,\\ j\neq i}}^{n}(-a_{ij}x_j^{(k-1)}) + b_i\right], \qquad \text{for } i = 1,2,,\ldots n$$

Example. Apply the Jacobi method to solve

$$\begin{aligned} 5x_1 - 2x_2 + 3x_n &= -1 \\ -3x_1 + 9x_2 + x_n &= 2 \\ 2x_1 - x_2 - 7x_n &= 3 \end{aligned}$$

Continue iterations until two successive approximations are identical when rounded to three significant digits.

Solution To begin, rewrite the system

$$x_1 = \frac{-1}{5} + \frac{2}{5}x_2 - \frac{3}{5}x_3$$

$$x_2 = \frac{2}{9} + \frac{3}{9}x_1 - \frac{1}{9}x_3$$

$$x_3 = -\frac{3}{7} + \frac{2}{7}x_1 - \frac{1}{7}x_2$$

Choose the initial guess $x_1 = 0, x_2 = 0, x_3 = 0$

The first approximation is

$$x_1^{(1)} = \frac{-1}{5} + \frac{2}{5}(0) - \frac{3}{5}(0) = -0.200$$

$$x_2^{(1)} = \frac{2}{9} + \frac{3}{9}(0) - \frac{1}{9}(0) = 0.222$$

$$x_3^{(1)} = -\frac{3}{7} + \frac{2}{7}(0) - \frac{1}{7}(0) = -0.429$$

Continue iteration, we obtain

n	$k=0$	$k=1$	$k=2$	$k=3$	$k=4$	$k=5$	$k=6$
$x_1^{(k)}$	0.000	-0.200	0.146	0.192			
$x_2^{(k)}$	0.000	0.222	0.203	0.328			
$x_2^{(k)}$	0.000	-0.429	-0.517	-0.416			

<u>The Jacobi Method in Matrix Form</u>

Consider to solve an $n \times n$ size system of linear equations $A\boldsymbol{x} = \boldsymbol{b}$ with $A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{bmatrix}$ and $\boldsymbol{b} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}$ for $\boldsymbol{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$.

We split A into

$$A = \begin{bmatrix} a_{11} & 0 & \dots & 0 \\ 0 & a_{22} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & a_{nn} \end{bmatrix} - \begin{bmatrix} 0 & \dots & 0 & 0 \\ -a_{21} & \dots & 0 & 0 \\ \vdots & & \ddots & \vdots \\ -a_{n1} & \dots & -a_{n,n-1} & 0 \end{bmatrix} - \begin{bmatrix} 0 & -a_{12} & \dots & -a_{1n} \\ 0 & 0 & & \vdots \\ \vdots & \vdots & \ddots & -a_{n-1,n} \\ 0 & 0 & \dots & 0 \end{bmatrix} = D - L - U$$

$A\boldsymbol{x} = \boldsymbol{b}$ is transformed into $(D - L - U)\boldsymbol{x} = \boldsymbol{b}$

$$D\boldsymbol{x} = (L + U)\boldsymbol{x} + \boldsymbol{b}$$

Assume D^{-1} exists and $D^{-1} = \begin{bmatrix} \frac{1}{a_{11}} & 0 & \dots & 0 \\ 0 & \frac{1}{a_{22}} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \frac{1}{a_{nn}} \end{bmatrix}$

Then

$$\boldsymbol{x} = D^{-1}(L + U)\boldsymbol{x} + D^{-1}\boldsymbol{b}$$

The matrix form of Jacobi iterative method is

$$\boldsymbol{x}^{(k)} = D^{-1}(L+U)\boldsymbol{x}^{(k-1)} + D^{-1}\boldsymbol{b} \qquad k = 1,2,3,\ldots$$

Define $T = D^{-1}(L+U)$ and $\boldsymbol{c} = D^{-1}\boldsymbol{b}$, Jacobi iteration method can also be written as

$$\boldsymbol{x}^{(k)} = T\boldsymbol{x}^{(k-1)} + \boldsymbol{c} \qquad k = 1,2,3,\ldots$$

<u>Numerical Algorithm of Jacobi Method</u>

Input: $A = [a_{ij}]$, $\boldsymbol{b}, \boldsymbol{XO} = \boldsymbol{x}^{(0)}$, tolerance TOL, maximum number of iterations N.

Step 1 Set $k = 1$

Step 2 while ($k \le N$) do Steps 3-6

Step 3 For for $i = 1,2,,\ldots n$

$$x_i = \frac{1}{a_{ii}}\left[\sum_{\substack{j=1,\\ j\neq i}}^{n}(-a_{ij}\boldsymbol{XO}_j) + b_i\right],$$

Step 4 If $||\boldsymbol{x} - \boldsymbol{XO}|| < TOL$, then OUTPUT $(x_1, x_2, x_3, \ldots x_n)$;

STOP.

Step 5 Set $k = k + 1$.

Step 6 For for $i = 1,2,,\ldots n$

Set $\boldsymbol{XO}_i = x_i$.

Step 7 OUTPUT $(x_1, x_2, x_3, \ldots x_n)$;

STOP.

Another stopping criterion in Step 4: $\frac{||\boldsymbol{x}^{(k)} - \boldsymbol{x}^{(k-1)}||}{||\boldsymbol{x}^{(k)}||}$

The Gauss-Seidel Method

<u>Main idea of Gauss-Seidel</u>

With the Jacobi method, the values of $x_i^{(k)}$ obtained in the kth iteration remain unchanged until the entire $(k+1)$th iteration has been calculated. With the Gauss-Seidel method, we use the new values $x_i^{(k+1)}$ as soon as they are known. For example, once we have computed $x_1^{(k+1)}$ from the first equation, its value is then used in the second equation to obtain the new $x_2^{(k+1)}$, and so on.

Example. Derive iteration equations for the Jacobi method and Gauss-Seidel method to solve

$$
\begin{gathered}
5x_1 - 2x_2 + 3x_n = -1 \\
-3x_1 + 9x_2 + x_n = 2 \\
2x_1 - x_2 - 7x_n = 3
\end{gathered}
$$

<u>The Gauss-Seidel Method.</u> For each $k \geq 1$, generate the components $x_i^{(k)}$ of $\boldsymbol{x}^{(\boldsymbol{k})}$ from $\boldsymbol{x}^{(\boldsymbol{k-1})}$ by

$$x_i^{(k)} = \frac{1}{a_{ii}}\left[-\sum_{j=1}^{i-1}(a_{ij}x_j^{(k)}) - \sum_{j=i+1}^{n}(a_{ij}x_j^{(k-1)}) + b_i\right], \qquad \text{for } i = 1,2,,\ldots n$$

Namely,

$$
\begin{gathered}
a_{11}x_1^{(k)} = -a_{12}x_2^{(k-1)} - \cdots - a_{1n}x_n^{(k-1)} + b_1 \\
a_{21}x_1^{(k)} + a_{22}x_2^{(k)} = -a_{23}x_3^{(k-1)} - \cdots - a_{2n}x_n^{(k-1)} + b_2 \\
\vdots \\
a_{n1}x_1^{(k)} + a_{n2}x_2^{(k)} + \cdots a_{nn}x_n^{(k)} = b_n
\end{gathered}
$$

Matrix form of Gauss-Seidel method.

$$(D-L)\boldsymbol{x}^{(\boldsymbol{k})} = U\boldsymbol{x}^{(k-1)} + \boldsymbol{b}$$

$$\boldsymbol{x}^{(\boldsymbol{k})} = (D-L)^{-1}U\boldsymbol{x}^{(k-1)} + (D-L)^{-1}\boldsymbol{b}$$

Define $T_g = (D-L)^{-1}U$ and $\boldsymbol{c}_g = (D-L)^{-1}\boldsymbol{b}$, Gauss-Seidel method can be written as

$$\boldsymbol{x}^{(k)} = T_g\boldsymbol{x}^{(k-1)} + \boldsymbol{c}_g \qquad k = 1,2,3,\ldots$$

Numerical Algorithm of Gauss-Seidel Method

Input: $A = [a_{ij}]$, $\boldsymbol{b}, \boldsymbol{XO} = \boldsymbol{x}^{(0)}$, tolerance TOL, maximum number of iterations N.

Step 1 Set $k = 1$

Step 2 while ($k \leq N$) do Steps 3-6

Step 3 For for $i = 1,2,, \dots n$

$x_i = \frac{1}{a_{ii}}\left[-\sum_{j=1}^{i-1}(a_{ij}x_j) - \sum_{j=i+1}^{n}(a_{ij}\boldsymbol{XO}_j) + b_i\right]$,

Step 4 If $||\boldsymbol{x} - \boldsymbol{XO}|| < TOL$, then OUTPUT $(x_1, x_2, x_3, \dots x_n)$;

STOP.

Step 5 Set $k = k + 1$.

Step 6 For for $i = 1,2,, \dots n$

Set $\boldsymbol{XO}_i = x_i$.

Step 7 OUTPUT $(x_1, x_2, x_3, \dots x_n)$;

STOP.

Convergence theorems of the iteration methods

Let the iteration method be written as

$$\boldsymbol{x}^{(k)} = T\boldsymbol{x}^{(k-1)} + \boldsymbol{c} \qquad \text{for each } k = 1,2,3,\dots$$

Lemma 7.18 If the spectral radius satisfies $\rho(T) < 1$, then $(I - T)^{-1}$ exists, and

$$(I - T)^{-1} = I + T + T^2 + \cdots = \sum_{j=0}^{\infty} T^j$$

Theorem 7.19 For any $\boldsymbol{x}^{(0)} \in R^n$, the sequence $\{\boldsymbol{x}^{(k)}\}_{k=0}^{\infty}$ defined by