

Lecture # 18
Iterative Methods for Large Linear Systems: Part II

We wish to solve

$$A\mathbf{x} = \mathbf{b}$$

where $A \in \mathbf{R}^{n \times n}$ is nonsingular. We then of n are being VERY LARGE, say, $n = 10^6$ or $n = 10^7$. Usually, the matrix is also sparse (mostly zeros) and Gaussian elimination is not feasible.

To start, let

$$A = D + L + U$$

where

$$D = \text{diag}(a_{11}, a_{22}, \dots, a_{nn})$$

$$L = \begin{pmatrix} 0 & 0 & \cdots & \cdots & \cdots & 0 & \cdots \\ a_{21} & 0 & \cdots & \cdots & \cdots & \cdots & 0 \\ a_{31} & a_{32} & 0 & \cdots & \cdots & \cdots & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{n-1,1} & a_{n-1,2} & \cdots & \cdots & \cdots & 0 & 0 \\ a_{n,1} & a_{n,2} & \cdots & \cdots & \cdots & a_{n,n-1} & 0 \end{pmatrix}$$

$$U = \begin{pmatrix} 0 & a_{12} & a_{13} & \cdots & \cdots & \cdots & a_{1,n} \\ 0 & 0 & a_{23} & \cdots & \cdots & \cdots & a_{2,n} \\ 0 & 0 & 0 & a_{34} & \cdots & \cdots & a_{3,n} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & \cdots & \cdots & 0 & a_{n-1,n} \\ 0 & 0 & \cdots & \cdots & \cdots & 0 & 0 \end{pmatrix}$$

That is, we split A into a diagonal matrix D , a strictly lower triangular matrix L , and a strictly upper triangular matrix U .

To get a good iterative method, we need two things.

1. The sequence $\{\mathbf{x}^{(k)}\}$ is easy to compute.
- 2.

$$\lim_{k \rightarrow \infty} \mathbf{x}^{(k)} = \mathbf{x}$$

Neither the Jacobi nor the Gauss–Seidel iterations converge for all nonsingular matrices A , but they do satisfy the first criterion.

The general form of this iterations comes from writing A as the difference of two matrices M and N , thus

$$A = M - N$$

and we get an iteration

$$M\mathbf{x}^{(k+1)} = N\mathbf{x}^{(k)} + \mathbf{b}. \quad (1)$$

For instance, the Jacobi iteration is

$$M = D, \quad N = -(L + U).$$

The Gauss–Seidel iteration is

$$M = D + L, \quad N = -U.$$

If we let $\mathbf{e}^{(k)} = \mathbf{x} - \mathbf{x}^{(k)}$ then

$$M\mathbf{e}^{(k+1)} = N\mathbf{e}^{(k)}.$$

Assume that M is nonsingular (otherwise we could not solve for $\mathbf{x}^{(k+1)}$ at each step. Then

$$\mathbf{e}^{(k+1)} = M^{-1}N\mathbf{e}^{(k)} \quad (2)$$

Using norms, we have that

$$\|\mathbf{e}^{(k+1)}\| \leq \|M^{-1}N\| \|\mathbf{e}^{(k)}\|.$$

Thus

$$\|\mathbf{e}^{(k)}\| \leq \|M^{-1}N\|^k \|\mathbf{e}^{(0)}\|.$$

so

$$\lim_{k \rightarrow \infty} \|\mathbf{e}^{(k)}\| = 0$$

if $\|M^{-1}N\|$ in any induced matrix norm!

Let $G = M^{-1}N$ be called the iteration matrix, then note that (2) may be written

$$\mathbf{e}^{(k+1)} = G\mathbf{e}^{(k)}$$

which is just the power iteration. Thus the direction of $\mathbf{e}^{(k)}$ should converge to that of the eigenvector associated with the eigenvalue of largest magnitude of G . This intuition leads to a theorem I will not prove—called the *fundamental theorem of stationary iterative methods*.

Theorem 1 *Let*

$$M\mathbf{x}^{(k+1)} = N\mathbf{x}^{(k)} + \mathbf{b}$$

describe an iterative method to solve $A\mathbf{x} = \mathbf{b}$. Then the sequence $\mathbf{x}^{(k)}$ converges to \mathbf{x} as $k \rightarrow \infty$ for any initial guess $\mathbf{x}^{(0)}$ if and only if $\rho(G) < 1$ where

$$\rho(G) = \max_{1 \leq i \leq n} |\lambda_i(G)|$$

and $\lambda_1(G), \dots, \lambda_n(G)$ are the eigenvalues of G . Moreover,

$$\lim_{k \rightarrow \infty} \frac{\|\mathbf{x} - \mathbf{x}^{(k+1)}\|}{\|\mathbf{x} - \mathbf{x}^{(k)}\|} = \rho(G)$$

in any vector norm.

One method that is used to accelerate Jacobi and Gauss–Seidel is the successive overrelaxation (SOR) method. First, we compute a Gauss–Seidel step. For $i = 1, \dots, n$ we let

$$\begin{aligned} x_i^{(k+1/2)} &= \left(-\sum_{j=1}^{i-1} a_{ij}x_j^{(k+1)} - \sum_{j=i+1}^n a_{ij}x_j^{(k)} \right) / a_{ii} \\ x_i^{(k+1)} &= (1 - \omega)x_i^{(k)} + \omega x_i^{(k+1/2)} \end{aligned}$$

The parameter ω is from the interval $(0, 2)$, but is usually chosen from the interval $(1, 2)$. If A is symmetric and positive definite, the iteration converges for all $\omega \in (0, 2)$. For certain matrices that arise in partial differential equations (the set of conditions is a bit complicated), the best choice for ω is

$$\omega_{opt} = \frac{2}{1 + \sqrt{1 - \rho^2(J)}}$$

where $\rho(J)$ is the spectral radius of the Jacobi iteration matrix. In practice, this is estimated by doing about 3 or 4 Jacobi iterations and computing

$$\rho(J) \approx \hat{\rho} = \frac{\|\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\|_2}{\|\mathbf{x}^{(k)} - \mathbf{x}^{(k-1)}\|_2}.$$

The SOR method is significantly faster than Jacobi or Gauss–Seidel. If this class were held forty years ago, we would have spent more time on it. Its

implementation is simple and straightforward, but its success depends upon choosing the parameter ω correctly.

The SSOR method was developed for symmetric positive definite systems. It does two SOR sweeps, a forward one and a backward one. The algorithm is as follows.

for $i = 1, \dots, n$

$$x_i^{(k+1/3)} = \left(-\sum_{j=1}^{i-1} a_{ij}x_j^{(k+1/2)} - \sum_{j=i+1}^n a_{ij}x_j^{(k)} \right) / a_{ii}$$

$$x_i^{(k+1/2)} = (1 - \omega)x_i^{(k)} + \omega x_i^{(k+1/3)}$$

for $i = n, \dots, 1$ backwards

$$x_i^{(k+2/3)} = \left(-\sum_{j=1}^{i-1} a_{ij}x_j^{(k+1/2)} - \sum_{j=i+1}^n a_{ij}x_j^{(k+1)} \right) / a_{ii}$$

$$x_i^{(k+1/2)} = (1 - \omega)x_i^{(k+1/2)} + \omega x_i^{(k+2/3)}$$

We note that A is symmetric, so $U = L^T$.

In matrix form, it is

$$\begin{aligned} (D + \omega L)\mathbf{x}^{(k+1/2)} &= [(1 - \omega)D - \omega L^T]\mathbf{x}^{(k)} + \omega \mathbf{b} \\ (D + \omega L^T)\mathbf{x}^{(k+1)} &= [(1 - \omega)D - \omega L]\mathbf{x}^{(k+1/2)} + \omega \mathbf{b} \end{aligned}$$

The splitting is

$$A = M - N$$

where

$$M = (\omega(2 - \omega))^{-1}(D + \omega L)D^{-1}(D + \omega L^T).$$

The matrix M is symmetric positive definite if A is. The SSOR method is insensitive to the choice of ω , so it is common to choose $\omega = 1$, yielding

$$M = (D + L)D^{-1}(D + L^T), \quad N = LD^{-1}L^T.$$