第三章 线性代数方程组的解法

§3.1 概述

一、研究数值解法的必要性

求：

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n = b_1 \\ a_{21}x_2 + a_{22}x_2 + \ldots + a_{2n}x_n = b_2 \\ \ldots\ldots\ldots \\ a_{n1}x_1 + a_{n2}x_2 + \ldots + a_{nn}x_n = b_n \end{cases}$$

的解 $x_1, x_2, \ldots x_n$ 的值，根据克莱姆(Gramer)法则可表示为两个行列式之比：

$$x_k = \frac{D_K}{D} (k = 1,2,\ldots n)$$

计算一个n 阶行列式需要做 $(n-1)(n!)$ 个乘法，求解上述方程共做 $N=(n+1)\times(n-1)(n!)+n$ 次乘除法。

如：$n=20$，$N\approx 9.7\times 10^{20}$，若用每秒完成万亿次（$10^{12}$）浮点乘法运算的计算机（当前国内运算速度最快），按每天工作24小时，完成这些计算约需30年。若使用一般的个人电脑，每秒不外完成十亿次（10^{9}）浮点乘法运算，则完成这些计算约需3万年。

二、线性代数方程组的常用解法

1、直接法：

　只包含有限次四则运算。若在计算过程中都不发生舍入误差的假定下，计算结果就是原方程组的精确解。

2、迭代法：

　把方程组的解向量看作是某种极限过程的极限，而且实现这一极限过程每一步的结果是把前一步所得的结果施行相同的演算步骤得到的。

Remark：由于运算过程中舍入误差的存在，实际上直接方法得到的解也是方程组的近始解。

§3.2 Gauss消去法

设有线性方程组

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \ldots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \ldots + a_{2n}x_n = b_2 \\ \ldots\ldots \\ a_{n1}x_1 + a_{n2}x_2 + \ldots + a_{nn}x_n = b_n \end{cases}$$

或写为矩阵形式 $A\vec{x} = \vec{b}$，其中

$$A = \begin{bmatrix} a_{11} & a_{12} & \ldots & a_{1n} \\ a_{21} & a_{22} & \ldots & a_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & \ldots & a_{nn} \end{bmatrix}$$ 为非奇异矩阵。 $$\vec{x} = \begin{bmatrix} x_1 \\ x_2 \\ \ldots \\ x_n \end{bmatrix} \quad \vec{b} = \begin{bmatrix} b_1 \\ b_2 \\ \ldots \\ b_n \end{bmatrix}$$

一、Gauss消去法

具体过程：

将$A\vec{x}=\vec{b}$改为$A^{(1)}\vec{x}=\vec{b}^{(1)}$

其中

$$\begin{cases} a_{11}^{(1)}x_1+a_{12}^{(1)}x_2+\ldots+a_{1n}^{(1)}x_n=b_1^{(1)} \\ a_{21}^{(1)}x_1+a_{22}^{(1)}x_2+\ldots+a_{2n}^{(1)}x_n=b_2^{(1)} \\ \cdots\cdots \\ a_{n1}^{(1)}x_1+a_{n2}^{(1)}x_2+\ldots+a_{nn}^{(1)}x_n=b_n^{(1)} \end{cases}$$

$$A^{(1)}=(a_{ij}^{(1)})_n=(a_{ij})_n \qquad \vec{b}_1=\vec{b}$$

Step1: 若 $a_{11}^{(1)} \neq 0$,令 $l_{i1} = a_{i1}^{(1)} \big/ a_{11}^{(1)}$ $(i = 2,3,\ldots n)$, 用 $-l_{i1}$ 乘第一个方程加到第 i 个方程 $(i = 2,3,\ldots n)$，并保留第一式，则得

$$\begin{cases} a_{11}^{(1)}x_1 + a_{12}^{(1)}x_2 + \ldots + a_{1n}^{(1)}x_n = b_1^{(1)} \\ \qquad a_{22}^{(2)}x_2 + \ldots + a_{2n}^{(2)}x_n = b_2^{(2)} \\ \qquad a_{32}^{(2)}x_2 + \ldots + a_{3n}^{(2)}x_n = b_3^{(2)} \\ \qquad \cdots\cdots \\ \qquad a_{n2}^{(2)}x_n + \ldots + a_{nn}^{(2)}x_n = b_n^{(2)} \end{cases}$$

记为 $A^{(2)}\vec{x} = \vec{b}^{(2)}$

其中 $a_{ij}^{(2)} = a_{ij}^{(1)} - l_{i1}a_{1j}^{(1)} \qquad (i, j = 2,3,\ldots,n)$

$b_i^{(2)} = b_i^{(1)} - l_{i1}b_1^{(1)} \qquad (i = 2,3,\ldots,n)$

Step2: 若 $a_{22}^{(2)} \neq 0$，令 $l_{i2} = a_{i2}^{(2)} / a_{22}^{(2)} \quad (i = 3,4,\ldots,n)$ ，用 $-l_{i2}$

乘第二个方程加到第i个方程 $(i = 3,4,\ldots,n)$ ，则得

$$
\begin{cases}
a_{11}^{(1)}x_1 + a_{12}^{(1)}x_2 + \cdots\cdots\cdots\cdots + a_{1n}^{(1)}x_n = b_1^{(1)} \\
\qquad a_{22}^{(2)}x_2 + \cdots\cdots\cdots\cdots + a_{2n}^{(2)}x_n = b_2^{(2)} \\
\qquad\qquad a_{33}^{(3)}x_3 + \ldots. + a_{3n}^{(3)}x_n = b_3^{(3)} \\
\cdots\cdots\cdots\cdots \\
\qquad\qquad a_{n3}^{(3)}x_3 + \ldots + a_{nn}^{(3)}x_n = b_n^{(3)}
\end{cases}
$$

记为 $A^{(3)}\vec{x} = \vec{b}^{(3)}$

其中 $a_{ij}^{(3)} = a_{ij}^{(2)} - l_{2i}a_{2j}^{(2)} \qquad (i, j = 3,4,\ldots,n)$

$b_i^{(3)} = b_i^{(2)} - l_{i2}b_2^{(2)} \qquad (i, j = 3,4,\ldots,n)$

第*k*-1步消元完成后，有

$$\begin{cases} a_{11}^{(1)}x_1 + a_{12}^{(1)}x_2 + \cdots\cdots\cdots\cdots + a_{1n}^{(1)}x_n = b_1^{(1)} \\ \qquad a_{22}^{(2)}x_2 + \cdots\cdots\cdots\cdots + a_{2n}^{(2)}x_n = b_2^{(2)} \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ \qquad\qquad a_{kk}^{(k)}x_k + \ldots + a_{kn}^{(k)}x_n = b_k^{(k)} \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ \qquad\qquad a_{nk}^{(k)}x_k + \ldots + a_{nn}^{(k)}x_n = b_n^{(k)} \end{cases}$$

设为 $A^{(k)}\vec{x} = \vec{b}^{(k)}$

Stepk：若 $a_{kk}^{(k)} \neq o$ ，令 $l_{ik} = a_{ik}^{(k)} / a_{kk}^{(k)}$ ，（i=k+1, k+2, …）

用 $-l_{ik}$ 来乘以第*k*-1步所得方程中的第*k*个方程，加到第i（$i=k+1,k+2,\ldots,n$）个方程，并保留第*k*个方程，则得：

$$
\begin{cases}
a_{11}^{(1)}x_1 + a_{12}^{(1)}x_2 + \ldots + a_{1n}^{(1)}x_n = b_1^{(1)} \\
\qquad a_{22}^{(2)}x_2 + \ldots + a_{2n}^{(2)}x_n = b_2^{(2)} \\
\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\
\qquad\qquad a_{kk}^{(k)}x_k + \ldots + a_{kn}^{(k)}x_n = b_k^{(k)} \\
\qquad\qquad a_{k+1,k+1}^{(k+1)}x_{k+1} + \ldots + a_{k+1,n}^{(k+1)}x_n = b_{k+1}^{(k+1)} \\
\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\
\qquad\qquad a_{n,k+1}^{(k+1)}x_{k+1} + \ldots + a_{n,n}^{(k+1)}x_n = b_n^{(k+1)}
\end{cases}
$$

上面的方程记为 $A^{(k+1)}\vec{x} = \vec{b}^{(k+1)}$

其中 $a_{ij}^{(k+1)} = a_{ij}^{(k)} - l_{ij}a_{kj}^{(k)}$ (i,j=k+1,…,n)

$b_i^{(k+1)} = b_i^{(k)} - l_{ik}b_k^{(k)}$ (i=k+1,…,n)

按上述做法，做完n-1步消元，原方程可化为同解的上三角方程组：

$$\begin{cases} a_{11}^{(1)}x_1 + a_{12}^{(1)}x_2 + \ldots + a_{1n}^{(1)}x_n = b_1^{(1)} \\ \qquad a_{22}^{(2)}x_2 + \ldots + a_{2n}^{(2)}x_n = b_2^{(2)} \\ \cdots\cdots \\ \qquad\qquad a_{nn}^{(n)}x_n = b_n^{(n)} \end{cases}$$

记为 $A^{(n)}\vec{x} = \vec{b}^{(n)}$

最后，若 $a_{nn}^{(n)} \neq 0$，逐步回代可得到原方程组的解：

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

（i=n-1,n-2,…2，1）

上面的求解过程称为Gauss顺序消去法。它通过一系列消元过程与最后的一步回代过程来得到方程组的解。

Remark1：在Gauss顺序消去法的消去过程中，可以将右端列向量视为方程组A的第n＋1列，直接对矩阵A（指现在的n行，n＋1列的增广矩阵）进行行初等变换，将其变换为上三角形矩阵，从而回代求解得到方程组的解。

Remark2：可以统计出，如果A为n阶方阵，则**Gauss**顺序消去法消去过程所需的乘除运算次数为

$$\sum_{k=1}^{n-1}\left(n-k+(n-k)(n-k+1)\right)=\frac{n^3}{3}+\frac{n^2}{2}-\frac{5n}{6}$$

回代过程所需的乘除运算次数为 $\sum_{k=1}^{n}\left(k-1+1\right)=\frac{n(n+1)}{2}$

故总的乘除运算次数为 $N=\frac{n^3}{3}+n^2-\frac{n}{3}\approx O(n^3)$

取n＝20，则N＝3060，与Gramer法则相比，是天壤之别。

Remark3：在消去过程中，也可以采用**Gauss-Jordan**消去法，将方程组化为对角形方程组，进而化为单位阵，则右端列向量就化为方程组的解向量。该方法不需回代过程，但总的计算量为$n^3/2+n^2-n/2$，比Gauss顺序消去法有所增加。

Remark4：在消去过程中，消去过程能够进行的前提条件是$a_{kk}^{(k)} \neq 0$。当detA=$\Delta_n \neq 0$时方程组存在唯一解，但未必能满足$a_{kk}^{(k)} \neq 0$的条件。要使Gauss顺序消去法能够求得方程组的解，应满足如下的定理：

定理：用高斯顺序消去法能够求解方程组A$\vec{x} = \vec{b}$的解的充要条件为A的各阶顺序主子式均不为零。

说明：①当 $\Delta_k = a_{11}^{(1)} \cdots a_{kk}^{(k)}$（**k=1, 2, …， n**）均不为零时高斯顺序消去法能进行下去，但当 $\left|a_{kk}^{(k)}\right| \approx 0$ 或相对于 $\left|a_{ik}^{(k)}\right|$（**i=k+1,k+2…， n**）比较小时，计算时产生的舍入误差将导致计算结果误差增大。

②当 detA=0, $a_{kk}^{(k)} = 0.$ 此时高斯顺序消去法能进行下去，但不能求出唯一解。

③由于舍入误差的原因，**Gauss**顺序消去法不是一个实用的方法，实用中应该采用选主元的**Gauss**消去法。

二、主元素消去法

$a_{kk}^{(k)}$称为第k步的主元

在计算过程中舍入误差增大迅速，造成计算解与真解相差甚远，这一方法就是不稳定的方法，反之，在计算过程中的舍入误差增大能得到控制，该方法就是稳定的。小主元是不稳定的根源，这就需要采用“选主元素”技术，即选取绝对值最大的元素作为主元。

1)列主元消去法

一种常用的方法是列主元消去法。设增广矩阵为

$$B=\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} & b_1 \\ a_{21} & a_{22} & \cdots & a_{2n} & b_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{i_1,1} & a_{i_1,2} & \cdots & a_{i_1,n} & b_{i_1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} & b_n \end{bmatrix} = \left[A^{(1)} \vdots \vec{b}^{(1)}\right]$$

在第一列中选取绝对值最大的元素,如若 $\left|a_{i_1,1}\right| = \max_{1\le i\le n}\left|a_{i1}\right|$

调换第一行与第i行，这时的 $a_{11}^{(1)}$ 就是原来的 $a_{i_1,1}$,再进行消去法的第一步，即

$$\begin{bmatrix} a_{11}^{(1)} & a_{12}^{(1)} & \cdots & a_{1n}^{(1)} & b_1^{(2)} \\ & a_{22}^{(2)} & \cdots & a_{2n}^{(2)} & b_2^{(2)} \\ & \vdots & \ddots & \vdots & \vdots \\ & a_{n2}^{(2)} & \cdots & a_{22}^{(2)} & b_n^{(2)} \end{bmatrix} = \left[A^{(2)} \vdots \vec{b}^{(2)}\right]$$

再考虑 $A^{(2)}$ 右下角矩阵，选取绝对值最大的元素作为主元素,经过行的对换把主元素移到 $a_{22}^{(2)}$，

再按消元公式计算 $a_{ij}^{(3)}$ (i,j=3,…n）。

然后每一步类似的都在右下角方阵中的第一列中选主元，再经行对调, 将主元素换到右下脚方阵中左上角的位置。再按下一步计算 $a_{ij}^{(k)}$（i, j=k…n)

直至将方程组化成上三角方程组，再进行回代就可求得解。

2) 全主元消去法

$$B=\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} & b_1 \\ a_{21} & a_{22} & \cdots & a_{2n} & b_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{i,1} & a_{i,2} & \cdots & a_{i,n} & b_i \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} & b_n \end{bmatrix} = \left[A^{(1)} \vdots \vec{b}^{(1)}\right]$$

在A中选取绝对值最大的元素作为主元素，如

$\left|a_{i_1,j_1}\right| = \max\limits_{\substack{1\le i\le n \\ 1\le j\le n}} \left|a_{ij}\right|$，**然后交换B 的第一行与第 i_1行，第一列与第 j_1列，这时的 $a_{11}^{(1)}$就是元素的 a_{i_1,j_1}，然后进行消元法的第一步，即**

$$\begin{bmatrix} a_{11}^{(1)} & a_{12}^{(1)} & \cdots & a_{1n}^{(1)} & b_1^{(2)} \\ & a_{22}^{(2)} & \cdots & a_{2n}^{(2)} & b_2^{(2)} \\ & \vdots & & & \vdots \\ & a_{n2}^{(2)} & \cdots & a_{nn}^{(2)} & b_n^{(2)} \end{bmatrix} = \left[A^{(2)} \vdots \vec{b}^{(2)}\right]$$

再考虑 $A^{(2)}$ 右下角矩阵，选取绝对值最大的元素作为主元素, 经过行的对换和列的对换把主元素移到 $a_{22}^{(2)}$ 的位置, 再按消元公式计算 $a_{ij}^{(3)}$（i,j=3,…n）,然后每一步类似的都在右下角方阵中进行完全选主元，经过行对调列对调将主元换到右下角方阵中左上角的位置，再按下一步计算 $a_{ij}^{(k)}$（i,j=k…n）直至将方程组化成上三角方程组，再进行回代就可得解。

Remark

Reamrk1：全主元消去法每一步所选取的主元的绝对值不低于列主元消去法的同一步所选主元的绝对值，因而计算结果更加可靠。

Reamrk2：全主元消去法每一步选主元要花费更多的机时，并且对增广矩阵做了列交换，从而未知量的次序发生了变化，对编程带来了困难。而列主元消去法的计算结果已比较可靠，故实用中常用列主元消去法。

Reamrk3：选主元的消去法与Gauss顺序消去法的乘除法的计算量是一样的，均为$n^3/3+n^2-n/3$。

三、选主元素消去法的应用

1.求解方程组系

设系数矩阵A可逆，求解方程组系$Ax= b_i$(i=1,2,…,m)。

将方程组系的系数矩阵与右端向量写成增广矩阵

$$[A/b_1,b_2,\dots,b_m]$$

按列选主元素后再用Gauss-Jordan消去法将左边的矩阵A化为单位矩阵，则得到

$$\left[\begin{array}{cccc|cccc} 1 & & & & \bar{b}_{11} & \bar{b}_{12} & \cdots & \bar{b}_{1m} \\ & 1 & & & \bar{b}_{21} & \bar{b}_{22} & \cdots & \bar{b}_{2m} \\ & & \ddots & & \vdots & \vdots & \ddots & \vdots \\ & & & 1 & \bar{b}_{n1} & \bar{b}_{n2} & \cdots & \bar{b}_{nm} \end{array}\right]$$

右端的列向量就是方程组系的解向量。

2.求逆矩阵

在1中令$m=n$，且右端列向量组成单位阵，则当A化为单位阵时，右端矩阵即为A的逆矩阵。这是实用中求逆矩阵的可靠方法。

3.求行列式

若要求矩阵A的行列式，可用主元素消去法将其化为上三角阵，对角元素设为b_{11}，b_{22}，…，b_{nn}，于是A的行列式为：

$$\det(A)=(-1)^m b_{11}b_{22}\dots b_{nn}$$

其中m为行列交换的次数。这是实用中求行列式的可靠方法。

§3.3 矩阵的三角分解法

1.Gauss顺序消去法的矩阵形式

设方程组A$\vec{x}=\vec{b}$中A的各阶顺序主子式均不为零，令

$$L_1=\begin{bmatrix} 1 & & & & \\ -l_{21} & 1 & & & \\ -l_{31} & \ddots & 1 & & \\ \vdots & \ddots & \ddots & \ddots & \\ -l_{n1} & \cdots & \cdots & \cdots & 1 \end{bmatrix} \quad L_2=\begin{bmatrix} 1 & & & & \\ 0 & 1 & & & \\ 0 & -l_{32} & 1 & & \\ \vdots & \vdots & \ddots & \ddots & \\ 0 & -l_{n2} & \cdots & \cdots & 1 \end{bmatrix}$$

$$\vdots$$

$$L_k = \begin{bmatrix} 1 & & & & & \\ & 1 & & & & \\ & & 1 & & & \\ & & -l_{k+1,k} & \ddots & & \\ & & \vdots & \vdots & \ddots & \\ & & -l_{n,k} & \cdots & \cdots & 1 \end{bmatrix}$$

$$\vdots$$

$$L_{n-1} = \begin{bmatrix} 1 & & & & \\ & 1 & & & \\ & & 1 & & \\ & & & \ddots & \\ & & & -l_{n,n-1} & 1 \end{bmatrix}$$

则n-1步消元过程为

$$L_1\left[A^{(1)} \vdots \vec{b}^{(1)}\right] = \left[A^{(2)} \vdots \vec{b}^{(2)}\right]$$

$$L_2\left[A^{(2)} \vdots \vec{b}^{(2)}\right] = \left[A^{(3)} \vdots \vec{b}^{(3)}\right]$$

$$\vdots$$

$$L_k\left[A^{(k)} \vdots \vec{b}^{(k)}\right] = \left[A^{(k+1)} \vdots \vec{b}^{(k+1)}\right]$$

$$\vdots$$

$$L_{n-1}\left[A^{(n-1)} \vdots \vec{b}^{(n-1)}\right] = \left[A^{(n)} \vdots \vec{b}^{(n)}\right]$$

将上述n-1步消元过程合并，得，

$$L_{n-1}L_{n-2}\cdots L_2L_1\left[A^{(1)} \vdots \vec{b}^{(1)}\right] = \left[A^{(n)} \vdots \vec{b}^{(n)}\right]$$

即 $\begin{cases} L_{n-1}L_{n-2}\cdots L_1 A^{(1)} = A^{(n)} \\ L_{n-1}L_{n-2}\cdots L_1 b^{(1)} = \vec{b}^{(n)} \end{cases}$ （1） 由 $A^{(1)} = A$得：

$$A = (L_{n-1}\cdots L_1)^{-1}A^{(n)} = L_1^{-1}L_2^{-1}\cdots L_{n-1}^{-1}A^{(n)}$$

容易验证：

每个 L_k 可逆，且 $L_k^{-1} = \begin{bmatrix} 1 & & & & & & \\ & \ddots & & & & & \\ & & 1 & & & & \\ & & l_{k+1,k} & \ddots & & & \\ & & l_{k+2,k} & & 1 & & \\ & & \vdots & & & \ddots & \\ & & l_{n,k} & & & & 1 \end{bmatrix}$ k=1,2,…， n-1

令$L=L_1^{-1}L_2^{-1}\cdots L_{n-1}^{-1}$ 则 $L=\begin{bmatrix} 1 & & & & \\ l_{21} & 1 & & & \\ l_{31} & l_{32} & \ddots & & \\ \vdots & \vdots & & \ddots & \\ l_{n1} & l_{n2} & \cdots & l_{n,n-1} & 1 \end{bmatrix}$

又令$U=A^{(n)}$，且 $U=\begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ & u_{22} & & \vdots \\ & & \ddots & \vdots \\ & & & u_{nn} \end{bmatrix}$ 则A=LU，

其中，L为单位下三角矩阵，它的对角元素皆为1。

U为上三角矩阵,由（1）式,

$$\text{即}\quad \begin{cases} L^{-1}A = U \\ L^{-1}\vec{b} = \vec{y} \end{cases}$$

Gauss消去法的实质就是用一连串的初等变换把系数方阵A化成上三角方阵U，同时把右端向量化$\vec{b}$**为**$\vec{y}$**,若矩阵A=LU，则LU叫做矩阵A的LU分解**.

2 .矩阵的三角分解及条件

定义1.设A为n阶矩阵（$n\geq 2$）,称A=LU为矩阵的三角分解，其中L是下三角矩阵，U是上三角矩阵。

Remark：三角分解不唯一，可以有不同的形式 。为确保唯一性，引入以下定义。

定义2.如果L是单位下三角矩阵，U是上三角矩阵，则称三角分解A=LU为Doolittle分解；如果L是下三角矩阵，U是单位上三角矩阵，则称A=LU为Crout分解。前面高斯顺序消去法对应了A=LU的Doolittle分解。

定理： 设A为n阶可逆矩阵，则A有唯一Doolittle(or Crout)分解的充要条件为：A的前n-1阶顺序主子式不为零。

$A\vec{x}=\vec{b}$ 的求解过程为： $\begin{cases} L\vec{y}=\vec{b} \\ U\vec{x}=\vec{y} \end{cases}$

可推导求解单位下三角形方程组 $L\vec{y}=\vec{b}$ 的递归公式为：

$$y_k = b_k - \sum_{m=1}^{k-1} l_{km} y_m \quad k=1,2,3,\cdots,n \qquad \sum_{k=1}^{0} = 0$$

求解上三角形方程组 $U\vec{x}=\vec{y}$ 的递归公式为：

$$x_k = \left(y_k - \sum_{m=k+1}^{n} u_{km} x_m \right) \Big/ u_{kk} \qquad k=n,n-1,\cdots 1。\sum_{k=n+1}^{n} = 0$$

Remark：实际中对A进行三角分解，不是利用初等变换矩阵，而是直接使用矩阵乘法得到。（若不加说明，后面我们讲到的三角分解一律指Doolittle型分解。）

3.直接三角分解法

设A=LU 即

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} = \begin{bmatrix} 1 & & & & \\ l_{21} & 1 & & & \\ l_{31} & l_{32} & 1 & & \\ \vdots & & & \ddots & \\ l_{n1} & l_{n2} & l_{n3} & \cdots & 1 \end{bmatrix} \begin{bmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ & u_{22} & \cdots & u_{2n} \\ & & \ddots & \vdots \\ & & & u_{nn} \end{bmatrix}$$

Step1：

比较第一行元素： $a_{1j} = u_{1j} (j = 1,2,\cdots n)$

比较第一列元素： $a_{i1} = l_{i1} u_{11}$ **解出** $l_{i1} = a_{i1} / u_{11} \ (i = 2,3,\cdots n)$

Step2：

比较第二行元素： $a_{2j} = l_{21}u_{1j} + u_{2j}(j = 2,3,\cdots,n)$

算出： $u_{2j} = a_{2j} - l_{21}u_{1j} \quad (j = 2,3,\cdots,n)$

比较第二列的元素： $a_{i2} = l_{i1}u_{12} + l_{i2}u_{22}$

得出： $l_{i2} = (a_{i2} - l_{i1}u_{12}) \big/ u_{22} \quad (i = 3,4,\cdots,n)$

一般地，设U的前k-1行以及L的前k-1列已求出，则

Stepk：

比较第k行元素

$$a_{kj} = \sum_{m=1}^{n} l_{km}u_{mj} = \sum_{m=1}^{k-1} l_{km}u_{mj} + u_{kj}l_{kk} + \sum_{m=k+1}^{n} l_{km}u_{mj}$$

可以算出： $u_{kj} = a_{kj} - \sum_{m=1}^{k-1} l_{km} u_{mj} \qquad j = k, k+1, \cdots, n$

比较第k列元素(i>k 即行指标>列指标，为算 l_{ik}, i>k)

$$a_{ik} = \sum_{m=1}^{n} l_{im} u_{mk} = \sum_{m=1}^{k-1} l_{im} u_{mk} + l_{ik} u_{kk}$$

算出: $l_{ik} = \left(a_{ik} - \sum_{m=1}^{k-1} l_{im} u_{mk} \right) \Big/ u_{kk} \qquad i = k+1, \cdots, n$

这组公式可用下图记忆（紧凑格式）：

$$\begin{matrix} u_{11} & u_{12} & u_{13} \cdots & u_{1n} \\ l_{21} & u_{22} & u_{23} \cdots & u_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ l_{n1} & l_{n2} & l_{n3} & u_{nn} \end{matrix}$$

$A\vec{x}=\vec{b}$ 的求解过程为： $\begin{cases} L\vec{y}=\vec{b} \\ U\vec{x}=\vec{y} \end{cases}$

可推导求解单位下三角形方程组 $L\vec{y}=\vec{b}$ **的递归公式为**：

$$y_k = b_k - \sum_{m=1}^{k-1} l_{km} y_m \qquad k=1,2,3,\cdots,n \qquad \sum_{k=1}^{0} = 0$$

求解上三角形方程组 $U\vec{x}=\vec{y}$ **的递归公式为:**

$$x_k = \left(y_k - \sum_{m=k+1}^{n} u_{km} x_m \right) \Big/ u_{kk} \qquad k=n,n-1,\cdots 1 \qquad \sum_{k=n+1}^{n} = 0$$

对比计算 u_{kj} 和 y_k 公式，$u_{kj} = a_{kj} - \sum_{m=1}^{k-1} l_{km} u_{mj}$，

$$y_k = b_k - \sum_{m=1}^{k-1} l_{km} y_m$$

发现计算y_k的规律与计算 u_{kj} 的规律类似,因此计算 $\vec{y}$ 的求方程组的过程可用三角分解的紧凑格式取代。事实上,这只要把 $\vec{b}$ 做为A的第n + 1列进行直接三角分解即可。

Reamrk：上述直接三角分解法所对应的是Gauss顺序消去法，二者的乘除运算次数是相当的。实际中对阶数较高的线性方程组，应采用选主元的三角分解法求解，以保证计算结果的可靠性。

4.平方根法（Cholesky分解）

定义：设A为n阶$(n \geq 2)$对称正定矩阵，L是非奇异下三角矩阵，称$A = LL^T$为矩阵A的Cholesky分解。

定理：n阶$(n \geq 2)$对称正定矩阵A，一定存在如下的分解式$(1) A = LDL^T$，其中L为单位下三角阵，D是对角元全为正的对角阵，且这种分解式唯一；$(2) A = LL^T$，其中L为下三角阵，当限定L的对角元为正时，$A = LL^T$（Cholesky分解）的分解式唯一。

平方根法的计算公式

（为简单起见设 $A=LL^T$，L为下三角矩阵）

令

$$A=\begin{bmatrix} l_{11} & & & \\ l_{21} & l_{22} & & \\ \vdots & & \ddots & \\ l_{n1} & & \cdots & l_{nn} \end{bmatrix}\begin{bmatrix} l_{11} & l_{21} & \cdots & l_{n1} \\ & l_{22} & & \\ & & \ddots & \\ & & & l_{nn} \end{bmatrix}$$

我们可以通过矩阵乘法比较来求L的下三角部分元素。具体计算公式如下:

$$
\begin{cases}
l_{11} = \sqrt{a_{11}} \\
l_{i1} = a_{i1} \big/ l_{11} \\
l_{kk} = \sqrt{a_{kk} - \sum_{m=1}^{k-1} l_{km}^2}, \quad l_{ik} = \left(a_{ik} - \sum_{m=1}^{k-1} l_{im} l_{km} \right) \Big/ l_{kk} \\
\quad (j = k+1, k+2, \ldots n)(k = 2,3,\ldots,n)
\end{cases}
$$

Remark1:由于在上式分解过程中有n次开方运算，故Choledsky分解法也称为平方根法。

Remark2: 因为 $a_{kk}=\sum_{m=1}^{k} l_{km}^2 \quad \therefore l_{km}^2 \le a_{kk} \le \max|a_{kk}|$ ，这说明 l_{km} 的平方不会超过A的最大对角元，因而舍入误差的放大受到控制，故用平方根法解对称正定的方程组时，不必考虑选主元的问题，算法本身数值稳定。

Remark3:可以证明，若用**Gauss**顺序消去法求解对称正定的方程组**Ax**＝**b**，则有

$$\max_{1\le i,j\le n}\left|a_{ij}^{(k)}\right| \le \max_{1\le i,j\le n} a_{ii} \qquad (k=1,2,\cdots,n)$$

其中 $a_{ij}^{(k)}$ 是第**k**步**Gauss**顺序消去过程所得元素。这说明用**Gauss**顺序消去法求解对称正定方程组也不用选主元。

Remark4:从运算量的角度看，平方根法是有利的。可以统计出，用平方根法求解Ax = b所需乘除法的运算次数为：

$$\frac{1}{6}\left(n^3+9n^2+2n\right)\approx\frac{1}{6}n^3$$

令有n次开方运算。 n次开方运算一般使用迭代法，所需乘除法的运算次数大约为n的常数倍。故平方根法总的乘除法运算次数大约为

$$\frac{1}{6}\left(n^3+9n^2+cn\right)\approx\frac{1}{6}n^3$$

Remark5：为避免开方运算，也可对A做$A=LDL^T$分解。(其中L为单位下三角阵,D为对角阵)

5、解三对角方程组的追赶法

1.矩阵对角占优的概念

设 $A=\left(a_{ij}\right)_{n*n}$

若对于$i=1,2,3,\cdots,n$ ，不等式$\left|a_{ii}\right|\geq\sum\limits_{\substack{j=1\\j\neq 1}}^{n}\left|a_{ij}\right|$均成立，且至少对某个$i_0$有 $\left|a_{i_0 i_0}\right|>\sum\limits_{\substack{j=1\\j\neq i_0}}^{n}\left|a_{i_0 j}\right|$，则称矩阵A按行对角占优。

若对于$i=1,2,3,\cdots,n$ ，均有 $\left|a_{ii}\right|>\sum\limits_{\substack{j=1\\j\neq i}}^{n}\left|a_{ij}\right|$，则对称矩阵A按行严格对角占优。

类似地，也有按列对角占优和按列严格对角占优的概念。

2.追赶法解三对角方程组

设
$$\begin{bmatrix} b_1 & c_1 & & & & & & \\ a_2 & b_2 & c_2 & & & & & \\ & a_3 & b_3 & c_3 & & & & \\ & & \ddots & \ddots & \ddots & & & \\ & & & a_i & b_i & c_i & & \\ & & & & \ddots & \ddots & \ddots & \\ & & & & & a_{n-1} & b_{n-1} & c_{n-1} \\ & & & & & & a_n & b_n \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ \vdots \\ x_i \\ \vdots \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ \vdots \\ d_i \\ \vdots \\ \vdots \\ d_n \end{bmatrix}$$

并满足严格对角占优条件 $|b_1| > |c_1|$ $|b_i| > |a_i| + |c_i|$

$|b_n| > |a_n|$ $i = 2,3,\cdots,n-1$

当A严格对角占优时，可以证明各阶顺序主子式非零。

当A为上述三对角阵时，A有更特殊的三角分解形式 *A=L U* 即：

$$\begin{bmatrix} b_1 & c_1 & & & \\ a_2 & b_2 & c_2 & & \\ & \ddots & \ddots & \ddots & \\ & & a_{n-1} & b_{n-1} & c_{n-1} \\ & & & a_n & b_n \end{bmatrix} = \begin{bmatrix} 1 & & & & \\ l_2 & 1 & & & \\ & l_3 & 1 & & \\ & & \ddots & \ddots & \\ & & & l_n & 1 \end{bmatrix} \begin{bmatrix} u_1 & c_1 & & & \\ & u_2 & c_2 & & \\ & & \ddots & \ddots & \\ & & & u_{n-1} & c_{n-1} \\ & & & & u_n \end{bmatrix}$$

事实上，比较第i行的第i+1列，立刻可得U的次上三角线元素为 c_i，由矩阵相乘可得 $b_1 = u_1$，一般地比较第i行第i−1列 $a_i = l_i u_{i-1}$，比较第i行第i列 $b_i = l_i c_{i-1} + u_i$

即 $\begin{cases} u_1 = b_1 \\ l_i = a_i \big/ u_{i-1} \\ u_i = b_i - l_i c_{i-1} \end{cases} \qquad i = 2,3,\cdots,n$

$A\vec{x} = \vec{d}$ 等价于 $LU\vec{x} = \vec{d}$

令 $\vec{y} = U\vec{x}$ 即 $\begin{cases} L\vec{y} = \vec{d} \\ U\vec{x} = \vec{y} \end{cases}$

由 $\begin{bmatrix} 1 & & & \\ l_2 & 1 & & \\ & \ddots & \ddots & \\ & & l_n & 1 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_n \end{bmatrix}$ 得 $\begin{cases} y_1 = d_1 \\ y_i = d_i - l_i y_{i-1} \end{cases} \qquad i = 2,3,\cdots,n$

由 $$\begin{bmatrix} u_1 & c_1 & & \\ & u_2 & \ddots & \\ & & \ddots & c_{n-1} \\ & & & u_n \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}=\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$ 得 $$\begin{cases} x_n = y_n / u_n \\ x_i = (y_i - c_i x_{i+1}) / u_i \end{cases} \quad i = n-1, \cdots, 2, 1$$

Remark：**只要三对角矩阵按行严格对角占优，则追赶法定能进行下去，且计算过程是稳定的（不必选主元素），其乘除法运算次数为5n - 4。上述方法称为解三对角方程组的追赶法，又称为Thomas方法。**

§3.4解线性方程组的迭代法

一、简单迭代法

1.简单迭代法的构造

迭代法的一般格式为

$$\vec{x}^{(k+1)} = f_k(\vec{x}^{(k)}, \vec{x}^{(k-1)}, \cdots \vec{x}^{(k-m)}) \quad k = 0,1,\cdots$$

因为计算$\vec{x}^{(k+1)}$一般要用到前面多步的值$\vec{x}^{(k)}, \vec{x}^{(k-1)}, \cdots, \vec{x}^{(k-m)}$故称为多步迭代法。

若 $m = 0$，即 $\vec{x}^{(k+1)} = f_k(\vec{x}^{(k)})$,$k = 0,1,\cdots$， **称为单步迭代法。**

若 f_k 为线性的，即 $\vec{x}^{(k+1)} = B_k\vec{x}^{(k)} + \vec{g}_k$, $k = 0,1,\cdots$, 称为单步线性迭代法,B_k称为迭代矩阵。

若 $B_k, \vec{g}_k$ 与 k 无关，即 $\vec{x}^{(k+1)} = B\vec{x}^{(k)} + \vec{g}$，$k$=0,1,…,，称为单步定常线性迭代法，或者叫简单迭代法。

设要求解的线性方程组为 $A\vec{x} = \vec{b}$ ，其中 A 为非奇异矩阵，$\vec{b}$ 为向量。

将该方程组等价变形为 $\vec{x} = B\vec{x} + \vec{g}$ 构造简单迭代格式 $\vec{x}^{(k+1)} = B\vec{x}^{(k)} + \vec{g}$, $k = 0,1,\cdots$。若 $\{\vec{x}^{(k)}\}$ 收敛于确定的向量 $\vec{x}^*$，则 $\vec{x}^*$ 就是方程组的解。此时称简单迭代法 $\vec{x}^{(k+1)} = B\vec{x}^{(k)} + \vec{g}$ ，$k = 0,1,\cdots$ 关于初始向量 $\vec{x}^{(0)}$ 收敛。

①如果对初始向量$\vec{x}^{(0)}$，$\lim\limits_{k\to\infty}\vec{x}^{(k)}=\vec{x}^{*}$，则称此简单迭代法关于初始向量$\vec{x}^{(0)}$收敛。一般谈及收敛，是指对任意$\vec{x}^{(0)}$，均有$\lim\limits_{k\to\infty}\vec{x}^{(k)}=\vec{x}^{*}$。

②同一个简单迭代法可以关于某一个$\vec{x}^{(0)}$收敛，而关于另外$\vec{x}^{(0)}$不收敛。

③$A\vec{x}=\vec{b}$ 变形为$\vec{x}=B\vec{x}+\vec{g}$的方式不唯一。

④ 当收敛时,只要 k 充分大，则可用 $\vec{x}^{(k+1)}$ 作为 $\vec{x}^{*}$ 的近似值。

2.简单迭代法的收敛性

(1) 收敛的充要条件

定义：设迭代矩阵B为n阶矩阵，$\lambda(B)$为矩阵B的特征值，称 $\rho(B)=\max\limits_{1\le i\le n}|\lambda_i(B)|$为矩阵$B$的谱半径。

定理:简单迭代法 $\vec{x}^{(k+1)}=B\vec{x}^{(k)}+\vec{g}$ ， $k=0,1,\cdots$ ， 对任意初始向量$\vec{x}^{(0)}$都收敛的充要条件是：

迭代矩阵的谱半径 $\rho(B)<1$

Remark:

① $\rho(B)<1$是判定收敛的根本法则。（判定较繁）

② $\rho(B)\ge 1$时，有可能存在某个初始向量 $\vec{y}^{(0)}$使简单迭代法收敛。（该向量不好找）

(2)收敛的充分条件

定理：设简单迭代公式为 $\vec{x}^{(k+1)} = B\vec{x}^{(k)} + \vec{g} \quad (k = 0,1,2,\cdots)$

若 $\|B\|_1 = \max\limits_{1\le j\le n} \sum\limits_{i=1}^{n} |b_{ij}| < 1$ **或** $\|B\|_\infty = \max\limits_{1\le i\le n} \sum\limits_{j=1}^{n} |b_{ij}| < 1$

则简单迭代格式关于任意初始向量 $\vec{x}^{(0)}$ **和** $\vec{g}$ **收敛。**

证明： 仅以 $\|B\|_\infty = \max\limits_{1\le i\le n} \sum\limits_{j=1}^{n} |b_{ij}| < 1$ 为例。

设$\vec{x}$是n阶非奇次线性方程组$A\vec{x} = \vec{b}$的解向量，

$\vec{x} = B\vec{x} + \vec{g}$ 是其等价形式。记

$$\delta_k = \max_{1\le i\le n} \left| x_i^{(k)} - x_i \right| (k = 0,1,2,\cdots)$$

由

$$\begin{cases} \vec{x}^{(k+1)} = B\vec{x}^{(k)} + \vec{g} \\ \vec{x} = B\vec{x} + \vec{g} \end{cases} \qquad k=0,1,\ldots$$

的分量形式

$$x_i^{(k+1)} = \sum_{j=1}^{n} b_{ij} x_j^{(k)} + g_i \qquad x_i = \sum_{j=1}^{n} b_{ij} x_j + g_i$$

相减有

$$x_i^{(k+1)} - x_i = \sum_{j=i}^{n} b_{ij} (x_j^{(k)} - x_j)$$

$$\left| x_i^{(k+1)} - x_i \right| \le \sum_{j=i}^{n} \left| b_{ij} \right| \left| x_j^{(k)} - x_j \right| \le \delta_k \sum_{j=i}^{n} \left| b_{ij} \right| \le \delta_k \|B\|_\infty$$

由于上式对于*i*=1,2,…,n都成立，故有

$$\delta_{k+1} \le \|B\|_\infty \delta_k$$

由此递推可得

$$\delta_{k+1} \le \|B\|_\infty \delta_k \le \|B\|_\infty^2 \delta_{k-1} \le \cdots \le \|B\|_\infty^{k+1} \delta_0$$

因为 $0 \le \|B\|_{\infty} < 1$，所以

$$\lim_{k\to\infty} \delta_{k+1} = 0$$

即

$$\lim_{k\to\infty} (x_i^{(k+1)} - x_i) = 0 \quad (i = 1,2,\cdots,n)$$

Remark：由此定理可以知道，在将方程组作等价变形时，若使矩阵*B*的元素b_{ij}的绝对值尽可能的小，就有可能使简单迭代法收敛。

3.Jacobi迭代法

(1)迭代格式

设有 *n* 阶方程组

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n = b_2 \\ \cdots \\ a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n = b_n \end{cases}$$

其中系数矩阵非奇异，且 $a_{ii} \neq 0$，i=1,2，……，n

将上式变形为

$$\begin{cases} x_1 = \dfrac{1}{a_{11}}(-a_{12}x_2 - a_{13}x_3 - \cdots - a_{1n}x_n + b_1) \\ x_2 = \dfrac{1}{a_{22}}(-a_{21}x_1 - a_{23}x_3 - \cdots - a_{2n}x_n + b_2) \\ \cdots \\ x_n = \dfrac{1}{a_{nn}}(-a_{n1}x_1 - a_{n2}x_2 - \cdots - a_{n,n-1}x_{n-1} + b_n) \end{cases}$$

建立迭代格式 $$\begin{cases} x_1^{(k+1)} = \dfrac{1}{a_{11}}(-a_{12}x_2^{(k)} - a_{13}x_3^{(k)} - \cdots - a_{1n}x_n^{(k)} + b_1) \\ x_2^{(k+1)} = \dfrac{1}{a_{22}}(-a_{21}x_1^{(k)} - a_{23}x_3^{(k)} - \cdots - a_{2n}x_n^{(k)} + b_2) \\ \cdots \\ x_n^{(k+1)} = \dfrac{1}{a_{nn}}(-a_{n1}x_1^{(k)} - a_{n2}x_2^{(k)} - \cdots - a_{n,n-1}x_{n-1}^{(k)} + b_n) \end{cases}$$

上面的迭代式称为雅可比（Jacobi）迭代格式。

用矩阵形式来表示方程组的迭代格式

设det(A)$\neq 0$，且 $a_{ii} \neq 0$

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} = \begin{bmatrix} a_{11} & 0 & 0 & 0 \\ 0 & a_{22} & \ddots & 0 \\ 0 & \ddots & \ddots & 0 \\ 0 & 0 & 0 & a_{nn} \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ a_{21} & 0 & \ddots & \ddots & 0 \\ a_{31} & a_{32} & 0 & \ddots & 0 \\ \vdots & \vdots & \vdots & \ddots & 0 \\ a_{n1} & a_{n2} & \cdots & \cdots & 0 \end{bmatrix} + \begin{bmatrix} 0 & a_{12} & a_{13} & \cdots & a_{1n} \\ 0 & 0 & a_{23} & \cdots & a_{2n} \\ 0 & \ddots & 0 & \cdots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

记$A=D+L+U$

则 $A\vec{x}=\vec{b} \Rightarrow (D+L+U)\vec{x}=\vec{b} \Rightarrow D\vec{x}=\vec{b}-(L+U)\vec{x}$

$$\Rightarrow \vec{x}=-D^{-1}(L+U)\vec{x}+D^{-1}\vec{b}$$

雅可比迭代式成为：

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

令 $B_J=-D^{-1}(L+U)=-D^{-1}(A-D)=E-D^{-1}A$

$$\vec{g}_1=D^{-1}\vec{b}$$

则得 $\vec{x}^{(k+1)}=B_J\vec{x}^{(k)}+\vec{g}_1$

(2) Jacobi迭代法的收敛条件

a.充要条件：Jacobi方法关于任意初始向量 $\vec{x}^{(0)}$都收敛的充要条件是 $\rho(B_J)<1$

b.充分条件：

（I） 若$\|B_J\|<1$则Jacobi方法关于任意初始向量$\vec{x}^{(0)}$都收敛。

（II）设系数矩阵 $A=(a_{ij})_{n\times n}$ 严格对角占优，

即 $|a_{ii}|>\sum_{j=1,i\neq j}^{n}|a_{ij}|$,（$i$=1,2,…, n）（按行）

或 $|a_{jj}|>\sum_{i=1,i\neq j}^{n}|a_{ij}|$ （j=1,2,…, n）(按列)

则Jacobi迭代法关于任意初始向量$\vec{x}^{(0)}$收敛。

证明： **由严格对角占优矩阵的定义有（按行）**

$$\sum_{j=1}^{n}|b_{ij}| = \sum_{j=1,\, j\neq i}^{n}\frac{|a_{ij}|}{|a_{ii}|} = \frac{1}{|a_{ii}|}\sum_{j=1,\, j\neq i}^{n}|a_{ij}| < 1 \quad i=1,2,\ldots,\quad n \quad \text{即} \|B_J\|_\infty < 1$$

$$\sum_{i=1}^{n}|b_{ij}| = \sum_{i=1, i\neq j}^{n}\frac{|a_{ij}|}{|a_{jj}|} = \frac{1}{|a_{jj}|}\sum_{i=1, i\neq j}^{n}|a_{ij}| < 1 \quad j=1,2,\ldots,\quad n \quad \text{即} \|B_J\|_1 < 1$$

因此Jacobi迭代法对任意初始向量$\vec{x}^{(0)}$收敛。

证毕

二、Seidel迭代法

1.迭代格式

设有简单迭代法 $\vec{x}^{(k+1)} = B\vec{x}^{(k)} + \vec{g}$, k=0,1,…

将B分解为 $B=B_1+B_2=\begin{bmatrix} 0 & & & \\ b_{21} & 0 & & \\ \vdots & \vdots & \ddots & \\ b_{n1} & b_{n2} & \cdots & 0 \end{bmatrix}+\begin{bmatrix} b_{11} & b_{12} & \cdots & b_{1n} \\ & b_{22} & \cdots & b_{2n} \\ & & \ddots & \vdots \\ & & & b_{nn} \end{bmatrix}$

则 $\vec{x}^{(k+1)} = B_1\vec{x}^{(k)} + B_2\vec{x}^{(k)} + \vec{g}$

将其修改为 $\vec{x}^{(k+1)} = B_1\vec{x}^{(k+1)} + B_2\vec{x}^{(k)} + \vec{g}$ k=0，1，…

上式称为由简单迭代法导出的Seidel迭代法。它的分量形式为

$$x_i^{(k+1)} = \sum_{j=1}^{i-1} b_{ij}x_j^{(k+1)} + \sum_{j=i}^{n} b_{ij}x_j^{(k)} + g_i \qquad i=1,2,\ldots\ldots, \ n$$

与简单迭代法相应的Seidel迭代可化为

$$\vec{x}^{(k+1)} = (I - B_1)^{-1} B_2 \vec{x}^{(k)} + (I - B_1)^{-1} \vec{g}$$

故Seidel迭代的迭代矩阵为： $B_S = (I - B_1)^{-1} B_2$

2.Seidel迭代法的收敛条件

(1)Seidel迭代收敛的充要条件 $\rho(B_s) < 1$

(2)迭代收敛的充分条件

若 $\|B\|_\infty < 1$或 $\|B\|_1 < 1$，$B = B_1 + B_2$，则Seidel迭代法关于任意的初始向量 $\vec{x}^{(0)}$收敛。

证明（了解）

证明： 仅以 $\|B\|_\infty = \max\limits_{1\le i\le n} \sum\limits_{j=1}^{n} |b_{ij}| < 1$ 为例。

由 $\begin{cases} \vec{x}^{(k+1)} = B_1\vec{x}^{(k+1)} + B_2\vec{x}^{(k)} + \vec{g} \\ \vec{x} = B\vec{x} + \vec{g} \end{cases}$ $k=0,1,\ldots$

的分量形式

$$x_i^{(k+1)} = \sum_{j=1}^{i-1} b_{ij}x_j^{(k+1)} + \sum_{j=i}^{n} b_{ij}x_j^{(k)} + g_i \quad x_i = \sum_{j=1}^{n} b_{ij}x_j + g_i$$

相减有 $$x_i^{(k+1)} - x_i = \sum_{j=1}^{i-1} b_{ij}(x_j^{(k+1)} - x_j) + \sum_{j=i}^{n} b_{ij}(x_j^{(k)} - x_j)$$

$$\left|x_i^{(k+1)} - x_i\right| \le \sum_{j=1}^{i-1} |b_{ij}|\left|x_j^{(k+1)} - x_j\right| + \sum_{j=i}^{n} |b_{ij}|\left|x_j^{(k)} - x_j\right|$$

记 $\delta_k = \max\limits_{1\le i\le n} \left|x_i^{(k)} - x_i\right|$ $\quad \beta_i = \sum\limits_{j=1}^{i-1} |b_{ij}|$ $\quad \gamma_i = \sum\limits_{j=i}^{n} |b_{ij}|$

有 $\left|x_i^{(k+1)} - x_i\right| \le \delta_{k+1}\beta_i + \delta_k\gamma_i$ 上式对 i=1,2,……， n都成立，

故有 $i = i_0$ 使 $\left|x_{i_0}^{(k+1)} - x_{i_0}\right| = \delta_{k+1}$

有 $\delta_{k+1} \le \delta_{k+1}\beta_{i_0} + \delta_k\gamma_{i_0}$

即 $$\delta_{k+1} \le \left(\frac{\gamma_{i_0}}{1-\beta_{i_0}}\right)\delta_k \le \cdots \le \left(\frac{\gamma_{i_0}}{1-\beta_{i_0}}\right)^{k+1}\delta_0$$

注意：$\beta_{i0} + \gamma_{i0} = \sum_{j=1}^{n}\left|b_{ij}\right| \le \|B\|_\infty < 1$

即 $1 - \beta_{i0} > \gamma_{i0} > 0$

所以 $0 \le \dfrac{\gamma_{i_0}}{1-\beta_{i_0}} < 1$。当 $k \to \infty$ 则 $\left(\dfrac{\gamma_{i_0}}{1-\beta_{i_0}}\right)^{k+1} \to 0$

即当$k \to \infty$ 时，则 $\delta_{k+1} = \max\limits_{1\le i\le n} \left| x_i^{(k+1)} - x_i \right| \to 0$

即 $\lim\limits_{k\to\infty} x_i^{(k)} = x_i \quad (i = 1,2,\cdots,n)$ 证毕

Remark： 当 $\|B\|_\infty < 1$ 或 $\|B\|_1 < 1$ 时，简单迭代法与相应的**Seidel**迭代法同时关于任意初始向量收敛。

3.与Jacobi方法对应的Seidel迭代格式

（1）迭代格式

其迭代式为
$$
\begin{cases}
x_1^{(k+1)} = \dfrac{1}{a_{11}}(-a_{12}x_2^{(k)} - a_{13}x_3^{(k)} - \cdots - a_{1n}x_n^{(k)} + b_1) \\
x_2^{(k+1)} = \dfrac{1}{a_{22}}(-a_{21}x_1^{(k+1)} - a_{23}x_3^{(k)} - \cdots - a_{2n}x_n^{(k)} + b_2) \\
\cdots \\
x_n^{(k+1)} = \dfrac{1}{a_{nn}}(-a_{n1}x_1^{(k+1)} - a_{n2}x_2^{(k+1)} - \cdots - a_{n.n-1}x_{n-1}^{(k+1)} + b_n)
\end{cases}
$$

矩阵形式为 $A\vec{x} = \vec{b} \Rightarrow (D + L + U)\vec{x} = \vec{b}$

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

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

因 $a_{ii} \neq 0$, 故 $(D+L)^{-1}$存在，于是Seidel迭代式为

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

令 $B_{GS} = -(D+L)^{-1}U \qquad \vec{g}_2 = (D+L)^{-1}\vec{b}$

则得 $\vec{x}^{(k+1)} = B_{GS}\vec{x}^{(k)} + \vec{g}_2$

B_{GS} **称为Gauss-Seidel迭代法的迭代矩阵。**

- **Remark1:今后若无特殊说明，凡谈到Gauss-Seidel迭代法（简称G-S迭代法），均指与Jacobi迭代法相应的Seidel迭代法。**

- **Remark2:并不是任何时候Gauss-Seidel迭代法都比Jacobi迭代法收敛快。甚至也有Jacobi法收敛而Gauss-Seidel迭代法不收敛的例子。**

（2）收敛条件

a.充要条件： G-S法关于任意初始向量$\vec{x}^{(0)}$都收敛的充要条件是： $\rho(B_{GS})<1$

b.充分条件：

①若$\|B_{GS}\|<1$则G-S方法关于任意初始向量$\vec{x}^{(0)}$都收敛。

②设系数矩阵$A=(a_{ij})$严格对角占优，则G-S方法关于任意初始向量$\vec{x}^{(0)}$收敛。

三、逐次超松弛迭代法
(SOR法-Successive Over Relaxation)

1.迭代公式

将G-S迭代格式

$$x_i^{(k+1)} = \frac{1}{a_{ii}}(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k)}) \qquad i = 1,2,\dots,n$$

改写为：

$$x_i^{(k+1)} = x_i^{(k)} + \frac{1}{a_{ii}}(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i}^{n} a_{ij} x_j^{(k)}) \quad i = 1,2,\dots,n$$

并记

$$r_i^{(k+1)} = b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i}^{n} a_{ij} x_j^{(k)} \qquad i = 1,2,\dots n$$

一般地，残量(余量) $r_i^{(k+1)} \neq 0$ 。

将残量乘以一个修正量加到 $x_i^{(k)}$ 上，作为新的结果

$$x_i^{(k+1)} = x_i^{(k)} + \frac{\omega}{a_{ii}}(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i}^{n} a_{ij} x_j^{(k)}) \qquad i = 1,2,\ldots n$$

这就是逐次超松驰迭代法（SOR方法),ω 称为松驰因子。

SOR方法的计算公式也常写为：

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

Remark：可见，SOR方法的得到的$x_i^{(k+1)}$可以看成是G-S方法的结果与 $x_i^{(k)}$ 的加权平均。

2.SOR方法的收敛性

当$\omega=1$时，SOR方法就是Seidel迭代格式。当$0<\omega<1$时，称为低松弛方法，当$\omega>1$时，称为超松弛方法。适当选取松弛因子ω的值，可以得到比G-S方法更快的收敛格式。

关于方法的收敛性有以下结论：

（1） SOR方法收敛的必要条件是$0 < \omega <2$。

（2）若系数矩阵A 对称正定，且$0 < \omega<2$，则SOR方法收敛。

（3）若系数矩阵A 严格对角占优，且$0< \omega\leq 1$，则SOR方法收敛。

Remark：能使SOR方法收敛最快的松弛因子叫做最佳松弛因子，记为ω_{opt}。

对某些特殊类型的矩阵，可以建立SOR方法最佳松弛因子理论。例如，对于具有对称正定，或严格对角占优等性质的矩阵A的线性方程组，可以建立最佳松弛因子

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

其中$\rho(J)$为解$Ax=b$的雅可比迭代法的迭代矩阵的谱半径。一般情况下确定ω_{opt}并不容易，实际计算时一般都是根据试算的情况确定ω_{opt}的一个近似值。

Remark：使用迭代法求解线代数方程组需注意的问题

1.迭代法的收敛性与初始向量的选取无关。但初始向量的选取对迭代的工作量有重大影响。

2.在使用实用误差估计式 $\left\|x^{(k+1)}-x^{(k)}\right\|<\varepsilon$ 停止迭代时，ε要选的恰当。

3.在迭代格式 $\vec{x}^{(k+1)}=B\vec{x}^{(k)}+\vec{f}$ 中，若$I-B$是病态阵，那么一般迭代法得不到好的结果。

4.如果$\|B\|=q<1$,则用此判断法比用 $\rho(B)<1$ 方便。

5.对某些方程组，有时可适当作些变形，以使得迭代法收敛。