1. Gauss-Seidel Method

Let's revisit Jacobi's method,

$$\begin{aligned}
x_1^{(k+1)} &= \frac{b_1 - \left(a_{12}x_2^{(k)} + \cdots + a_{1n}x_n^{(k)}\right)}{a_{11}},\\
x_2^{(k+1)} &= \frac{b_2 - \left(a_{21}x_1^{(k)} + a_{23}x_3^{(k)} + \cdots + a_{2n}x_n^{(k)}\right)}{a_{22}},\\
&\vdots\\
x_n^{(k+1)} &= \frac{b_n - \left(a_{n1}x_1^{(k)} + \cdots + a_{(n-1)n}x_{n-1}^{(k)}\right)}{a_{nn}}.
\end{aligned}$$

Consider the first component of the iterates, $x_1^{(k+1)}$. It only depends on $x_2^{(k)}, \ldots, x_n^{(k)}$ but not $x_1^{(k)}$. So, we make an independent update $x_1^{(k+1)}$. Then for $x_2^{(k+1)}$, we are still using the **old** value $x_1^{(k)}$, even though we have just computed the latest update $x_1^{(k+1)}$. Gauss-Seidel method utilizes this brilliant realisation. Since the Jacobi method sequentially computes the iterates by each component, we can utilize the immediate information obtained within a single step of the algorithm to help us advance the sequence. In other words, if we know $x_1^{(k+1)}$ is a better candidate to help us determine $x_2^{(k+1)}$ than $x_1^{(k)}$, we should use it by all means. This means $x_3^{(k+1)}$ will use the immediate updates of $x_1^{(k+1)}$ and $x_2^{(k+1)}$ instead of $x_1^{(k)}$ and $x_2^{(k)}$. So on and so forth.

How do we write this out in a more compact form? First, we list the actual equations. Let's consider a 3×3 matrix as an example of a more general approach.

Example. Consider the Gauss-Seidel iterative method for a 3×3 system,

$$\begin{aligned}
a_{11}x_1 + a_{12}x_2 + a_{13}x_3 &= b_1;\\
a_{21}x_1 + a_{22}x_2 + a_{13}x_3 &= b_2;\\
a_{31}x_1 + a_{32}x_2 + a_{33}x_3 &= b_3.
\end{aligned}$$

Gauss-Seidel method surmises

$$\begin{aligned}
x_1^{(k+1)} &= \frac{b_1 - \left(a_{12}x_2^{(k)} + a_{13}x_3^{(k)}\right)}{a_{11}},\\
x_2^{(k+1)} &= \frac{b_2 - \left(\boxed{a_{21}x_1^{(k+1)}} + a_{23}x_2^{(k)}\right)}{a_{22}},\\
x_3^{(k+1)} &= \frac{b_3 - \left(\boxed{a_{31}x_1^{(k+1)}} + \boxed{a_{32}x_2^{(k+1)}}\right)}{a_{33}},
\end{aligned}$$

where we highlight the boxed terms: they are the immediate updates from the previous lines, within the same k^{th} step of the algorithm.

Now, let's put the $(k+1)^{th}$ iterates on one side, and k^{th} ones on the other – that is, move the boxes to the LHS. We have

$$\begin{aligned}
a_{11}x_1^{(k+1)} &= b_1 - \left(a_{12}x_2^{(k)} + a_{13}x_3^{(k)}\right),\\
a_{21}x_1^{(k+1)} + \boxed{a_{22}x_2^{(k+1)}} &= b_2 - a_{23}x_2^{(k)},\\
\boxed{a_{31}x_1^{(k+1)}} + \boxed{a_{32}x_2^{(k+1)}} + a_{33}x_3^{(k+1)} &= b_3.
\end{aligned}$$

How shall we write the LHS as a matrix-vector multiplication? The vector is obviously the update $\boldsymbol{x}^{(k+1)}$. The matrix, in fact, is the lower triangular section of A. We see it this way:

$$\begin{aligned}
a_{11}x_1^{(k+1)} + 0x_2^{(k+1)} + 0x_3^{(k+1)} &= b_1 - \left(0x_1^{(k)} + a_{12}x_2^{(k)} + a_{13}x_3^{(k)}\right),\\
a_{21}x_1^{(k+1)} + a_{22}x_2^{(k+1)} + 0x_3^{(k+1)} &= b_2 - \left(0x_1^{(k)} + 0x_2^{(k)} + a_{23}x_2^{(k)}\right),\\
a_{31}x_1^{(k+1)} + a_{32}x_2^{(k+1)} + a_{33}x_3^{(k+1)} &= b_3 - \left(0x_1^{(k)} + 0x_2^{(k)} + 0x_3^{(k)}\right).
\end{aligned}$$

More concretely,

$$LHS = \begin{bmatrix} a_{11} & 0 & 0 \\ a_{21} & a_{22} & 0 \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} x_1^{(k+1)} \\ x_2^{(k+1)} \\ x_3^{(k+1)} \end{bmatrix} = \begin{bmatrix} a_{11} & 0 & 0 \\ 0 & a_{22} & 0 \\ 0 & 0 & a_{33} \end{bmatrix} \begin{bmatrix} x_1^{(k+1)} \\ x_2^{(k+1)} \\ x_3^{(k+1)} \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ a_{21} & 0 & 0 \\ a_{31} & a_{32} & 0 \end{bmatrix} \begin{bmatrix} x_1^{(k+1)} \\ x_2^{(k+1)} \\ x_3^{(k+1)} \end{bmatrix} = (D+L)\,\boldsymbol{x}^{(k+1)}.$$

In the RHS, we see that the data vector $\boldsymbol{b}$ stays put, while the rest of the coefficients form the upper triangular part of A, namely,

$$RHS = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} - \begin{bmatrix} 0 & a_{12} & a_{13} \\ 0 & 0 & a_{23} \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1^{(k)} \\ x_2^{(k)} \\ x_3^{(k)} \end{bmatrix} = \boldsymbol{b} - U\boldsymbol{x}^{(k)}.$$

In general, the Gauss-Seidel method reads

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

Taking an inverse of $D+L$, we have

$$\text{(Gauss-Seidel)} \quad \boldsymbol{x}^{(k+1)} = -\left(D+L\right)^{-1} U\boldsymbol{x}^{(k)} + \left(D+L\right)^{-1}\boldsymbol{b}.$$

In component form, for $i = 1, 2, \ldots, n$, we have

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

by considering the individual equations. To compute the update for the $x_i^{(k+1)}$, we need to use $\left\{x_j^{(k+1)}\right\}_{j=1}^{i-1}$, that is, the immediate updates within the same step for all previous unknowns $x_1^{(k+1)}, \ldots x_{i-1}^{(k+1)}$, AND the ones we haven't updated, $\left\{x_j^{(k)}\right\}_{j=i+1}^{n}$.

2. Comparison

Compare the Gauss-Seidel method to the Jacobi method,

$$\text{(Gauss-Seidel)} \quad \boldsymbol{x}^{(k+1)} = -\left(D+L\right)^{-1} U\boldsymbol{x}^{(k)} + \left(D+L\right)^{-1}\boldsymbol{b}.$$

$$\text{(Jacobi)} \quad \boldsymbol{x}^{(k+1)} = -D^{-1}\left(L+U\right)\boldsymbol{x}^{(k)} + D^{-1}\boldsymbol{b}.$$

Both methods are in the form of

$$\boldsymbol{x}^{(k+1)} = T\boldsymbol{x}^{(k)} + c$$

where T is a matrix (operator), different for the methods applied, i.e.

$$T_J = -D^{-1}\left(L+U\right),$$
$$T_{GS} = -\left(D+L\right)^{-1}U.$$

To understand how $\boldsymbol{x}^{(k)}$ converges, we look at the sequential error,

$$\begin{aligned} \left\|\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^{(k)}\right\| &= \left\|T\left(\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^{(k)}\right)\right\| \\ &\leq \|T\| \left\|\boldsymbol{x}^{(k)} - \boldsymbol{x}^{(k-1)}\right\|, \end{aligned}$$

which boils it down to studying the matrix norm $\|T\|$. In fact, if $\|T\| < 1$, the sequential error goes to 0.

3. Example of Gauss-Seidel

Let's consider the same system as the one in Jacobi iteration, $A = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$ and $\boldsymbol{b} = \begin{bmatrix} 3 \\ 3 \end{bmatrix}$, with an initial guess $\boldsymbol{x}^{(0)} = (0, 0)$. Gauss-Seidel method says

$$x_1^{(1)} = \frac{b_1 - a_{12}x_2^{(0)}}{a_{11}} = \frac{3 - 1\times 0}{2} = \boxed{\frac{3}{2}},$$

$$x_2^{(1)} = \frac{b_1 - a_{21}\boxed{x_1^{(1)}}}{a_{22}} = \frac{3 - 1\times\boxed{\frac{3}{2}}}{2} = \frac{3}{4}.$$

We have the next iterates

$$x_1^{(2)} = \frac{3 - 1\times\frac{3}{4}}{2} = \boxed{\frac{9}{8}}, \quad x_2^{(2)} = \frac{3 - 1\times\boxed{\frac{9}{8}}}{2} = \frac{15}{16},$$

and

$$x_1^{(3)} = \frac{3 - 1 \times \frac{15}{16}}{2} = \boxed{\frac{33}{32}}, \quad x_2^{(3)} = \frac{3 - 1 \times \boxed{\frac{33}{32}}}{2} = \frac{63}{64}$$

where the boxed terms highlight the immediate usage of a previous update.

4. Convergence of Jacobi Iteration

We are back at the formula

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

Now, we want to see if the error between successive approximations is getting smaller or not. We first compute

$$\begin{aligned} e^{(k+1)} = \left\|\boldsymbol{x}^{(k+1)} - \boldsymbol{x}^{(k)}\right\| &= \left\|T_J\left(\boldsymbol{x}^{(k)} - \boldsymbol{x}^{(k-1)}\right)\right\| \\ &= \left\|-D^{-1}(L+U)\left(\boldsymbol{x}^{(k)} - \boldsymbol{x}^{(k-1)}\right)\right\| \\ &\le \left\|D^{-1}(L+U)\right\| \left\|\left(\boldsymbol{x}^{(k)} - \boldsymbol{x}^{(k-1)}\right)\right\| \\ &= \left\|D^{-1}(L+U)\right\| e^{(k)} \end{aligned}$$

So, we see that if $\left\|D^{-1}(L+U)\right\| < 1$, then the sequential error will decrease (true for any matrix norm $\|\cdot\|$).

Is there anything special about matrices $A = D + L + U$ that satisfies $\left\|D^{-1}(L+U)\right\| < 1$? Let's look at what $D^{-1}(L+U)$ really means. Suppose we are looking at the l^∞-norm. Then,

$$\left\|D^{-1}(L+U)\right\|_\infty = \max_{1\le i\le n} \frac{1}{|a_{ii}|} \sum_{j\neq i}^{n} |a_{ij}|\,.$$

Requiring that $\left\|D^{-1}(L+U)\right\|_\infty < 1$ means that

$$\max_{1\le i\le n} \frac{1}{|a_{ii}|} \sum_{j\neq i}^{n} |a_{ij}| < 1 \implies \sum_{j\neq i}^{n} |a_{ij}| < |a_{ii}|\,, \text{ for each } i = 1, 2, \ldots, n.$$

This is equivalent to saying that the diagonal entry must be very large, at least as large as the absolute sum of the rest of the entries in the same row. This type of matrices is call **strictly diagonally dominant matrix** (if the inequality is $\le$, then it is just **diagonally dominant**). Jacobi method is **guaranteed** to work for these matrices.

You may wonder, what if my matrix is not diagonally dominant, does the Jacobi method still work? The answer is, sometimes. There is in fact a much more subtle condition called Sassenfeld condition that is less stringent than $\left\|D^{-1}(L+U)\right\| < 1$. It is a milder set of sufficient conditions that include a wider range of matrices that will converge under the Jacobi method.