4. Linear systems of equations

$$
\begin{aligned}
a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \cdots\cdots + a_{1N}x_N &= b_1 \\
a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + \cdots\cdots + a_{2N}x_N &= b_2 \\
&\vdots \\
&\vdots \\
a_{N1}x_1 + a_{N2}x_2 + a_{N3}x_3 + \cdots\cdots + a_{NN}x_N &= b_N
\end{aligned}
$$

In matrix form:

$$
\begin{pmatrix}
a_{11} & a_{12} & a_{13} & \cdots\cdots & a_{1N} \\
a_{21} & a_{22} & a_{23} & \cdots\cdots & a_{2N} \\
 & & \vdots & & \\
 & & \vdots & & \\
a_{N1} & a_{N2} & a_{N3} & \cdots\cdots & a_{NN}
\end{pmatrix}
\begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ \vdots \\ x_N \end{pmatrix}
=
\begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ \vdots \\ b_N \end{pmatrix}
$$

1

Given: matrix A and vector b

Solve: $Ax = b$

- $\det(A) \neq 0$ (A is non-singular, A^{-1} exists)

  $\Rightarrow$ there exists a unique solution $x = A^{-1}b$

- $\det(A) = 0$ (A is singular, A^{-1} does not exist) $\Rightarrow$ $\begin{cases} \infty \text{ solutions} \\ \text{no solution} \end{cases}$

1. Gauss elimination method
2. LU decomposition
3. Jacobi iteration method
4. Gauss-Seidel iteration method
5. Relaxation methods
6. Conjugate gradient methods
7. Preconditioner

2

Definition (vector norm): On a vector space V, a vector norm is a function $\|\bullet\|$ from V to the set of non-negative real numbers that obeys the three postulates:

(i) $\|x\| \geq 0$ for any $x \in V$, and $\|x\| = 0$ if and only if $x = 0$

(ii) $\|\lambda x\| = |\lambda|\|x\|$ for real λ

(iii) $\|x + y\| \leq \|x\| + \|y\|$ for $x, y \in V$

Commonly used norms $\|\bullet\|$ for $V = R^n$:

(i) $L_2 - norm$: $\|x\|_2 \equiv \left(\sum_{i=1}^{N} x_i^2 \right)^{1/2}$

(ii) $L_p - norm$: $\|x\|_p \equiv \left(\sum_{i=1}^{N} |x_i|^p \right)^{1/p}$

(iii) $L_\infty - norm$: $\|x\|_\infty \equiv \underset{1 \leq i \leq N}{Max} |x_i|$

3

Definition (matrix norm): A matrix norm subordinating to a vector norm is defined by $\|A\| \equiv Sup\{\|Ax\| : x \in R^N, \|x\| = 1\}$.

Sup = least upper bound

- $L_\infty - norm$: $\|A\|_\infty = \underset{1 \leq i \leq N}{Max} \sum_{j=1}^{N} |a_{ij}|$

- $L_2 - norm$: $\|A\|_2 = \underset{1 \leq i \leq N}{Max} |\sigma_i|$ where σ_i's are the singular values of A

  (σ^2 are eigenvalues of A^*A; A^* = conjugate transpose of A)

$\|A\|_2 = \underset{1 \leq i \leq N}{Max} |\lambda_i|$ if all eigenvalues λ_i of A are real and positive.

4

Definition (Condition number)

Def $\kappa(A) \equiv \|A\| \bullet \|A^{-1}\|$ is called the condtion number of a matrix A.

- $\kappa(A) \geq 1$
- $\kappa(AB) \leq \kappa(A)\kappa(B)$
- $\kappa(A) = \underset{\|x\|=\|y\|}{Sup} \|Ax\| / \|Ay\|$
- $\kappa_2(A) = \lambda_{max} / \lambda_{min}$ if all eigenvalues λ are real and positive

§ **Well/Ill-posed problem**

Given: $Ax = b$, A^{-1} exists

Suppose that the vector b is perturbed to be $\tilde{b}$,

If $Ax = b$ and $A\tilde{x} = \tilde{b}$, by how much $\tilde{x}$ is diffent from x ?

§ **Well/Ill-posed problem**

Consider $\|x - \tilde{x}\| = \|A^{-1}b - A^{-1}\tilde{b}\| = \|A^{-1}(b - \tilde{b})\|$

$$\leq \|A^{-1}\| \bullet \|b - \tilde{b}\| = \|A^{-1}\| \bullet \|Ax\| \bullet \frac{\|b - \tilde{b}\|}{\|b\|}$$

$$\leq \|A^{-1}\| \bullet \|A\| \bullet \|x\| \bullet \frac{\|b - \tilde{b}\|}{\|b\|}$$

The relative error resulting in solving x due to a disturbed b is

$$\frac{\|x - \tilde{x}\|}{\|x\|} \leq \|A^{-1}\| \bullet \|A\| \frac{\|b - \tilde{b}\|}{\|b\|} = \kappa(A) \frac{\|b - \tilde{b}\|}{\|b\|}$$

Ill-conditioned if $\kappa(A)$ is large (solution x is sensitive to a slight variation of b).

Well-conditioned if $\kappa(A)$ is of small to moderate magnitude.

§ Backward Substitution (Upper Triangular Matrix)

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} & \cdots\cdots & a_{1,N-1} & a_{1N} \\ 0 & a_{22} & a_{23} & \cdots\cdots & a_{2,N-1} & a_{2N} \\ 0 & 0 & a_{33} & \cdots\cdots & a_{3,N-1} & a_{3N} \\ & & & \vdots & & \\ & & & \vdots & & \\ 0 & 0 & \cdots\cdots & a_{N-2,N-2} & a_{N-2,N-1} & a_{N-2,N} \\ 0 & 0 & \cdots\cdots & 0 & a_{N-1,N-1} & a_{N-1,N} \\ 0 & 0 & \cdots\cdots & 0 & 0 & a_{NN} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ \vdots \\ x_{N-2} \\ x_{N-1} \\ x_N \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ \vdots \\ b_{N-2} \\ b_{N-1} \\ b_N \end{pmatrix}$$

$$x_N = b_N / a_{NN}$$

$$a_{N-1,N-1}x_{N-1} + a_{N-1,N}x_N = b_{N-1}$$

$$a_{N-2,N-2}x_{N-2} + a_{N-2,N-1}x_{N-1} + a_{N-2,N}x_N = b_{N-2}$$

$$x_i = \left(b_i - \sum_{j=i+1}^{N} a_{ij}x_j \right) \Big/ a_{ii}$$

§ forward Substitution (Lower Triangular Matrix)

$$\begin{pmatrix} a_{11} & 0 & 0 & \cdots\cdots & 0 & 0 \\ a_{21} & a_{22} & 0 & \cdots\cdots & 0 & 0 \\ a_{31} & a_{32} & a_{33} & \cdots\cdots & 0 & 0 \\ \vdots & & \vdots & & & \\ a_{N-2,1} & a_{N-2,2} & \cdots\cdots & a_{N-2,N-2} & 0 & 0 \\ a_{N-1,1} & a_{N-1,2} & \cdots\cdots & a_{N-1,N-2} & a_{N-1,N-1} & 0 \\ a_{N,1} & a_{N,2} & \cdots\cdots & a_{N,N-2} & a_{N,N-1} & a_{NN} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_{N-2} \\ x_{N-1} \\ x_N \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ b_{N-2} \\ b_{N-1} \\ b_N \end{pmatrix}$$

$$x_1 = b_1 / a_{11}$$

$$a_{21}x_1 + a_{22}x_2 = b_2$$

$$x_i = \left(b_i - \sum_{j=1}^{i-1} a_{ij}x_j \right) \Big/ a_{ii}$$

§ Gaussian Elimination Method

$$E_1: \quad a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \cdots\cdots + a_{1N}x_N = b_1$$
$$E_2: \quad a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + \cdots\cdots + a_{2N}x_N = b_2$$
$$\vdots$$
$$E_i: \quad a_{i1}x_1 + a_{i2}x_2 + a_{i3}x_3 + \cdots\cdots + a_{iN}x_N = b_i$$
$$\vdots$$
$$E_N: \quad a_{N1}x_1 + a_{N2}x_2 + a_{N3}x_3 + \cdots\cdots + a_{NN}x_N = b_N$$

STEP 1: provided $a_{11} \neq 0$, DO $-(a_{i1}/a_{11})E_1 + E_i$ for $i = 2,3,\cdots,N$

$$-(a_{i1}/a_{11})\{\ a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \cdots\cdots + a_{1N}x_N = b_1\ \}$$
$$+\{a_{i1}x_1 + a_{i2}x_2 + a_{i3}x_3 + \cdots\cdots + a_{iN}x_N = b_i\}$$
$$\Rightarrow \quad 0x_1 + a_{i2}^{(2)}x_2 + a_{i3}^{(2)}x_3 + \cdots\cdots + a_{iN}^{(2)}x_N = b_i^{(2)}$$

$$a_{ik}^{(2)} = a_{ik} - (a_{i1}/a_{11})a_{1k}$$
$$b_i^{(2)} = b_i - (a_{i1}/a_{11})b_1$$

Therefore, after the do loop, we have

$$E_1: \quad a_{11}^{(2)}x_1 + a_{12}^{(2)}x_2 + a_{13}^{(2)}x_3 + \cdots\cdots + a_{1N}^{(2)}x_N = b_1^{(2)}$$
$$E_2: \quad 0x_1 + a_{22}^{(2)}x_2 + a_{23}^{(2)}x_3 + \cdots\cdots + a_{2N}^{(2)}x_N = b_2^{(2)}$$
$$\vdots$$
$$E_i: \quad 0x_1 + a_{i2}^{(2)}x_2 + a_{i3}^{(2)}x_3 + \cdots\cdots + a_{iN}^{(2)}x_N = b_i^{(2)}$$
$$\vdots$$
$$E_N: \quad 0x_1 + a_{N2}^{(2)}x_2 + a_{N3}^{(2)}x_3 + \cdots\cdots + a_{NN}^{(2)}x_N = b_N^{(2)}$$

with updated coefficients

$$a_{1k}^{(2)} = a_{1k}\ , \quad k = 1,2,\cdots,N$$
$$a_{ik}^{(2)} = a_{ik} - (a_{i1}/a_{11})a_{1k}\ , \quad i,k = 2,3,\cdots,N$$
$$b_1^{(2)} = b_1$$
$$b_i^{(2)} = b_i - (a_{i1}/a_{11})b_1\ , \quad i = 2,3,\cdots,N$$

STEP 2: provided $a_{22}^{(2)} \neq 0$, DO $-(a_{i2}^{(2)}/a_{22}^{(2)})E_2 + E_i$ for $i = 3,4,\cdots,N$

$$E_1: \quad a_{11}^{(3)}x_1 + a_{12}^{(3)}x_2 + a_{13}^{(3)}x_3 + \cdots\cdots + a_{1N}^{(3)}x_N = b_1^{(3)}$$
$$E_2: \quad 0x_1 + a_{22}^{(3)}x_2 + a_{23}^{(3)}x_3 + \cdots\cdots + a_{2N}^{(3)}x_N = b_2^{(3)}$$
$$E_3: \quad 0x_1 + 0x_2 + a_{33}^{(3)}x_3 + \cdots\cdots + a_{3N}^{(3)}x_N = b_3^{(3)}$$
$$\vdots$$
$$E_i: \quad 0x_1 + 0x_2 + a_{i3}^{(3)}x_3 + \cdots\cdots + a_{iN}^{(3)}x_N = b_i^{(3)}$$
$$\vdots$$
$$E_N: \quad 0x_1 + 0x_2 + a_{N3}^{(3)}x_3 + \cdots\cdots + a_{NN}^{(3)}x_N = b_N^{(3)}$$

$$a_{1k}^{(3)} = a_{1k}^{(2)} = a_{1k}\ , \quad k = 1,2,\cdots,N$$
$$a_{2k}^{(3)} = a_{2k}^{(2)}\ , \quad k = 2,\cdots,N$$
$$a_{ik}^{(3)} = a_{ik}^{(2)} - (a_{i2}^{(2)}/a_{22}^{(2)})a_{2k}^{(2)}\ , \quad i,k = 3,4,\cdots,N$$

$$b_1^{(3)} = b_1^{(2)}$$
$$b_2^{(3)} = b_2^{(2)}$$
$$b_i^{(3)} = b_i^{(2)} - (a_{i2}^{(2)}/a_{22}^{(2)})b_2^{(2)}\ , \quad i = 3,4,\cdots,N$$

STEP i : repeat the process (provided $a_{ii}^{(i)} \neq 0$) until $i = N-1$

$$
\begin{aligned}
E_1: \quad & a_{11}^{(N)}x_1 + a_{12}^{(N)}x_2 + a_{13}^{(N)}x_3 + \quad \cdots\cdots\cdots \quad + a_{1N}^{(N)}x_N = b_1^{(N)} \\
E_2: \quad & 0x_1 + a_{22}^{(N)}x_2 + a_{23}^{(N)}x_3 + \quad \cdots\cdots\cdots \quad + a_{2N}^{(N)}x_N = b_2^{(N)} \\
E_3: \quad & 0x_1 + 0x_2 + a_{33}^{(N)}x_3 + \quad \cdots\cdots\cdots \quad + a_{3N}^{(N)}x_N = b_3^{(N)} \\
& \vdots \\
E_i: \quad & 0x_1 + 0x_2 + 0x_3 + \cdots + a_{ii}^{(N)}x_3 + \cdots\cdots\cdots + a_{iN}^{(N)}x_N = b_i^{(N)} \\
& \vdots \\
E_N: \quad & 0x_1 + 0x_2 + 0x_3 + \quad \cdots\cdots\cdots \quad + 0x_{N-1} + a_{NN}^{(N)}x_N = b_N^{(N)}
\end{aligned}
$$

$Ux = b^{(N)}$

$\Rightarrow$ backward substitution

§ Pivoting --- whenever a_{ii}=0 or is very small

(i) "pivoting" : interchange the ith row with the pth row where p is the smallest integer > i and $a_{pi} \neq 0$

e.g. $i = 3, a_{33} = 0$

$$
\begin{aligned}
E_1: \quad & a_{11}^{(3)}x_1 + a_{12}^{(3)}x_2 + a_{13}^{(3)}x_3 + \cdots\cdots\cdots + a_{1N}^{(3)}x_N = b_1^{(3)} \\
E_2: \quad & 0x_1 + a_{22}^{(3)}x_2 + a_{23}^{(3)}x_3 + \cdots\cdots\cdots + a_{2N}^{(3)}x_N = b_2^{(3)} \\
E_3: \quad & 0x_1 + 0x_2 + 0x_3 + \cdots\cdots\cdots + a_{3N}^{(3)}x_N = b_3^{(3)} \\
& \vdots \\
E_i: \quad & 0x_1 + 0x_2 + a_{i3}^{(3)}x_3 + \cdots\cdots\cdots + a_{iN}^{(3)}x_N = b_i^{(3)} \\
& \vdots \\
E_N: \quad & 0x_1 + 0x_2 + a_{N3}^{(3)}x_3 + \cdots\cdots\cdots + a_{NN}^{(3)}x_N = b_N^{(3)}
\end{aligned}
$$

(ii) "partial pivoting" : choose p in such a way that

$$p \geq i \quad \text{and} \quad |a_{pi}| = \max_{i \leq j \leq N} |a_{ji}|$$

(iii) "scaled pivoting": choose p by comparing the relative magnitudes of a_{ji} instead of the absolute magnitude, i.e. $p \geq i$ and

$$S_m \equiv \max_{i \leq j \leq N} |a_{mj}| \quad , \quad \frac{|a_{pi}|}{S_p} = \max_{i \leq j \leq N} \frac{|a_{ji}|}{S_j}$$

Example

$$
\begin{aligned}
& 0x_1 + 0x_2 + 891.2x_3 + 211x_4 + 349.7x_5 + \cdots\cdots\cdots + 1086.1x_N = 415.8 \\
& 0x_1 + 0x_2 + 8.9x_3 + 1.4x_4 + 10.2x_5 + \cdots\cdots\cdots + 3.7x_N = 0.9815
\end{aligned}
$$

Absolute values: 891.2 > 8.9
Relative values 891.2/1086.1 ≈ 0.82 < 8.9/10.2 ≈ 0.87

- "scaling" is used for comparison but not really computed.

(iii) "maximal pivoting": pivoting not only in rows but also in columns at ith step: search for p, q such that p, $q \geq i$ and

$$|a_{pq}| = \max_{i \leq j,k \leq N} |a_{jk}|$$

$\Rightarrow$ interchange the i^{th} row with the p^{th} row

interchange the i^{th} column with the p^{th} column

Example: $i = 3$

E_3: $0x_1 + 0x_2 + 3.3x_3 + 4.8x_4 + 7.17x_5 + \cdots\cdots + 6.8x_N = 5.38$

E_4: $0x_1 + 0x_2 + 8.9x_3 + 1.4x_4 + 10.2x_5 + \cdots\cdots + 3.7x_N = 0.9815$

E_5: $0x_1 + 0x_2 + 4.1x_3 + 4.3x_4 - 3.2x_5 + \cdots\cdots + 0.7x_N = 6.891$

Choose $p = 4$ and $q = 5$

$E_3' = E_4$: $0x_1 + 0x_2 + 10.2x_5 + 1.4x_4 + 8.9x_3 + \cdots\cdots + 3.7x_N = 0.9815$

$E_4' = E_3$: $0x_1 + 0x_2 + 7.17x_5 + 4.8x_4 + 3.3x_3 + \cdots\cdots + 6.8x_N = 5.38$

$E_5' = E_5$: $0x_1 + 0x_2 - 3.2x_5 + 4.3x_4 + 4.1x_3 + \cdots\cdots + 0.7x_N = 6.891$

- Notice x_q interchanges with x_i as well

§ Matrix factorization (LU decomposition)

Given: $Ax = b$, most often $Ax(t) = b(t)$ but $A \neq A(t)$

PROPERTY: A has a diagonal structure
(most nonzero elements are on the main diagonal.)

FIND: ***A = LU***, where $L(U)$ is a lower (upper) triangular matrix

$$L = \begin{pmatrix} \ell_{11} & 0 & 0 & \cdots\cdots & 0 & 0 \\ \ell_{21} & \ell_{22} & 0 & \cdots\cdots & 0 & 0 \\ \ell_{31} & \ell_{32} & \ell_{33} & \cdots\cdots & 0 & 0 \\ & & & \vdots & & \\ \ell_{N-1,1} & \ell_{N-1,2} & \ell_{N-1,3} & \cdots\cdots & \ell_{N-1,N-1} & 0 \\ \ell_{N,1} & \ell_{N,2} & \ell_{N,3} & \cdots\cdots & \ell_{N,N-1} & \ell_{NN} \end{pmatrix} \quad U = \begin{pmatrix} u_{11} & u_{12} & u_{13} & \cdots\cdots & u_{1,N-1} & u_{1N} \\ 0 & u_{22} & u_{23} & \cdots\cdots & u_{2,N-1} & u_{2N} \\ 0 & 0 & u_{33} & \cdots\cdots & u_{3,N-1} & u_{3N} \\ & & & \vdots & & \\ 0 & 0 & 0 & \cdots\cdots & u_{N-1,N-1} & u_{N-1,N} \\ 0 & 0 & 0 & \cdots\cdots & 0 & u_{NN} \end{pmatrix}$$

If so, we can solve $Ax=b$ by backward/forward substitution

- $Ax = LUx = L(UX) \equiv Ly = b$

(i) forward substitution

$Ly = b$

$y_1 = b_1 / \ell_{11}$

$$y_k = \left(b_k - \sum_{j=1}^{k-1} \ell_{k,j} y_j \right) \Big/ \ell_{kk}$$

(ii) backward substitution

$Ux = y$

$x_n = y_n / u_{N,N}$

$$x_k = \left(y_k - \sum_{j=k+1}^{N} u_{k,j} x_j \right) \Big/ u_{kk}$$

§ General LU-decomposition

$$\begin{pmatrix} \ell_{11} & 0 & 0 & \cdots\cdots & 0 & 0 \\ \ell_{21} & \ell_{22} & 0 & \cdots\cdots & 0 & 0 \\ \ell_{31} & \ell_{32} & \ell_{33} & \cdots\cdots & 0 & 0 \\ & & & \vdots & & \\ \ell_{N-1,1} & \ell_{N-1,2} & \ell_{N-1,3} & \cdots\cdots & \ell_{N-1,N-1} & 0 \\ \ell_{N,1} & \ell_{N,2} & \ell_{N,3} & \cdots\cdots & \ell_{N,N-1} & \ell_{NN} \end{pmatrix} \begin{pmatrix} u_{11} & u_{12} & u_{13} & \cdots\cdots & u_{1,N-1} & u_{1N} \\ 0 & u_{22} & u_{23} & \cdots\cdots & u_{2,N-1} & u_{2N} \\ 0 & 0 & u_{33} & \cdots\cdots & u_{3,N-1} & u_{3N} \\ & & & \vdots & & \\ 0 & 0 & 0 & \cdots\cdots & u_{N-1,N-1} & u_{N-1,N} \\ 0 & 0 & 0 & \cdots\cdots & 0 & u_{NN} \end{pmatrix} = A$$

# of constraints = N^2

degrees of freedom = N^2+N

Example: assume ℓ_{ii} assigned for $i = 1, 2, \cdots, N$

By observation, $\ell_{11} u_{11} = a_{11}$

$\ell_{11} u_{1k} = a_{1k}$ for $k = 1, 2, \cdots, N$

§ General LU-decomposition

$$\begin{pmatrix} \ell_{11} & 0 & 0 & \cdots & 0 & 0 \\ \ell_{21} & \ell_{22} & 0 & \cdots & 0 & 0 \\ \ell_{31} & \ell_{32} & \ell_{33} & \cdots & 0 & 0 \\ \vdots & & & & & \\ \ell_{N-1,1} & \ell_{N-1,2} & \ell_{N-1,3} & \cdots & \ell_{N-1,N-1} & 0 \\ \ell_{N,1} & \ell_{N,2} & \ell_{N,3} & \cdots & \ell_{N,N-1} & \ell_{NN} \end{pmatrix} \begin{pmatrix} u_{11} & u_{12} & u_{13} & \cdots & u_{1,N-1} & u_{1N} \\ 0 & u_{22} & u_{23} & \cdots & u_{2,N-1} & u_{2N} \\ 0 & 0 & u_{33} & \cdots & u_{3,N-1} & u_{3N} \\ & & & & \vdots & \\ 0 & 0 & 0 & \cdots & u_{N-1,N-1} & u_{N-1,N} \\ 0 & 0 & 0 & \cdots & 0 & u_{NN} \end{pmatrix} = A$$

$$\ell_{21}u_{11} = a_{21}$$

$$\ell_{21}u_{1k} + \ell_{22}u_{2k} = a_{1k} \quad \text{for } k = 2,3,\cdots,N$$

$$\ell_{31}u_{11} = a_{31}$$

$$\ell_{31}u_{12} + \ell_{32}u_{22} = a_{32}$$

$$\ell_{31}u_{1k} + \ell_{32}u_{2k} + \ell_{33}u_{3k} = a_{3k} \quad \text{for } k = 3,4,\cdots,N$$

$$\ell_{i1}u_{11} = a_{i1}$$

$$\ell_{i1}u_{12} + \ell_{i2}u_{22} = a_{i2}$$

$$\ell_{i1}u_{13} + \ell_{i2}u_{23} + \ell_{i3}u_{33} = a_{i3}$$

$$\cdots$$

$$\ell_{i1}u_{1,i-1} + \ell_{i2}u_{2,i-1} + \cdots + \ell_{i,i-2}u_{i-2,i-1} + \ell_{i,i-1}u_{i-1,i-1} = a_{i,i-1}$$

$$\ell_{i1}u_{1k} + \ell_{i2}u_{2k} + \cdots + \ell_{i,i-1}u_{i-1,k} + \ell_{ii}u_{ik} = a_{ik} \quad \text{for } k = i, i+1, \cdots, N$$

$$\sum_{m=1}^{k} \ell_{im}u_{mk} = a_{ik} \quad , \; k = 1,2,\cdots,i-1$$

$$\sum_{m=1}^{i} \ell_{im}u_{mk} = a_{ik} \quad , \; k = i, i+1, \cdots, N$$

- **special cases:**

$$\begin{cases} \ell_{ii} = 1 & : \text{ Doolittle's factorization} \\ u_{ii} = 1 & : \text{ Crout's factorization} \\ U = L^T & : \text{ Cholesky's factorization} \begin{pmatrix} \text{possible only if } A \text{ is} \\ \text{real,symmetric, and positive-definte} \end{pmatrix} \end{cases}$$

- A matrix A is positive-definite if $x^TAx > 0$ for any $x \neq 0$.

Theorem:
If Gaussian elimination can be performed on the linear system $Ax=b$ without row interchanges, then the matrix A can be factored into $A=LU$.

§ Gaussian elimination procedures

STEP 1: provided $a_{11} \neq 0$, DO $-(a_{i1}/a_{11})E_1 + E_i$ for $i = 2,3,\cdots,N$

$$m_{i1} = a_{i1}/a_{11}$$

$$M^{(1)} = \begin{pmatrix} 1 & 0 & 0 & & \cdots & & 0 \\ -m_{21} & 1 & 0 & 0 & & \cdots & 0 \\ -m_{31} & 0 & 1 & 0 & 0 & \cdots & 0 \\ & & & \vdots & & & \\ -m_{N-1,1} & 0 & 0 & 0 & 0 & \cdots & 1 & 0 \\ -m_{N,1} & 0 & 0 & 0 & 0 & \cdots & 0 & 1 \end{pmatrix} = L^{(1)}$$

It can be shown that

$$M^{(1)}Ax = M^{(1)}b$$

$$A^{(2)}x = b^{(2)}$$

§ Gaussian elimination procedures

STEP 2: provided $a_{22}^{(2)} \neq 0$, DO $-\left(a_{i2}^{(2)} / a_{22}^{(2)}\right) E_2 + E_i$ for $i = 3, 4, \cdots, N$

$m_{i2} = a_{i2}^{(2)} / a_{22}^{(2)}$

$$M^{(2)} = \begin{pmatrix} 1 & 0 & 0 & & & \cdots & 0 \\ 0 & 1 & 0 & 0 & & \cdots & 0 \\ 0 & -m_{32} & 1 & 0 & 0 & \cdots & 0 \\ 0 & -m_{42} & 0 & 1 & 0 & 0 \cdots & 0 \\ & & & \vdots & & & \\ 0 & -m_{N-1,2} & 0 & 0 & 0 & \cdots 1 & 0 \\ 0 & -m_{N,2} & 0 & 0 & 0 & 0 \cdots 0 & 1 \end{pmatrix} = L^{(2)}$$

It can be shown that

$$M^{(2)} A^{(2)} x = M^{(2)} b^{(2)}$$

$$A^{(3)} x = b^{(3)}$$

§ Gaussian elimination procedures

STEP $N-1$: provided $a_{N-1,N-1}^{(N-1)} \neq 0$, DO $-\left(a_{N-1,N}^{(N-1)} / a_{N-1,N-1}^{(N-1)}\right) E_{N-1} + E_N$

$$A^{(N)} = \begin{pmatrix} a_{11}^{(N)} & a_{12}^{(N)} & a_{13}^{(N)} & \cdots\cdots & a_{1,N-1}^{(N)} & a_{1,N}^{(N)} \\ 0 & a_{22}^{(N)} & a_{23}^{(N)} & \cdots\cdots & a_{2,N-1}^{(N)} & a_{2,N}^{(N)} \\ 0 & 0 & a_{33}^{(N)} & \cdots\cdots & a_{3,N-1}^{(N)} & a_{3,N}^{(N)} \\ & & & \vdots & & \\ 0 & 0 & 0 & \cdots\cdots & a_{N-1,N-1}^{(N)} & a_{N-1,N}^{(N)} \\ 0 & 0 & 0 & \cdots\cdots & 0 & a_{N,N}^{(N)} \end{pmatrix} = U$$

$$A^{(N)} = M^{(N-1)} A^{(N-1)} = M^{(N-1)} M^{(N-2)} A^{(N-2)} = \cdots = M^{(N-1)} M^{(N-2)} \cdots M^{(2)} M^{(1)} A$$

$$A = \left(M^{(N-1)} M^{(N-2)} \cdots M^{(2)} M^{(1)}\right)^{-1} U$$

$$A = \left(\left(M^{(1)}\right)^{-1} \left(M^{(2)}\right)^{-1} \cdots \left(M^{(N-2)}\right)^{-1} \left(M^{(N-1)}\right)^{-1}\right) U$$

$$M^{(1)} = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ -m_{21} & 1 & 0 & \cdots & 0 \\ -m_{31} & 0 & 1 & 0 \cdots & 0 \\ & & \vdots & & \\ -m_{N-1,1} & 0 & 0 & 0 \cdots 1 & 0 \\ -m_{N,1} & 0 & 0 & 0 \cdots 0 & 1 \end{pmatrix} \quad M^{(2)} = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & -m_{32} & 1 & 0 \cdots & 0 \\ 0 & -m_{42} & 0 & 1 \cdots & 0 \\ & & \vdots & & \\ 0 & -m_{N-1,2} & 0 & 0 \cdots 1 & 0 \\ 0 & -m_{N,2} & 0 & 0 \cdots 0 & 1 \end{pmatrix}$$

$$\left(M^{(1)}\right)^{-1} = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ m_{21} & 1 & 0 & \cdots & 0 \\ m_{31} & 0 & 1 & 0 \cdots & 0 \\ & & \vdots & & \\ m_{N-1,1} & 0 & 0 & 0 \cdots 1 & 0 \\ m_{N,1} & 0 & 0 & 0 \cdots 0 & 1 \end{pmatrix} \quad \left(M^{(2)}\right)^{-1} = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & m_{32} & 1 & 0 \cdots & 0 \\ 0 & m_{42} & 0 & 1 \cdots & 0 \\ & & \vdots & & \\ 0 & m_{N-1,2} & 0 & 0 \cdots 1 & 0 \\ 0 & m_{N,2} & 0 & 0 \cdots 0 & 1 \end{pmatrix}$$

$$\left(M^{(1)}\right)^{-1} \left(M^{(2)}\right)^{-1} \cdots \left(M^{(N-2)}\right)^{-1} \left(M^{(N-1)}\right)^{-1} =$$

$$\begin{pmatrix} 1 & 0 & 0 & \cdots\cdots & 0 & 0 \\ m_{21} & 1 & 0 & \cdots\cdots & 0 & 0 \\ m_{31} & m_{32} & 1 & \cdots\cdots & 0 & 0 \\ & & & \vdots & & \\ m_{N-1,1} & m_{N-1,2} & m_{N-1,3} & \cdots\cdots & 1 & 0 \\ m_{N,1} & m_{N,2} & m_{N,3} & \cdots\cdots & m_{N,N-1} & 1 \end{pmatrix} = L$$

$$L = \left(M^{(1)}\right)^{-1} \left(M^{(2)}\right)^{-1} \cdots \left(M^{(N-2)}\right)^{-1} \left(M^{(N-1)}\right)^{-1}$$

$$A = \left(M^{(N-1)} M^{(N-2)} \cdots M^{(2)} M^{(1)}\right)^{-1} U$$

$$A = LU$$

§ Iterative technique ~ good for large, sparse matrices

Given: $Ax = b$

STEP1: take a guess for the solution, say $x^{(k)}$

STEP2: compute the error (residue) $r^{(k)} \equiv b - Ax^{(k)}$

STEP3: $r^{(k)}$ too large? $\Rightarrow$ create a new guess $x^{(k+1)}$

i.e. generate a sequence $\{x^{(k)}\}$ in some way

so that the sequence converges to the real solution x

- criterion for stopping the iteration: (i) $\|r^{(k)}\| < \varepsilon$

(ii) $\dfrac{\|x^{(k+1)} - x^{(k)}\|}{\|x^{(k)}\|} < \varepsilon$

(iii) $\dfrac{\|Ax^{(k)} - b\|}{\|b\|} < \varepsilon$

- best situation: $\|r^{(k)}\|_\infty \to 0$ as $k \to \infty$

i.e. $r_i^{(k)} \to 0$ as $k \to \infty$ for i=1, 2,$\cdots$, N.

- iterative schemes: (usually) Given some matrix T and vector c,

$$x^{(k+1)} = Tx^{(k)} + c$$

Theorem

For any $x^{(0)} \in R^n$, the sequence $\{x^{(k)}\}_{k=0}^{\infty}$ defined by $x^{(k+1)} = Tx^{(k)} + c$

for each $k \geq 1$ and $c \neq 0$ converges to the unique solution of $x = Tx + c$

if and only if $\rho(T) < 1$. Moreover, $\|x^{(k)} - x\| \approx \rho(T)^k \|x^{(0)} - x\|$

- The spectral radius $\rho(T)$ of a matrix T is defined as the maximum absolute value of eigenvalues of the matrix T, i.e. Max $|\lambda|$.
- A scheme constructs the matrix T and the vector c in such a way that the solution of x=Tx+c is also the solution of Ax=b.

$x^{(k+1)} = Tx^{(k)} + c \qquad \rho(T) < 1 \Rightarrow \|x^{(k)} - x\| \approx \rho(T)^k \|x^{(0)} - x\|$

< show>

$$\begin{aligned} x^{(k+1)} &= Tx^{(k-1)} + c \\ &= T\left(Tx^{(k-2)} + c\right) + c = T^2 x^{(k-2)} + Tc + c \\ &= T^2\left(Tx^{(k-3)} + c\right) + Tc + c = T^3 x^{(k-3)} + T^2 c + Tc + c \\ &= \cdots \\ &= T^k x^{(0)} + \sum_{m=0}^{k-1} T^m c \end{aligned}$$

- $\|T^k x^{(0)}\| \leq \|T\|^k \|x^{(0)}\| \to 0 \cdot \|x^{(0)}\| = 0$ as $k \to \infty$ if $\rho(T) < 1$
- $\sum_{m=0}^{\infty} T^m c \to (1-T)^{-1} c$ if $\rho(T) < 1$

Therefore $x^{(k)} \to (1-T)c$ as $k \to \infty$

i.e. $(1-T)x = c$ or $x = Tx + c$

§ Error Analysis

Let x^* be the exact solution and $e^{(k)} \equiv x^{(k)} - x^*$.

$$e^{(k+1)} = x^{(k+1)} - x^* = \left(Tx^{(k)} + c\right) - \left(Tx^* + c\right) = Te^{(k)}$$

$$e^{(k)} = T^k e^{(0)}$$

Therefore, $\left(\dfrac{\|e^k\|}{\|e^0\|}\right)^{1/k} \leq \|T^k\|^{1/k} \to \rho(T)$

To reduce error by one order, i.e $\left(\dfrac{\|e^k\|}{\|e^0\|}\right)^{1/k} \leq \left\|\dfrac{1}{10}\right\|^{1/k} \approx \rho(T)$

Thus the iteration number is required by $k \geq -\dfrac{1}{\log_{10}(\rho(T))}$

- The smaller ρ, the faster the convergence is or the better the scheme is. But it cannot be pre-known for a given problem $Ax = b$ and a given scheme.

§ Jacobi iteration method

from E_i : $a_{i1}x_1 + a_{i2}x_2 + a_{i3}x_3 + \cdots\cdots + a_{iN}x_N = b_i$

provided $a_{ii} \neq 0$ (if a_{ii}=0, do reordering like pivoting)

$$a_{ii}x_i + \sum_{j=1}^{i-1} a_{ij}x_j + \sum_{j=i+1}^{N} a_{ij}x_j = b_i$$

$$x_i = \frac{1}{a_{ii}}\left(b_i - \sum_{j=1}^{i-1} a_{ij}x_j - \sum_{j=i+1}^{N} a_{ij}x_j \right), \quad i = 1,2,\cdots,N$$

Write $A = D + L + U$

where D, L, U are the diagonal, lower triangular, and upper triangular parts of A.

$$x = D^{-1}(b - Lx - Ux) = D^{-1}b - D^{-1}(L+U)x$$

§ Jacobi iteration method

DO i = 1,2,…, N (forward loop)

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

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

$$T_{Jacobi} = -D^{-1}(L+U)$$

$$c_{Jacobi} = D^{-1}b$$

Given different matrices $A = D + L + U$ $\Rightarrow$ different T $\Rightarrow$ different ρ

§ Gauss-Seidel iteration method

Jacobi method: DO i = 1,2,…, N (forward loop)

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

Consequently, $x_1^{(k+1)}, x_2^{(k+1)}, \cdots, x_{i-1}^{(k+1)}$ are computed before $x_i^{(k+1)}$.

Gauss-Seidel :

when computing $x_i^{(k+1)}$, replace $x_j^{(k)}$ by $x_j^{(k+1)}$ for $j = 1,2,\cdots,i-1$.

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

$$x^{(k+1)} = D^{-1}\left(b - Lx^{(k+1)} - Ux^{(k)}\right)$$

$$Dx^{(k+1)} = b - Lx^{(k+1)} - Ux^{(k)}$$

$$(D+L)x^{(k+1)} = b - Ux^{(k)}$$

$$x^{(k+1)} = (D+L)^{-1}b - (D+L)^{-1}Ux^{(k)}$$

$$T_{Gauss-Seidel} = -(D+L)^{-1}U$$

$$c_{Gauss-Seidel} = (D+L)^{-1}b$$

Theorem: If A is strictly diagonally dominant, then for any choice of $x^{(0)}$, both the Jacobi and Gauss-Seidel methods give sequences $\{x^{(k)}\}_{k=0}^{\infty}$ that converges to the unique solution of $Ax = b$.

§ Alternative form of Gauss-Seidel iteration method

Define $x^{(k,i)} \equiv \left\{ \begin{array}{l} \text{the resulting vector after the } (i-1)^{th} \text{ step and} \\ \text{before the } i^{th} \text{ step during the } k^{th} \text{ iteration} \end{array} \right\}$

Define $x^{(k,i)} \equiv \left(x_1^{(k)}, x_2^{(k)}, \cdots, x_{i-1}^{(k)}, x_i^{(k-1)}, x_{i+1}^{(k-1)}, \cdots, x_N^{(k-1)} \right)^T$

going to do : $x_i^{(k)} = \frac{1}{a_{ii}} \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k)} - \sum_{j=i+1}^{N} a_{ij} x_j^{(k-1)} \right)$

its residual vector : $r^{(k,i)} = b - Ax^{(k,i)}$

m^{th} component : $r_m^{(k,i)} = b_m - \sum_{j=1}^{N} a_{mj} x_j^{(k,i)} = b_m - \sum_{j=1}^{i-1} a_{mj} x_j^{(k)} - \sum_{j=i}^{N} a_{mj} x_j^{(k-1)}$

$= b_m - \sum_{j=1}^{i-1} a_{mj} x_j^{(k)} - \sum_{j=i+1}^{N} a_{mj} x_j^{(k-1)} - a_{mi} x_i^{(k-1)}$

m^{th} component : $r_m^{(k,i)} = b_m - \sum_{j=1}^{i-1} a_{mj} x_j^{(k)} - \sum_{j=i+1}^{N} a_{mj} x_j^{(k-1)} - a_{mi} x_i^{(k-1)}$

In particular, we are interested in the i^{th} component,

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

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

therefore $x_i^{(k)} = x_i^{(k-1)} + \frac{r_i^{(k,i)}}{a_{ii}}$

~ Gauss-Seidel iteration method ~

Before updating the i^{th} component of x :

m^{th} component : $r_m^{(k,i)} = b_m - \sum_{j=1}^{i-1} a_{mj} x_j^{(k)} - \sum_{j=i+1}^{N} a_{mj} x_j^{(k-1)} - a_{mi} x_i^{(k-1)}$

After updating the i^{th} component of x :

m^{th} component : $r_m^{(k,i+1)} = b_m - \sum_{j=1}^{i-1} a_{mj} x_j^{(k)} - \sum_{j=i+1}^{N} a_{mj} x_j^{(k-1)} - a_{mi} x_i^{(k)}$

In particular, we are interested in the i^{th} component,

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

$$= a_{ii} x_i^{(k)} - a_{ii} x_i^{(k)} = 0$$

- Gauss-Seidel method enforces $r_i^{(k,i+1)} = 0$ to obtain $x_i^{(k)}$ and in such a way, expects to have a convergence.

§ Relaxation method

Whether an iteration method converges or not is problem-dependent.

example: Gauss-Seidel method $x_i^{(k)} = x_i^{(k-1)} + \frac{r_i^{(k,i)}}{a_{ii}}$

- diverge $\Rightarrow$ each time correct too much?
- converge but slowly $\Rightarrow$ correct too little?

Can we adjust the amount of correction at each time?

With a certain positive value of ω, one can perform

$$x_i^{(k)} = x_i^{(k-1)} + \omega \cdot \frac{r_i^{(k,i)}}{a_{ii}}$$

- $0 < \omega < 1$: under-relaxation ($\Leftarrow$ diverge)
- $\omega > 1$: over-relaxation ($\Leftarrow$ converge slowly)
- $\omega = 1$: Gauss-Seidel method

$$x_i^{(k)} = x_i^{(k-1)} + \omega \cdot \frac{r_i^{(k,i)}}{a_{ii}}$$

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

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

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

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

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

$$T_{relaxation} = (D + \omega L)^{-1}\left\{ (1-\omega) D - \omega U \right\}$$

$$c_{relaxation} = \omega (D + \omega L)^{-1} b$$

41

§ Minimization Problem

Given: $Ax = b$, where A is symmetric and positive-definite
(thus all eigenvalues of A are real and positive)

define $\phi : R^N \to R$, $\phi(x) \equiv \frac{1}{2} x^T A x - x^T b$ for all $x \in R^N$

$\Rightarrow$ The solution of $Ax = b$ minimizes the function $\phi(x)$.

<show> $\phi(x) \equiv \frac{1}{2} x^T A x - x^T b = \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} x_i a_{ij} x_j - \sum_{j=1}^{N} b_j x_j$

look for the minimum: $\frac{\partial \phi}{\partial x_k} = 0 = \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \left(\frac{\partial x_i}{\partial x_k} a_{ij} x_j + x_i a_{ij} \frac{\partial x_j}{\partial x_k} \right) - \sum_{j=1}^{N} b_j \frac{\partial x_j}{\partial x_k}$

$$0 = \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \left(\delta_{ik} a_{ij} x_j + x_i a_{ij} \delta_{jk} \right) - \sum_{j=1}^{N} b_j \delta_{jk} \Rightarrow 0 = \frac{1}{2} \left(\sum_{j=1}^{N} a_{kj} x_j + \sum_{i=1}^{N} x_i a_{ik} \right) - b_k$$

$$\Rightarrow \nabla \phi = Ax - b = 0$$

42

§ Gradient iteration method (1D minimization)

Suppose x_k is the current iterate and $p_k \in R^N$ is a given direction vector.

Next iterate: $x_{k+1} = x_k + \alpha\, p_k$ with a value of α which minimizes $\phi(x)$.

$$i.e. \quad \min_{\alpha \in R} \ \phi\left(x^{(k)} + \alpha\, p^{(k)} \right)$$

$$\phi(x_{k+1}) = \frac{1}{2} (x_k + \alpha\, p_k)^T A (x_k + \alpha\, p_k) - (x_k + \alpha\, p_k)^T b$$

$$= \left(\frac{1}{2} x_k^T A x_k - x_k^T b \right) + \alpha p_k^T (A x_k - b) + \frac{1}{2} \alpha^2 p_k^T A p_k$$

Wanted $\frac{\partial \phi(x_{k+1})}{\partial \alpha} = 0 \quad \Rightarrow 0 = -p_k^T r_k + \alpha p_k^T A p_k$

$$\alpha = \frac{p_k^T r_k}{p_k^T A p_k}$$

43

§ Gradient iteration method (1D minimization)

$$x_{k+1} = x_k + \alpha\, p_k$$

$$\alpha = \frac{p_k^T r_k}{p_k^T A p_k}$$

$$\frac{\partial}{\partial \alpha_k} \phi(x_k + \alpha p_k) = 0$$

$$p_k^T \nabla \phi(x_{k+1}) = 0$$

$$LHS = p_k^T (A x_{k+1} - b) = p_k^T A (x_k + \alpha p_k) - p_k^T b$$

$$= p_k^T (A x_k - b) + \alpha p_k^T A p_k = p_k^T (-r_k) + p_k^T r_k = 0$$

44

§ Gradient iteration method (1D minimization)

Step 1: take an initial guess x_0 and take $p_0 = -\nabla\phi(x_0) = -(Ax_0 - b) = r_0$

Step 2: if $\|r_k\| > \varepsilon$, $x_{k+1} = x_k + \alpha_k p_k$

$$p_k = -\nabla\phi(x_k) = r_k = b - Ax_k$$

$$\alpha_k = \frac{p_k^T r_k}{p_k^T A p_k} = \frac{r_k^T r_k}{r_k^T A r_k}$$

- since $\phi(x)$ decreases mostly along the negative gradient direction.

$$x_{k+1} = x_k + \left(\frac{r_k^T r_k}{r_k^T A r_k}\right) r_k$$

$$r_k = b - Ax_k$$

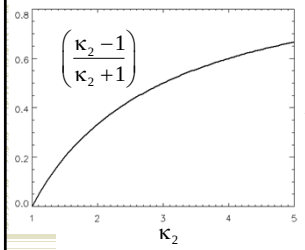
§ Properties of Gradient iteration method

Theorem: For any initial iterate x_0, the sequence $\{x_k\}_{k=0}^{\infty}$ of the gradient method converges to the solution $x^* = A^{-1}b$ and satisfies the error estimates

$$\|x_k - x^*\|_A \le \left(\frac{\kappa_2(A)-1}{\kappa_2(A)+1}\right)^k \|x_0 - x^*\|_A$$

$$\kappa_2(A) = \lambda_{\max}/\lambda_{\min} > 1$$

where the A-norm is defined as $\|x\|_A \equiv \sqrt{x^T A x}$.



- The smaller κ_2, the faster the method converges.

§ Preconditioned Gradient method

Suppose $M \in R^N \times R^N$ is symmetric and positive-define.

Consider $Ax = b$

$$M^{-1/2}Ax = M^{-1/2}b$$

$$\left(M^{-1/2}AM^{-1/2}\right)\left(M^{1/2}x\right) = \left(M^{-1/2}b\right)$$

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

- $\tilde{A}$ is symmetric and positive-definite

Now, instead of solving $Ax = b$, we solve $\tilde{A}\tilde{x} = \tilde{b}$

- The convergence rate is now determined by $\kappa_2(\tilde{A})$

$$\kappa_2(\tilde{A}) = \kappa_2\left(M^{-1/2}AM^{-1/2}\right) = \kappa_2\left(M^{-1/2}M^{-1/2}A\right) = \kappa_2\left(M^{-1}A\right)$$

- An appropriate preconditioner M such that $\kappa_2(M^{-1}A) < \kappa_2(A)$ can accelerate the convergence.

$$\tilde{A}\tilde{x} = \tilde{b} \quad \text{or} \quad \left(M^{-1/2}AM^{-1/2}\right)\left(M^{1/2}x\right) = \left(M^{-1/2}b\right)$$

$$\tilde{x}_{k+1} = \tilde{x}_k + \left(\frac{\tilde{r}_k^T \tilde{r}_k}{\tilde{r}_k^T \tilde{A} \tilde{r}_k}\right)\tilde{r}_k$$

$$\tilde{r}_k = \tilde{b} - \tilde{A}\tilde{x}_k$$

- $\tilde{r}_k = \tilde{b} - \tilde{A}\tilde{x}_k = \left(M^{-1/2}b\right) - \left(M^{-1/2}AM^{-1/2}\right)\left(M^{1/2}x_k\right)$

$$= M^{-1/2}b - M^{-1/2}Ax_k = M^{-1/2}\left(b - Ax_k\right)$$

$$\tilde{r}_k = M^{-1/2}r_k$$

$$\tilde{A}\tilde{x}=\tilde{b} \quad \text{or} \quad \left(M^{-1/2}AM^{-1/2}\right)\left(M^{1/2}x\right)=\left(M^{-1/2}b\right)$$

$$\tilde{x}_{k+1}=\tilde{x}_k+\left(\frac{\tilde{r}_k^T\tilde{r}_k}{\tilde{r}_k^T\tilde{A}\tilde{r}_k}\right)\tilde{r}_k$$

$$\tilde{r}_k=\tilde{b}-\tilde{A}\tilde{x}_k$$

- $\tilde{r}_k=M^{-1/2}r_k$
- $\tilde{\alpha}_k=\dfrac{\tilde{r}_k^T\tilde{r}_k}{\tilde{r}_k^T\tilde{A}\tilde{r}_k}=\dfrac{\left(M^{-1/2}r_k\right)^T\left(M^{-1/2}r_k\right)}{\left(M^{-1/2}r_k\right)^T\left(M^{-1/2}AM^{-1/2}\right)\left(M^{-1/2}r_k\right)}$

$$=\frac{r_k^T\left(M^{-1/2}\right)^T\left(M^{-1/2}r_k\right)}{r_k^T\left(M^{-1/2}\right)^T\left(M^{-1/2}AM^{-1/2}\right)\left(M^{-1/2}r_k\right)}$$

$$=\frac{r_k^TM^{-1}r_k}{r_k^T\left(M^{-1}AM^{-1}\right)r_k} \quad =\frac{r_k^T\left(M^{-1}r_k\right)}{\left(M^{-1}r_k\right)^TA\left(M^{-1}r_k\right)}$$

$$\tilde{A}\tilde{x}=\tilde{b} \quad \text{or} \quad \left(M^{-1/2}AM^{-1/2}\right)\left(M^{1/2}x\right)=\left(M^{-1/2}b\right)$$

$$\tilde{x}_{k+1}=\tilde{x}_k+\left(\frac{\tilde{r}_k^T\tilde{r}_k}{\tilde{r}_k^T\tilde{A}\tilde{r}_k}\right)\tilde{r}_k$$

$$\tilde{r}_k=\tilde{b}-\tilde{A}\tilde{x}_k$$

- $x_{k+1}=M^{-1/2}\tilde{x}_{k+1}=M^{-1/2}\left(\tilde{x}_k+\tilde{\alpha}_k\tilde{r}_k\right)$

$$=M^{-1/2}\left(M^{1/2}x_k+\tilde{\alpha}_k\left(M^{-1/2}r_k\right)\right)$$

$$=x_k+\tilde{\alpha}_k\left(M^{-1}r_k\right)$$

- $r_{k+1}=M^{1/2}\tilde{r}_{k+1}=M^{1/2}\left(\tilde{b}-\tilde{A}\tilde{x}_{k+1}\right)=M^{1/2}\left(\tilde{b}-\tilde{A}\left(\tilde{x}_k+\tilde{\alpha}_k\tilde{r}_k\right)\right)$

$$=M^{1/2}\left(\tilde{r}_k-\tilde{\alpha}_k\tilde{A}\tilde{r}_k\right)=M^{1/2}\left(M^{-1/2}r_k-\tilde{\alpha}_k\left(M^{-1/2}AM^{-1/2}\right)M^{-1/2}r_k\right)$$

$$=r_k-\tilde{\alpha}_kA\left(M^{-1}r_k\right)$$

$$\tilde{A}\tilde{x}=\tilde{b} \quad \text{or} \quad \left(M^{-1/2}AM^{-1/2}\right)\left(M^{1/2}x\right)=\left(M^{-1/2}b\right)$$

Step 1: take an initial guess x_0 and take $p_0=-\nabla\phi(x_0)=-(Ax_0-b)=r_0$

Step 2: if $\|r_k\|>\varepsilon$, $\quad \tilde{\alpha}=\dfrac{r_k^T\left(M^{-1}r_k\right)}{\left(M^{-1}r_k\right)^TA\left(M^{-1}r_k\right)}$

$$x_{k+1}=x_k+\tilde{\alpha}_k\left(M^{-1}r_k\right)$$

$$r_{k+1}=r_k-\tilde{\alpha}_kA\left(M^{-1}r_k\right)$$

§ Conjugate direction method

In the gradient iteration method:

$$x_k=x_{k-1}+\alpha_{k-1}p_{k-1}=x_{k-2}+\alpha_{k-2}p_{k-2}+\alpha_{k-1}p_{k-1}=\cdots$$

$$=x_0+\sum_{j=0}^{k-1}\alpha_jp_j \quad \left(\frac{\partial\phi}{\partial\alpha_j}=0 \text{ enforced for one } j \text{ each time}\right)$$

$$x_N=x_0+\sum_{j=0}^{N-1}\alpha_jp_j$$

- The gradient method searches the minimum of $\phi(x)$ in a direction $p_k=r_k$.

~ These searching directions $\{p_k\}$ may not be the fastest path leading to the minimum.

§ Conjugate direction method

x_0

$p_0,\ \alpha_0 \ \Rightarrow\ x_1 = x_0 + \alpha_0 p_0$

$p_1,\ \alpha_1 \ \Rightarrow\ x_2 = x_1 + \alpha_1 p_1$

Suppose we have moved along some direction p_0 to a minimum and now propose to move along some new direction p_1 .

We don't want that the new correction spoils our minimization along the p_0 direction.

$$\begin{aligned} p_0^T \nabla\phi(x_1) &= 0 \\ p_1^T \nabla\phi(x_2) &= 0 \end{aligned} \quad \Rightarrow \quad \begin{cases} p_0^T \nabla\phi(x_2) = 0 \\ p_1^T \nabla\phi(x_2) = 0 \end{cases}$$

§ Conjugate direction method

$$\begin{aligned} p_0^T \nabla\phi(x_1) &= 0 \\ p_1^T \nabla\phi(x_2) &= 0 \end{aligned} \quad \Rightarrow \quad \begin{cases} p_0^T \nabla\phi(x_2) = 0 \\ p_1^T \nabla\phi(x_2) = 0 \end{cases}$$

$$\begin{aligned} p_0^T \nabla\phi(x_2) &= p_0^T (Ax_2 - b) = p_0^T \left(A(x_1 + \alpha_1 p_1) - b\right) \\ &= p_0^T (Ax_1 - b) + p_0^T A\alpha_1 p_1 \\ &= p_0^T \nabla\phi(x_1) + p_0^T A\alpha_1 p_1 \\ &= \alpha_1 \left(p_0^T A p_1\right) \end{aligned}$$

$p_1^T \nabla\phi(x_2) = 0$

$p_0^T A p_1 = 0$

§ Conjugate direction method

• Improvement: select searching directions in a particular way such that

$\{p_k\}_{k=0}^{N-1}$ are linearly independent, e.g. make them A – orthogonal

$$p_k^T A p_j = 0 \quad \text{for } j = 0,1,2,\cdots,k-1$$

$$p_i^T A p_j = 0 \quad \text{for } i \neq j \text{ and } 0 \le i, j \le k-1$$

Thus for any $x \in R^N$, $\exists \{\alpha_k\}_{k=0}^{N} \ni x = \sum_{k=0}^{N} \alpha_k p_k$

i.e $\{p_k\}_{k=0}^{N-1}$ is a basis of R^N and $\{\alpha_k\}_{k=0}^{N-1}$ are the coordinates of x with respect to the basis.

§ Conjugate Gradient method

(one way to generate a set of A – orthongal vectors $\{p_k\}$)

starting: $p_0 = r_0 = b - Ax_0$

iterate: $p_k = r_k - \sum_{j=0}^{k-1} \left(\frac{r_k^T A p_j}{p_j^T A p_j} \right) p_j$

result: If $p_i^T A p_j = 0$ for $i \neq j,\ i, j \le k-1$, then $p_i^T A p_k = 0$ for all $i \le k-1$.

<show> for $i \le k-1$:

$$p_i^T A p_k = p_i^T A r_k - \sum_{j=0}^{k-1} \left(\frac{r_k^T A p_j}{p_j^T A p_j} \right) \underbrace{p_i^T A p_j}_{= 0 \text{ except } j = i}$$

$$p_i^T A p_k = p_i^T A r_k - \left(\frac{r_k^T A p_i}{p_i^T A p_i} \right) \left(p_i^T A p_i \right)$$

$$= p_i^T A r_k - \left(p_i^T A^T r_k \right)^T = 0 \ (\because A \text{ is symmetric})$$

§ Conjugate Gradient method

Step 1: take an initial guess x_0 and take $p_0 = r_0 = b - Ax_0$

DO

Step 2: $\alpha_k = \dfrac{r_k^T p_k}{p_k^T A p_k}$ and $x_{k+1} = x_k + \alpha_k p_k$ for $k \geq 0$

Step 3: $r_k = b - Ax_k$ and $p_k = r_k - \sum_{j=0}^{k-1} \left(\dfrac{r_k^T A p_j}{p_j^T A p_j} \right) p_j$ for $k \geq 1$

END DO

Theorem: The conjugate direction algorithm runs at most N iterations with $x_N = x^*$.

Theorem: $\|x_k - x^*\|_A \leq \dfrac{2c^k}{1+c^{2k}} \|x_0 - x^*\|_A$,

$$c = \frac{\sqrt{\kappa_2(A)} - 1}{\sqrt{\kappa_2(A)} + 1} \quad , \quad \kappa_2(A) = \lambda_{\max} / \lambda_{\min}$$

57

§ Conjugate Gradient method

Step 1: take an initial guess x_0 and take $p_0 = r_0 = b - Ax_0$

DO

Step 2: $\alpha_k = \dfrac{r_k^T r_k}{p_k^T A p_k}$ and $x_{k+1} = x_k + \alpha_k p_k$ for $k \geq 0$

Step 3: $r_k = b - Ax_k$ and $p_k = r_k - \dfrac{r_k^T r_k}{\left(r_{k-1}^T r_{k-1} \right)} p_{k-1}$ for $k \geq 1$

END DO

Theorem: $\|x_k - x^*\|_A \leq \dfrac{2c^k}{1+c^{2k}} \|x_0 - x^*\|_A$,

$$c = \frac{\sqrt{\kappa_2(A)} - 1}{\sqrt{\kappa_2(A)} + 1} \quad , \quad \kappa_2(A) = \lambda_{\max} / \lambda_{\min}$$

58