Simultaneous Linear Equations

Topic: Gauss-Seidel Method

Major: Industrial Engineering

Gauss-Seidel Method

An *iterative* method.

Basic Procedure:

-Algebraically solve each linear equation for x_i

-Assume an initial guess solution array

-Solve for each x_i and repeat

-Use absolute relative approximate error after each iteration to check if error is within a pre-specified tolerance.

Gauss-Seidel Method

Why?

The Gauss-Seidel Method allows the user to control round-off error.

Elimination methods such as Gaussian Elimination and LU Decomposition are prone to prone to round-off error.

Also: If the physics of the problem are understood, a close initial guess can be made, decreasing the number of iterations needed.

Gauss-Seidel Method

Algorithm

A set of *n* equations and *n* unknowns:

$$a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \ldots + a_{1n}x_n = b_1$$

$$a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + \ldots + a_{2n}x_n = b_2$$

$$\vdots \qquad \vdots$$

$$a_{n1}x_1 + a_{n2}x_2 + a_{n3}x_3 + \ldots + a_{nn}x_n = b_n$$

If: the diagonal elements are non-zero

Rewrite each equation solving for the corresponding unknown

ex:

First equation, solve for x_1

Second equation, solve for x_2

Gauss-Seidel Method

Algorithm

Rewriting each equation

$$x_1 = \frac{c_1 - a_{12}x_2 - a_{13}x_3 \ldots\ldots - a_{1n}x_n}{a_{11}}$$ ← From Equation 1

$$x_2 = \frac{c_2 - a_{21}x_1 - a_{23}x_3 \ldots\ldots - a_{2n}x_n}{a_{22}}$$ ← From equation 2

$$\vdots \qquad \vdots \qquad \vdots$$

$$x_{n-1} = \frac{c_{n-1} - a_{n-1,1}x_1 - a_{n-1,2}x_2 \ldots\ldots - a_{n-1,n-2}x_{n-2} - a_{n-1,n}x_n}{a_{n-1,n-1}}$$ ← From equation n-1

$$x_n = \frac{c_n - a_{n1}x_1 - a_{n2}x_2 - \ldots\ldots - a_{n,n-1}x_{n-1}}{a_{nn}}$$ ← From equation n

Gauss-Seidel Method

Algorithm

General Form of each equation

$$x_1 = \frac{c_1 - \sum\limits_{\substack{j=1 \\ j \neq 1}}^{n} a_{1j} x_j}{a_{11}}$$

$$x_{n-1} = \frac{c_{n-1} - \sum\limits_{\substack{j=1 \\ j \neq n-1}}^{n} a_{n-1,j} x_j}{a_{n-1,n-1}}$$

$$x_2 = \frac{c_2 - \sum\limits_{\substack{j=1 \\ j \neq 2}}^{n} a_{2j} x_j}{a_{22}}$$

$$x_n = \frac{c_n - \sum\limits_{\substack{j=1 \\ j \neq n}}^{n} a_{nj} x_j}{a_{nn}}$$

Gauss-Seidel Method

Algorithm

General Form for any row 'i'

$$x_i = \frac{c_i - \sum_{\substack{j=1 \\ j \neq i}}^{n} a_{ij} x_j}{a_{ii}}, i = 1, 2, \ldots, n.$$

How or where can this equation be used?

Gauss-Seidel Method

Solve for the unknowns

Assume an initial guess for [X]

$$\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{n-1} \\ x_n \end{bmatrix}$$

Use rewritten equations to solve for each value of xi.

Important: Remember to use the most recent value of x_i. Which means to apply values calculated to the calculations remaining in the **current** iteration.

Gauss-Seidel Method

Calculate the Absolute Relative Approximate Error

$$\left|\varepsilon_{a}\right|_{i}=\left|\frac{x_{i}^{\text{new}}-x_{i}^{\text{old}}}{x_{i}^{\text{new}}}\right|\times 100$$

So when has the answer been found?

The iterations are stopped when the absolute relative approximate error is less than a pre-specified tolerance for all unknowns.

Example: Production Optimization

To find the number of toys a company should manufacture per day to optimally use their injection-molding machine and the assembly line, one needs to solve the following set of equations. The unknowns are the number of toys for boys, *x1*, number of toys for girls, *x2*, and the number of unisexual toys, *x3*.

$$\begin{bmatrix} 0.3333 & 0.1667 & 0.6667 \\ 0.1667 & 0.6667 & 0.3333 \\ 1.05 & -1.00 & 0.00 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 756 \\ 1260 \\ 0 \end{bmatrix}$$

Find the values of x_1, x_2, and x_3 using the Gauss Seidel Method

Example: Production Optimization

The system of equations is: $$\begin{bmatrix} 0.3333 & 0.1667 & 0.6667 \\ 0.1667 & 0.6667 & 0.3333 \\ 1.05 & -1.00 & 0.00 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 756 \\ 1260 \\ 0 \end{bmatrix}$$

Initial Guess: Assume an initial guess of $$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1000 \\ 1000 \\ 100 \end{bmatrix}$$

Example: Production Optimization

Rewriting each equation

$$\begin{bmatrix} 0.3333 & 0.1667 & 0.6667 \\ 0.1667 & 0.6667 & 0.3333 \\ 1.05 & -1.00 & 0.00 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 756 \\ 1260 \\ 0 \end{bmatrix}$$

$$x_1 = \frac{756 - 0.1667x_2 - 0.6667x_3}{0.3333}$$

$$x_2 = \frac{1260 - 0.1667x_1 - 0.3333x_3}{0.6667}$$

$$x_3 = \frac{0 - 1.05x_1 - (-1.00)x_2}{0}$$

Example: Production Optimization

$$x_3 = \frac{0 - 1.05x_1 - (-1.00)x_2}{0}$$

The Equation for x_3 is divided by 0 which is undefined. Therefore the order of the equations will need to be reordered. Equation 3 and equation 1 will be switched. By switching equations 3 and 1, the matrix will also become diagonally dominant.

The system of equations becomes:

$$\begin{bmatrix} 1.05 & -1.00 & 0.00 \\ 0.1667 & 0.6667 & 0.3333 \\ 0.3333 & 0.1667 & 0.6667 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 1260 \\ 75 \end{bmatrix}$$

Example: Production Optimization

Rewriting each equation

$$\begin{bmatrix} 1.05 & -1.00 & 0.00 \\ 0.1667 & 0.6667 & 0.3333 \\ 0.3333 & 0.1667 & 0.6667 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 1260 \\ 75 \end{bmatrix}$$

$$x_1 = \frac{0-(-1.00)x_2-(0)x_3}{1.05}$$

$$x_2 = \frac{1260-0.1667x_1-0.3333x_3}{0.6667}$$

$$x_3 = \frac{756-0.3333x_1-0.1667x_2}{0.6667}$$

Example: Production Optimization

Applying the initial guess and solving for a_i

$$x_1 = \frac{0-(-1.00)\times 1000 - 0\times 100}{1.05} = 952.381$$

Initial Guess

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1000 \\ 1000 \\ 100 \end{bmatrix}$$

$$x_2 = \frac{1260 - 0.1667\times 952.381 - 0.3333\times 100}{0.6667} = 1601.782$$

$$x_3 = \frac{756 - 0.3333\times 952.381 - 0.1667\times 1601.782}{0.6667} = 257.3187$$

When solving for x_2, how many of the initial guess values were used?

Example: Production Optimization

Finding the absolute relative approximate error

$$\left|\varepsilon_a\right|_i = \left|\frac{x_i^{new} - x_i^{old}}{x_i^{new}}\right| \times 100$$

$$\left|\varepsilon_a\right|_1 = \left|\frac{952.381 - 1000}{952.381}\right| x100 = 5.0000\,\%$$

$$\left|\varepsilon_a\right|_2 = \left|\frac{1601.782 - 1000}{1601.782}\right| x100 = 37.5695\%$$

$$\left|\varepsilon_a\right|_3 = \left|\frac{257.3187 - 100}{257.3187}\right| x100 = 61.1377\%$$

At the end of the first iteration

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 952.381 \\ 1601.782 \\ 257.3187 \end{bmatrix}$$

The maximum absolute relative approximate error is 61.1377%

Example: Production Optimization

Iteration #2

Using

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 952.381 \\ 1601.782 \\ 257.3187 \end{bmatrix}$$

from iteration #1

the values of x_i are found:

$$x_1 = \frac{0-(-1.00)\times 1601.782 - 0\times 257.3187}{1.05} = 1525.507$$

$$x_2 = \frac{1260 - 0.1667\times 1525.507 - 0.3333\times 257.3187}{0.6667} = 1379.832$$

$$x_3 = \frac{756 - 0.3333\times 1525.507 - 0.1667\times 1379.832}{0.6667} = 26.29471$$

Example: Production Optimization

Finding the absolute relative approximate error

$$\left|\in_a\right|_1 = \left|\frac{1525.507 - 952.381}{1525.507}\right| x100 = 37.5695\,\%$$

$$\left|\in_a\right|_2 = \left|\frac{1379.832 - 1601.782}{1379.832}\right| x100 = 16.0853\,\%$$

$$\left|\in_a\right|_3 = \left|\frac{26.29471 - 257.3187}{26.3187}\right| x100 = 878.5947\,\%$$

At the end of the second iteration

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1525.507 \\ 1379.832 \\ 26.29471 \end{bmatrix}$$

The maximum absolute relative approximate error is 878.5947%

Example: Production Optimization

Repeating more iterations, the following values are obtained

Iteration	a_1	$\left\|\in_a\right\|_1 \%$	a_2	$\left\|\in_a\right\|_2 \%$	a_3	$\left\|\in_a\right\|_3 \%$
1	952.381	5.00000	1601.782	37.56953	257.3187	61.13768
2	1525.507	37.56953	1379.832	16.08533	26.29471	878.5947
3	1314.125	16.08533	1548.18	10.87393	89.87627	70.74343
4	1474.457	10.87393	1476.305	4.868566	27.69402	224.5331
5	1406.004	4.868566	1524.507	3.161815	49.86268	44.45942
6	1451.911	3.161815	1501.946	1.502123	32.55386	53.16978

! Notice – After six iterations, the absolute relative approximate errors are decreasing, but are still high.

Example: Production Optimization

Repeating more iterations, the following values are obtained

Iteration	a_1	$\left\|\in_a\right\|_1 \%$	a_2	$\left\|\in_a\right\|_2 \%$	a_3	$\left\|\in_a\right\|_3 \%$
20	1439.848	0.000643	1511.835	0.00035	36.11219	0.00915
21	1439.842	0.00035	1511.838	0.000193	36.11398	0.004958

The value of

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1439.842 \\ 1511.838 \\ 36.11398 \end{bmatrix}$$

closely approaches the true value of

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1439.844 \\ 1511.836 \\ 36.11334 \end{bmatrix}$$

Gauss-Seidel Method: Potential Pitfall

Diagonally dominant: The coefficient on the diagonal must be at least equal to the sum of the other coefficients in that row and at least one row with a diagonal coefficient greater than the sum of the other coefficients in that row.

Non-Diagonally dominant

$$[A]=\begin{bmatrix} 2 & 5.81 & 34 \\ 45 & 43 & 1 \\ 123 & 16 & 1 \end{bmatrix}$$

Diagonally dominant

$$[B]=\begin{bmatrix} 124 & 34 & 56 \\ 23 & 53 & 5 \\ 96 & 34 & 129 \end{bmatrix}$$

Most physical systems, such as the previous example, result in simultaneous linear equations that have diagonally dominant coefficient matrices

Gauss-Seidel Method

Summary

-Advantages of the Gauss-Seidel Method

-Algorithm for the Gauss-Seidel Method

-Pitfalls of the Gauss-Seidel Method

Gauss-Seidel Method

Questions?