Iteration Methods for Systems of Linear Equations

General method *x*=H*x*+g
Jacobi method
Gauss-Seidel method

Problem:
Solve the system of linear equations.

$$\begin{aligned} -1x_1 + 4x_2 \quad - 1x_4 &= 2 \\ +4x_1 - 1x_2 - x_3 \quad &= 1 \\ -1x_1 \quad + 4x_3 - 1x_4 &= 0 \\ -1x_2 - 1x_3 + 4x_4 &= 1 \end{aligned}$$

```
A = {{-1, 4, 0, -1}, {4, -1, -1, 0}, {-1, 0, 4, -1}, {0, -1, -1, 4}};
% // MatrixForm
```

$$\begin{pmatrix} -1 & 4 & 0 & -1 \\ 4 & -1 & -1 & 0 \\ -1 & 0 & 4 & -1 \\ 0 & -1 & -1 & 4 \end{pmatrix}$$

```
n = Length[A]
```

4

```
b = {2, 1, 0, 1}; b // MatrixForm
```

$$\begin{pmatrix} 2 \\ 1 \\ 0 \\ 1 \end{pmatrix}$$

Direct method solution through LinearSolve:

```
xr = LinearSolve[A, b]; xr // N // MatrixForm
```

$$\begin{pmatrix} 0.5 \\ 0.75 \\ 0.25 \\ 0.5 \end{pmatrix}$$


```
A.xr == b
```

True

Solution via General method : *x = H.x + g*

System of equations, iteration scheme:

$$\begin{aligned} -x_1 + 4x_2 \quad - x_4 = 2 \quad &\rightarrow x_1 = \quad 4x_2 \quad - x_4 - 2 \\ 4x_1 - x_2 - x_3 \quad = 1 \quad &\rightarrow x_2 = 4x_1 \quad - x_3 \quad - 1 \\ -x_1 \quad + 4x_3 - x_4 = 0 \quad &\rightarrow x_3 = x_1 \quad - 3x_3 + x_4 + 0 \\ -x_2 - x_3 + 4x_4 = 1 \quad &\rightarrow x_4 = \quad + x_2 + x_3 - 3x_4 + 1 \end{aligned}$$

Matrix H

```
H = {{0, 4, 0, -1}, {4, 0, -1, 0}, {1, 0, -3, 1}, {0, 1, 1, -3}};
H // MatrixForm
```

$$\begin{pmatrix} 0 & 4 & 0 & -1 \\ 4 & 0 & -1 & 0 \\ 1 & 0 & -3 & 1 \\ 0 & 1 & 1 & -3 \end{pmatrix}$$

Vector g

```
g = {-2, -1, 0, 1}; g // MatrixForm
```

$$\begin{pmatrix} -2 \\ -1 \\ 0 \\ 1 \end{pmatrix}$$

Initial vector x(0) and iteration sceme:

```
Clear[x];
x[0] = {0, 0, 0, 0};
pit = 20;
Do[
 x[k + 1] = H .x[k] + g // N;
 Print[k + 1, ". iteration: x=", x[k + 1]], {k, 0, pit}]
```

1. iteration: x={-2., -1., 0., 1.}

2. iteration: x={-7., -9., -1., -3.}

3. iteration: x={-35., -28., -7., 0.}

4. iteration: x={-114., -134., -14., -34.}

5. iteration: x={-504., -443., -106., -45.}

6. iteration: x={-1729., -1911., -231., -413.}

7. iteration: x={-7233., -6686., -1449., -902.}

8. iteration: x={-25844., -27484., -3788., -5428.}

9. iteration: x={-104510., -99589., -19908., -14987.}

10. iteration: x={-383371., -398133., -59773., -74535.}

11. iteration: x=$\{-1.518\times10^6, -1.47371\times10^6, -278587., -234300.\}$

12. iteration: x=$\{-5.66055\times10^6, -5.79341\times10^6, -916538., -1.0494\times10^6\}$

13. iteration: x=$\{-2.21242\times10^7, -2.17257\times10^7, -3.96033\times10^6, -3.56175\times10^6\}$

14. iteration: x=$\{-8.33409\times10^7, -8.45366\times10^7, -1.3805\times10^7, -1.50007\times10^7\}$

15. iteration: x=$\{-3.23146\times10^8, -3.19559\times10^8, -5.69267\times10^7, -5.33394\times10^7\}$

16. iteration: x=$\{-1.2249\times10^9, -1.23566\times10^9, -2.05705\times10^8, -2.16467\times10^8\}$

17. iteration: x=$\{-4.72616\times10^9, -4.69387\times10^9, -8.24246\times10^8, -7.91961\times10^8\}$

18. iteration: x=$\{-1.79835\times10^{10}, -1.80804\times10^{10}, -3.04538\times10^9, -3.14224\times10^9$

19. iteration: x=$\{-6.91793\times10^{10}, -6.88888\times10^{10}, -1.19896\times10^{10}, -1.16991\times10$

20. iteration: x=$\{-2.63856\times10^{11}, -2.64728\times10^{11}, -4.49095\times10^{10}, -4.57812\times10$

21. iteration: x=$\{-1.01313\times10^{12}, -1.01051\times10^{12}, -1.74909\times10^{11}, -1.72294\times10$

Iteration scheme diverges.

Theorem. The iteration scheme $x^{(k+1)} = H\,x^{(k)} + g$ converges if and only if spectral radius $\rho(H)$<1.
$\rho(H)=\max\{\,|\,\lambda_i\,|\},\ i = 1, \ \dots n$

```
Eigenvalues[H]
```

$\{-5, 1 + 2\sqrt{2}, -3, 1 - 2\sqrt{2}\}$


```
Eigenvalues[H] // N
```

{-5., 3.82843, -3., -1.82843}

```
ro = Max[Abs[Eigenvalues[H]]]
```

5

Iteration scheme diverges, ρ>1

Jacobi method and Gauss - Seidel method

Property I.
If matrix A is strictly diagonally dominant ($|a_{i\,i}| > \sum_{j=1}^{n} a_{i\,j},\ i \neq j$), Jacobi method and Gauss - Seidel method converge.
Methods sometimes converges even if this condition is not satisfied. It is necessary, that the diagonal elements in the matrix are greater in absolute value than the other elements.

Property II.
If matrix A is symmetric ($A = A^T$) and positive definite ($x^T.A.x > 0,\ x \neq \bar{0}$, Jacobi method and Gauss - Seidel method converge.

Originally Matrix A is not diagonally dominant. But after changing the order of the equations, rows reoder and the system matrix becomes diagonally dominant and symmetric.

$$A = \begin{pmatrix} -1 & 4 & 0 & -1 \\ 4 & -1 & -1 & 0 \\ -1 & 0 & 4 & -1 \\ 0 & -1 & -1 & 4 \end{pmatrix} = \begin{pmatrix} 4 & -1 & -1 & 0 \\ -1 & 4 & 0 & -1 \\ -1 & 0 & 4 & -1 \\ 0 & -1 & -1 & 4 \end{pmatrix}$$

Jacobi method

Loop through matrix notation

$$x^{k+1} = -D^{-1}.(M + N).x^k + D^{-1}.b$$

$$H = -D^{-1}.(M + N),\ g = D^{-1}.b$$

```
A = {{4, -1, -1, 0}, {-1, 4, 0, -1}, {-1, 0, 4, -1}, {0, -1, -1, 4}};
A // MatrixForm
```

$$\begin{pmatrix} 4 & -1 & -1 & 0 \\ -1 & 4 & 0 & -1 \\ -1 & 0 & 4 & -1 \\ 0 & -1 & -1 & 4 \end{pmatrix}$$

```
b = {1, 2, 0, 1}
```

{1, 2, 0, 1}

```
Diag = DiagonalMatrix[{4, 4, 4, 4}]; Diag // MatrixForm
```

$$\begin{pmatrix} 4 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 \\ 0 & 0 & 4 & 0 \\ 0 & 0 & 0 & 4 \end{pmatrix}$$

```
M = {{0, 0, 0, 0}, {-1, 0, 0, 0}, {-1, 0, 0, 0}, {0, -1, -1, 0}};
M // MatrixForm
```

$$\begin{pmatrix} 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & -1 & 0 \end{pmatrix}$$

```
NU = {{0, -1, -1, 0}, {0, 0, 0, -1}, {0, 0, 0, -1}, {0, 0, 0, 0}};
NU // MatrixForm
```

$$\begin{pmatrix} 0 & -1 & -1 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

```
A == M + Diag + NU
```

True

```
H = -Inverse[Diag].(M + NU); H // MatrixForm
```

$$\begin{pmatrix} 0 & \frac{1}{4} & \frac{1}{4} & 0 \\ \frac{1}{4} & 0 & 0 & \frac{1}{4} \\ \frac{1}{4} & 0 & 0 & \frac{1}{4} \\ 0 & \frac{1}{4} & \frac{1}{4} & 0 \end{pmatrix}$$

```
Eigenvalues[H]
```

$$\left\{-\frac{1}{2}, \frac{1}{2}, 0, 0\right\}$$

The Jacobi scheme will converge, $\rho<1$, $\rho = 1/2$.

```
g = Inverse[Diag].b
```

$$\left\{\frac{1}{4}, \frac{1}{2}, 0, \frac{1}{4}\right\}$$


```
Clear[x]
x[0] = {0, 0, 0, 0};
it = 20;
Do[
 x[k + 1] = H.x[k] + g;
 Print[k + 1, ". iteration: x=", x[k + 1] // N], {k, 0, it}]
```

```
1. iteration: x={0.25, 0.5, 0., 0.25}
2. iteration: x={0.375, 0.625, 0.125, 0.375}
3. iteration: x={0.4375, 0.6875, 0.1875, 0.4375}
4. iteration: x={0.46875, 0.71875, 0.21875, 0.46875}
5. iteration: x={0.484375, 0.734375, 0.234375, 0.484375}
6. iteration: x={0.492188, 0.742188, 0.242188, 0.492188}
7. iteration: x={0.496094, 0.746094, 0.246094, 0.496094}
8. iteration: x={0.498047, 0.748047, 0.248047, 0.498047}
9. iteration: x={0.499023, 0.749023, 0.249023, 0.499023}
10. iteration: x={0.499512, 0.749512, 0.249512, 0.499512}
11. iteration: x={0.499756, 0.749756, 0.249756, 0.499756}
12. iteration: x={0.499878, 0.749878, 0.249878, 0.499878}
13. iteration: x={0.499939, 0.749939, 0.249939, 0.499939}
14. iteration: x={0.499969, 0.749969, 0.249969, 0.499969}
15. iteration: x={0.499985, 0.749985, 0.249985, 0.499985}
16. iteration: x={0.499992, 0.749992, 0.249992, 0.499992}
17. iteration: x={0.499996, 0.749996, 0.249996, 0.499996}
18. iteration: x={0.499998, 0.749998, 0.249998, 0.499998}
19. iteration: x={0.499999, 0.749999, 0.249999, 0.499999}
20. iteration: x={0.5, 0.75, 0.25, 0.5}
21. iteration: x={0.5, 0.75, 0.25, 0.5}
```

The Jacobi scheme converges and the solution is reached in 20th iteration.

Find the norm of residual A.x - b for the 10th iteration of x, and it's accuracy (distance between x(10) and x(9)).

```
r = A.x[10] - b
```

$$\left\{-\frac{1}{1024}, -\frac{1}{1024}, -\frac{1}{1024}, -\frac{1}{1024}\right\}$$

```
nrJ = Sqrt[r.r]
```

$\frac{1}{512}$

```
nrJ // N
```

0.00195313

```
d = x[10] - x[9]
```

$\{\frac{1}{2048}, \frac{1}{2048}, \frac{1}{2048}, \frac{1}{2048}\}$

```
ndJ = Sqrt[d.d]
```

$\frac{1}{1024}$

```
ndJ // N
```

0.000488281

In Jacobi method, the norm of residual of the 10th iteration achieves 0.00195313 and the accuracy 0.000488281 .

Gauss-Seidel method

Loop through matrix notation

$$x^{k+1} = -(M+D)^{-1}.N.x^k + (M+D)^{-1}.b$$

$$H = -(M+D)^{-1}.N, \; g = (M+D)^{-1}.b$$

```
H = -Inverse[M + Diag].NU; H // MatrixForm
```

$$\begin{pmatrix} 0 & \frac{1}{4} & \frac{1}{4} & 0 \\ 0 & \frac{1}{16} & \frac{1}{16} & \frac{1}{4} \\ 0 & \frac{1}{16} & \frac{1}{16} & \frac{1}{4} \\ 0 & \frac{1}{32} & \frac{1}{32} & \frac{1}{8} \end{pmatrix}$$

```
Eigenvalues[H]
```

$\{\frac{1}{4}, 0, 0, 0\}$

The Gauss -Seidel scheme will converge, ρ<1, ρ = 1/4.

```
g = Inverse[M + Diag].b
```

$\{\frac{1}{4}, \frac{9}{16}, \frac{1}{16}, \frac{13}{32}\}$


```
Clear[x]
x[0] = {0, 0, 0, 0};
it = 20;
Do[
 x[k + 1] = H .x[k] + g;
 Print[k + 1, ". iteration: x=", x[k + 1] // N], {k, 0, it}]
```

```
1. iteration: x={0.25, 0.5625, 0.0625, 0.40625}
2. iteration: x={0.40625, 0.703125, 0.203125, 0.476563}
3. iteration: x={0.476563, 0.738281, 0.238281, 0.494141}
4. iteration: x={0.494141, 0.74707, 0.24707, 0.498535}
5. iteration: x={0.498535, 0.749268, 0.249268, 0.499634}
6. iteration: x={0.499634, 0.749817, 0.249817, 0.499908}
7. iteration: x={0.499908, 0.749954, 0.249954, 0.499977}
8. iteration: x={0.499977, 0.749989, 0.249989, 0.499994}
9. iteration: x={0.499994, 0.749997, 0.249997, 0.499999}
10. iteration: x={0.499999, 0.749999, 0.249999, 0.5}
11. iteration: x={0.5, 0.75, 0.25, 0.5}
12. iteration: x={0.5, 0.75, 0.25, 0.5}
13. iteration: x={0.5, 0.75, 0.25, 0.5}
14. iteration: x={0.5, 0.75, 0.25, 0.5}
15. iteration: x={0.5, 0.75, 0.25, 0.5}
16. iteration: x={0.5, 0.75, 0.25, 0.5}
17. iteration: x={0.5, 0.75, 0.25, 0.5}
18. iteration: x={0.5, 0.75, 0.25, 0.5}
19. iteration: x={0.5, 0.75, 0.25, 0.5}
20. iteration: x={0.5, 0.75, 0.25, 0.5}
21. iteration: x={0.5, 0.75, 0.25, 0.5}
```

The Gauss -Seidel scheme converges and the solution is reached in 11th iteration.

Find the norm of residual A.x - b for the 10th iteration of x, and it's accuracy (distance between x(9) and x(10)).

```
r = A.x[10] - b
```

$\{-\frac{9}{2097152}, -\frac{9}{8388608}, -\frac{9}{8388608}, 0\}$

```
nrGS = Sqrt[r.r]
```

$\frac{27}{4194304\sqrt{2}}$

```
nrGS // N
```

4.55186×10^{-6}

```
d = x[10] - x[9]
```

$\{\frac{9}{2097152}, \frac{9}{4194304}, \frac{9}{4194304}, \frac{9}{8388608}\}$

```
ndGS = Sqrt[d.d]
```

$\frac{45}{8388608}$

```
ndGS // N
```

5.36442×10^{-6}

In Gauss - Seidel method, the norm of residual of the 10^{th} iteration achieves 4.55186×10^{-6} and the accuracy 5.36442×10^{-6} .

Compare both methods in the norms of residuals A.x - b for the 10th iteration of x.

```
nrGS // N
```

4.55186×10^{-6}

```
nrJ // N
```

0.00195313

The Gauss -Seidel residual 4.55186×10^{-6} is more less than the Jacobi one 0.00195313.

■ Loop with break condition: norm $||x^{k+1}-x^k|| < \delta$

```
delta = 10^-4;
Clear[x]
x[0] = {0, 0, 0, 0};
it = 20;
Do[
 x[k + 1] = H .x[k] + g;
 Print[k + 1, ". iteration: x=", x[k + 1] // N];
 d = x[k + 1] - x[k];
 If[Sqrt[d.d] < delta, Break[]],
 {k, 0, it}]
Print["norm ||x^(k+1)-x^k||= ", Sqrt[d.d] // N]
```

1. iteration: x={0.25, 0.5625, 0.0625, 0.40625}

2. iteration: x={0.40625, 0.703125, 0.203125, 0.476563}

3. iteration: x={0.476563, 0.738281, 0.238281, 0.494141}

4. iteration: x={0.494141, 0.74707, 0.24707, 0.498535}

5. iteration: x={0.498535, 0.749268, 0.249268, 0.499634}

6. iteration: x={0.499634, 0.749817, 0.249817, 0.499908}

7. iteration: x={0.499908, 0.749954, 0.249954, 0.499977}

8. iteration: x={0.499977, 0.749989, 0.249989, 0.499994}

norm $||x^{k+1}-x^k||=$ 0.0000858307

▪ Loop with break condition: residual $||A.x^{k+1}-b|| < \delta$

```
delta = 10^-4;
Clear[x]
x[0] = {0, 0, 0, 0};
it = 20;
Do[
 x[k + 1] = H .x[k] + g;
 Print[k + 1, ". iteration: x=", x[k + 1] // N];
 res = A.x[k + 1] - b;
 If[Sqrt[res.res] < delta, Break[]],
 {k, 0, it}]
Print["residual||A .x^(k+1)-b||= ", Sqrt[res.res] // N]
```

```
1. iteration: x={0.25, 0.5625, 0.0625, 0.40625}
2. iteration: x={0.40625, 0.703125, 0.203125, 0.476563}
3. iteration: x={0.476563, 0.738281, 0.238281, 0.494141}
4. iteration: x={0.494141, 0.74707, 0.24707, 0.498535}
5. iteration: x={0.498535, 0.749268, 0.249268, 0.499634}
6. iteration: x={0.499634, 0.749817, 0.249817, 0.499908}
7. iteration: x={0.499908, 0.749954, 0.249954, 0.499977}
8. iteration: x={0.499977, 0.749989, 0.249989, 0.499994}
```

residual$||A.x^{k+1}-b||=$ 0.0000728298

Try different initial vector


```
delta = 10^-4;
Clear[x]
x[0] = {1, 2, 3, -4};
it = 20;
Do[
 x[k + 1] = H .x[k] + g;
 Print[k + 1, ". iteration: x=", x[k + 1] // N];
 res = A.x[k + 1] - b;
 If[Sqrt[res.res] < delta, Break[]],
 {k, 0, it}]
Print["residual||A .x^(k+1)-b||= ", Sqrt[res.res] // N]
```

```
1. iteration: x={1.5, -0.125, -0.625, 0.0625}
2. iteration: x={0.0625, 0.53125, 0.03125, 0.390625}
3. iteration: x={0.390625, 0.695313, 0.195313, 0.472656}
4. iteration: x={0.472656, 0.736328, 0.236328, 0.493164}
5. iteration: x={0.493164, 0.746582, 0.246582, 0.498291}
6. iteration: x={0.498291, 0.749146, 0.249146, 0.499573}
7. iteration: x={0.499573, 0.749786, 0.249786, 0.499893}
8. iteration: x={0.499893, 0.749947, 0.249947, 0.499973}
9. iteration: x={0.499973, 0.749987, 0.249987, 0.499993}
```

residual$||A.x^{k+1}-b||=$ 0.000084968