第三章 线形方程组数值求解

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

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \vdots & \vdots \\ a_{n1} & a_{n2} & \ldots & a_{nn} \end{bmatrix} \times \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}$$

当D＝|A|≠0，方程组有解，可用Cramer法则求解。

$$x_i = \frac{D_i}{D}, D = \det(A), D_i = \det\begin{pmatrix} a_{11} & \cdots & a_{1i-1} & b_1 & a_{1i+1} & \cdots & a_{1n} \\ \vdots & & \vdots & \vdots & \vdots & & \vdots \\ a_{n1} & \cdots & a_{ni-1} & b_n & a_{ni+1} & \cdots & a_{nn} \end{pmatrix}$$

用行列式解法求解线性方程组:

n阶方程组，要计算$n+1$个n阶行列式的值，

总共需要做$n!\,(n-1)\,(n+1)$次乘法运算。

n=100?

9.33168828223997582664310689323812e+161次

美国Frontier 2.68517145451146197525275377192224e+136

太湖之光 3.18178380953938827605218780285886e+137

年！

线性方程组的数值解法

①直接法：

准确，可靠，理论上得到的解是精确的

- *Gauss消去法；*
- *Gauss-Jordan消去法；*
- *矩阵分解法等。*

②迭代法：速度快，但有误差

- *Jocobi迭代法；*
- *Seidel迭代法；*
- *超松弛(Sor)迭代法。*

§3.1 求解线形方程组的消元法

消(元)去法是求解线形方程组

Ax=b　(3)

的一种直接方法。尽管它比较古老，但它具有演算步骤，推算公式都系统化的特点。因此,它至今仍然是求解方程组的一种有效的方法。

下面有2种方程的解我们可以直接求出：

***n*次运算**

①

$$A = diag(a_{11}, a_{22}, \cdots, a_{nn}) \Rightarrow x_i = \frac{b_i}{a_{ii}}, i = 1, \cdots, n$$

$$A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 5 \end{bmatrix} \qquad b = \begin{bmatrix} -1 \\ 6 \\ -15 \end{bmatrix}$$

$$\begin{aligned} x_1 \qquad\qquad &= -1 \\ 3x_2 \qquad &= 6 \\ 5x_3 &= -15 \end{aligned}$$

$$x_1 = -1$$

$$x_2 = \frac{6}{3} = 2$$

$$x_3 = \frac{-15}{5} = -3$$

②三角形方程组的解法

三角形方程组包括上三角形方程组和下三角形方程组，是是最简单的线性方程组之一。上三角方程组的一般形式是：

$$
\left\{
\begin{aligned}
a_{11}x_1 + a_{12}x_2 + \cdots\cdots + a_{1n}x_n &= b_1 \\
a_{22}x_2 + \cdots\cdots + a_{2n}x_n &= b_2 \\
\cdots\cdots\cdots\cdots &= \cdots \\
a_{n-1n-1}x_{n-1} + a_{n-1n}x_n &= b_{n-1} \\
a_{nn}x_n &= b_n
\end{aligned}
\right.
$$

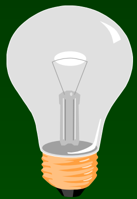
其中$a_{ii} \neq 0 \quad (i = 1,2,\ldots\ldots,n)$

$$A=\begin{pmatrix} l_{11} & & & \\ l_{21} & l_{22} & & \\ \vdots & \vdots & \ddots & \\ l_{n1} & l_{n2} & \cdots & l_{nn} \end{pmatrix} \Rightarrow x_i = \frac{b_i - \sum_{j=1}^{i-1} l_{ij} x_j}{l_{ii}}, i=1,\cdots,n$$

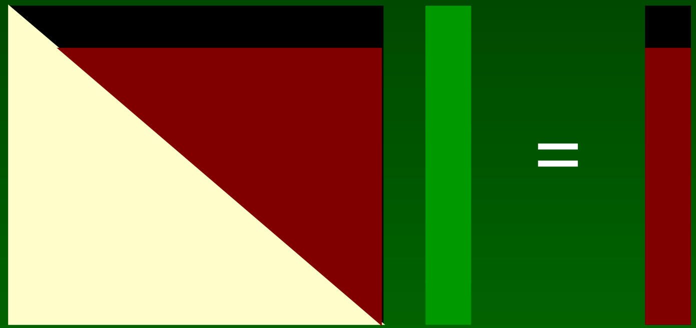
$$A=\begin{pmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ & u_{22} & \cdots & u_{2n} \\ & & \ddots & \vdots \\ & & & u_{nn} \end{pmatrix} \Rightarrow x_i = \frac{b_i - \sum_{j=i+1}^{n} u_{ij} x_j}{u_{ii}}, i=n,\cdots,1$$

（n+1）n/2
次运算

➢ 高斯消元法：



思路 首先将*A*化为上三角阵，再回代求解 。



$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} & b_1 \\ a_{21} & a_{22} & \cdots & a_{2n} & b_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} & b_n \end{pmatrix} \Longrightarrow \begin{pmatrix} a_{11}^{(1)} & a_{12}^{(1)} & a_{13}^{(1)} & \cdots & a_{1n}^{(1)} & b_1^{(1)} \\ 0 & a_{22}^{(2)} & a_{23}^{(2)} & \cdots & a_{2n}^{(2)} & b_2^{(2)} \\ 0 & 0 & a_{33}^{(3)} & \cdots & a_{3n}^{(3)} & b_3^{(3)} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & a_{nn}^{(n)} & b_n^{(n)} \end{pmatrix}$$

对方程组，作如下的变换，解不变

①交换两个方程的次序

②一个方程的两边同时乘以一个非0的数

③一个方程的两边同时乘以一个非0数，加到另一个方程

因此，对应的对增广矩阵***(A，b)***，作如下的变换，解不变

①交换矩阵的两行

②某一行乘以一个非0的数

③某一个乘以一个非0数，加到另一行

消元法就是对增广矩阵作上述行的变换，变为我们已知的2种类型之一，而后求根。

步骤如下：

第一步：

$$\text{第}1\text{行}\times\frac{-a_{i1}}{a_{11}}+\text{第}\,i\,\text{行}, i=2,\cdots,n$$

$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} & b_1 \\ a_{21} & a_{22} & \cdots & a_{2n} & b_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} & b_n \end{pmatrix} \Longrightarrow \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} & b_1 \\ 0 & a_{22}^{(2)} & \cdots & a_{2n}^{(2)} & b_2^{(2)} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & a_{n2}^{(2)} & \cdots & a_{nn}^{(2)} & b_n^{(2)} \end{pmatrix}$$

运算量： *(n-1)*(1+n)*

第二步：

$$第2行\times\frac{-a_{i2}^{(2)}}{a_{22}^{(2)}}+第i行,\ i=3,\cdots,n$$

$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} & b_1 \\ 0 & a_{22}^{(2)} & \cdots & a_{2n}^{(2)} & b_2^{(2)} \\ \vdots & \boxed{\begin{matrix}\vdots \\ a_{n2}^{(2)}\end{matrix}} & \ddots & \vdots & \vdots \\ 0 & & \cdots & a_{nn}^{(2)} & b_n^{(2)} \end{pmatrix} \Rightarrow \begin{pmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} & b_1 \\ 0 & a_{22}^{(2)} & a_{23}^{(2)} & \cdots & a_{2n}^{(2)} & b_2^{(2)} \\ 0 & 0 & a_{33}^{(3)} & \cdots & a_{3n}^{(3)} & b_3^{(3)} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & a_{n3}^{(3)} & \cdots & a_{nn}^{(3)} & b_n^{(3)} \end{pmatrix}$$

运算量： ***(n-2)*(1+n-1)=(n-2)n***

类似的做下去，我们有：

第k步：

$$第k行 \times \frac{-a_{ik}^{(k)}}{a_{kk}^{(k)}} + 第i行, i = k+1, \cdots, n$$

运算量： $(n-k)*(1+n-k+1)=(n-k)(n-k+2)$

$n-1$步以后，我们可以得到变换后的矩阵为：

$$\begin{pmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} & b_1 \\ 0 & a_{22}^{(2)} & a_{23}^{(2)} & \cdots & a_{2n}^{(2)} & b_2^{(2)} \\ 0 & 0 & a_{33}^{(3)} & \cdots & a_{3n}^{(3)} & b_3^{(3)} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & a_{nn}^{(n)} & b_n^{(n)} \end{pmatrix}$$

因此，总的运算量为：

$$\sum_{k=1}^{n-1}(n-k)(n-k+2)$$

加上 解上述上三角阵的运算量***(n+1)n/2***，总共为：

$$\frac{n^3}{3}+n^2-\frac{n}{3}=O(n^3)$$

n=100?

343300次!

例题分析

例1：用消去法解方程组

$$\begin{cases} x_1 + x_2 + x_3 = 6; \\ 4x_2 - x_3 = 5; \\ 2x_1 - 2x_2 + x_3 = 1. \end{cases}$$

解：用增广矩阵表示求解过程

$$(A|b) = \begin{bmatrix} 1 & 1 & 1 & 6 \\ 0 & 4 & -1 & 5 \\ 2 & -2 & 1 & 1 \end{bmatrix} \to \begin{bmatrix} 1 & 1 & 1 & 6 \\ 0 & 4 & -1 & 5 \\ 0 & -4 & -1 & -11 \end{bmatrix}$$

$$\to \begin{bmatrix} 1 & 1 & 1 & 6 \\ 0 & 4 & -1 & 5 \\ 0 & 0 & -2 & -6 \end{bmatrix}$$

$$(-2)\times r_1 + r_3 \to r_3, \quad r_2 + r_3 \to r_3$$

Gaussian Elimination Algorithm

form $\underline{A} = [A\ b]$
for $i = 1 \ldots n - 1$
  for $k = i + 1 \ldots n$
    for $j = i \ldots n + 1$
      $\underline{a}_{kj} = \underline{a}_{kj} - \underline{a}_{ij}\,(\underline{a}_{ki} / \underline{a}_{ii})$
    end
  end
end

Note that there is nothing "wrong" with this system. A is full rank. The solution exists and is unique.

```
function x = GEshow(A,b,ptol)
% GEshow  Show steps in Gauss elimination and back substitution
%         No pivoting is used.
% Synopsis:  x = GEshow(A,b)
%            x = GEshow(A,b,ptol)
% Input:     A,b  = coefficient matrix and right hand side vector
%            ptol = (optional) tolerance for detection of zero pivot
%                   Default:  ptol = 50*eps
% Output:    x = solution vector, if solution exists
if nargin<3, ptol = 50*eps;  end
[m,n] = size(A);
if m~=n,  error('A matrix needs to be square');  end
nb = n+1;   Ab = [A b];    %  Augmented system
fprintf('\nBegin forward elmination with Augmented system:\n');
disp(Ab);
```

```
% --- Elimination
for i = 1:n-1              %  loop over pivot rows
  pivot = Ab(i,i);
  if abs(pivot)<ptol, error('zero pivot encountered');  end
  for k = i+1:n             %  k = index of next row to be eliminated
     Ab(k,i:nb) = Ab(k,i:nb) - (Ab(k,i)/pivot)*Ab(i,i:nb);
  end
  fprintf('\nAfter elimination in column %d with pivot = %f\n',i,pivot);
  disp(Ab);
end
% --- Back substitution
x = zeros(n,1);           %  preallocate memory for and initialize x
x(n) = Ab(n,nb)/Ab(n,n);
for i=n-1:-1:1
  x(i) = (Ab(i,nb) - Ab(i,i+1:n)*x(i+1:n))/Ab(i,i);
end
```

注意到，计算过程中$a_{kk}^{(k)}$处在被除的位置，因此整个计算过程要保证它不为0，所以，Gauss消元法的可行条件为：

$$a_{kk}^{(k)} \neq 0$$

就是要求A的所有顺序主子式均不为0，即

$$\det\begin{pmatrix} a_{11} & \cdots & a_{1i} \\ \vdots & \ddots & \vdots \\ a_{i1} & \cdots & a_{ii} \end{pmatrix} \neq 0, i = 1, \cdots, n$$

例：单精度解方程组

$$\begin{cases} 10^{-9}x_1 + x_2 = 1 \\ x_1 + x_2 = 2 \end{cases}$$

/* 精确解为 $x_1 = \dfrac{1}{1-10^{-9}} = 1.\overbrace{00\ldots0}^{8个}100\ldots$ 和 $x_2 = 2 - x_1 = 0.\overbrace{99\ldots9}^{8个}899\ldots$ */

用Gauss消元法计算：

$$m_{21} = a_{21}/a_{11} = 10^9$$

$$a_{22} = 1 - m_{21}\times 1 = 0.\overbrace{0\ldots0}^{8个}1 \times 10^9 - 10^9 \doteq -10^9$$

$$b_2 = 2 - m_{21}\times 1 \doteq -10^9$$

$$\Rightarrow \begin{bmatrix} 10^{-9} & 1 & \\ 0 & -10^9 & -10^9 \end{bmatrix}$$

小主元可能导致计算失败。

$$\Rightarrow \quad x_2 = 1, \quad x_1 = 0$$

列主元消元法

在Gauss消元第k步之前，
做如下的事情：

若 $\max\limits_{k\le i\le n}|a_{ik}^{(k)}|=|a_{jk}^{(k)}|$ 交换k行和j行

例：

$$\begin{bmatrix}10^{-9} & 1 & 1\\ 1 & 1 & 2\end{bmatrix}\Rightarrow\begin{bmatrix}1 & 1 & 2\\ 10^{-9} & 1 & 1\end{bmatrix}\Rightarrow\begin{bmatrix}1 & 1 & 2\\ 0 & 1 & 1\end{bmatrix}$$

$$\Rightarrow\quad x_2=1,\quad x_1=1$$

行的交换，不改变方程组的解，同时又有效地克服了Gauss消元地缺陷。

Gauss Elimination with Partial Pivoting

form $\tilde{A} = [A\ b]$
for $i = 1 \ldots n - 1$
  find i_p such that
  $| \tilde{a}_{ipi} | = \max(|\tilde{a}_{ki}|)$ for $k = i \ldots n$
  exchange row i_p with row i
  for $k = i + 1 \ldots n$
    for $j = i \ldots n + 1$
      $\tilde{a}_{k,j} = \tilde{a}_{k,j} - (\tilde{a}_{k,i} / \tilde{a}_{i,i})\, \tilde{a}_{i,j}$
    end
  end
end

Gauss-Jordan消元法

将在Gauss消元第k步，变为

$$\text{第k行} \times \frac{-a_{ik}^{(k)}}{a_{kk}^{(k)}} + \text{第}\, i\,\text{行}, i = 1, \cdots, k-1, k+1, \cdots, n$$

将该行上三角的部分也变为0

最后变为一个对角阵。

它运算次数比Gauss消元多。使用于计算多个系数一样地方程组，如

$$AX = B$$

X，B均为矩阵

Gauss消去法的矩阵表示

$$L_1 = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ -m_{21} & 1 & 0 & \dots & 0 \\ -m_{31} & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & 0 \\ -m_{n1} & 0 & 0 & \dots & 1 \end{bmatrix} \quad A^{(1)} = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} & b_1 \\ a_{21} & a_{22} & \cdots & a_{2n} & b_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} & b_n \end{pmatrix}$$

$$m_{i1} = \frac{a_{i1}^{(1)}}{a_{11}^{(1)}} \qquad i = 2,3,\dots,n$$

$$A^{(2)} = L_1 A^{(1)} \qquad\qquad b^{(2)} = L_1 b^{(1)}$$

Gauss消去法的矩阵表示

$$A^{(2)}=\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} & b_1 \\ 0 & a_{22}^{(2)} & \cdots & a_{2n}^{(2)} & b_2^{(2)} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & a_{n2}^{(2)} & \cdots & a_{nn}^{(2)} & b_n^{(2)} \end{pmatrix} \qquad L_2=\begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 0 & \dots & 0 \\ 0 & -m_{32} & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & 0 \\ 0 & -m_{n2} & 0 & \dots & 1 \end{bmatrix}$$

$$m_{i2}=\frac{a_{i2}^{(2)}}{a_{22}^{(2)}} \qquad i=3,4,\dots,n$$

$$A^{(3)}=L_2A^{(2)}=L_2L_1A^{(1)} \qquad b^{(3)}=L_2L_1b^{(1)}$$

每一步消去过程相当于左乘初等变换矩阵L_k

Gauss 消去法的矩阵表示

$$L_k = \begin{pmatrix} 1 & & & & & \\ & \ddots & & & & \\ & & 1 & & & \\ & & -m_{k+1,k} & 1 & & \\ & & \vdots & & \ddots & \\ & & -m_{n,k} & & & 1 \end{pmatrix}$$

$$L_k \cdot \left(A^{(k)}, b^{(k)}\right) = \left(A^{(k+1)}, b^{(k+1)}\right)$$

$$L_{n-1} \cdot \cdots \cdot L_2 \cdot L_1 \cdot \left(A^{(1)}, b^{(1)}\right) = \left(A^{(n)}, b^{(n)}\right)$$

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

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

L_i 与 L_i^{-1}

$$L_i = \begin{pmatrix} 1 & & & & & \\ 0 & 1 & & & & \\ \vdots & & 1 & & & \\ 0 & \cdots & -m_{i+1,i} & 1 & & \\ \vdots & & \vdots & \vdots & \ddots & \\ 0 & & -m_{ni} & & & 1 \end{pmatrix}$$

i列

$$L_i^{-1} = \begin{pmatrix} 1 & & & & & & \\ 0 & 1 & & & & & \\ \vdots & & \ddots & & & & \\ \vdots & & & 1 & & & \\ 0 & & & m_{i+1,i} & 1 & & \\ \vdots & & & \vdots & & \ddots & \\ 0 & & & m_{ni} & & & 1 \end{pmatrix}$$

i列

LU形式

$$A^{(1)} = L_1^{-1} L_2^{-1} \dots L_{n-1}^{-1} A^{(n)} = L A^{(n)} = L U$$

$$\begin{pmatrix} 1 & & \\ m_{21} & 1 & \\ m_{31} & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & & \\ 0 & 1 & \\ 0 & m_{22} & 1 \end{pmatrix} = \begin{pmatrix} 1 & & \\ m_{21} & 1 & \\ m_{31} & m_{22} & 1 \end{pmatrix}$$

$$L = \begin{pmatrix} 1 & & & \\ m_{21} & 1 & & \\ \vdots & & \ddots & \\ m_{n1} & m_{n2} & \cdots & 1 \end{pmatrix}$$

定义1. 不带行交换的Gauss 消去法的消元过程,产生一个单位下三角矩阵L和一个上三角矩阵U,即

$$A = LU$$

该过程称之为 矩阵A的LU分解.

由上述分析不难得到

k阶顺序主子式

$$A=\begin{pmatrix} a_{11} & \cdots & a_{1k} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots & & \vdots \\ a_{k1} & \cdots & a_{kk} & \cdots & a_{kn} \\ \vdots & & \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nk} & \cdots & a_{nn} \end{pmatrix} = \begin{pmatrix} 1 & & & & \\ \vdots & \ddots & & & \\ m_{k1} & \cdots & 1 & & \\ \vdots & & \vdots & \ddots & \\ m_{n1} & \cdots & m_{nk} & \cdots & 1 \end{pmatrix} \cdot \begin{pmatrix} a_{11}^{(1)} & \cdots & a_{1k}^{(1)} & \cdots & a_{1n}^{(1)} \\ & \ddots & \vdots & & \vdots \\ & & a_{kk}^{(k)} & \cdots & a_{kn}^{(k)} \\ & & & \ddots & \vdots \\ & & & & a_{nn}^{(n)} \end{pmatrix}$$

$$A_k = L_k U_k \qquad \det A_k = \det U_k = \prod_{i=1}^{k} a_{ii}^{(i)}$$

§ 2.2　直接三角分解法

一、基本的三角分解法(Doolittle法)

若n阶方阵$A=(a_{ij})_{n\times n}$的顺序主子式$D_k\neq 0,\ k=1,2,\cdots,n$

则由上节可知,A的LU分解$A=LU$存在且唯一,即

$$A=\begin{pmatrix} a_{11} & \cdots & a_{1k} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots & & \vdots \\ a_{k1} & \cdots & a_{kk} & \cdots & a_{kn} \\ \vdots & & \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nk} & \cdots & a_{nn} \end{pmatrix}=\begin{pmatrix} 1 & & & & \\ \vdots & \ddots & & & \\ m_{k1} & \cdots & 1 & & \\ \vdots & & \vdots & \ddots & \\ m_{n1} & \cdots & m_{nk} & \cdots & 1 \end{pmatrix}\cdot\begin{pmatrix} a_{11}^{(1)} & \cdots & a_{1k}^{(1)} & \cdots & a_{1n}^{(1)} \\ & \ddots & \vdots & & \vdots \\ & & a_{kk}^{(k)} & \cdots & a_{kn}^{(k)} \\ & & & \ddots & \vdots \\ & & & & a_{nn}^{(n)} \end{pmatrix}$$

$$=LU$$

LU 分解求解线性方程组

$$AX=b \quad \rightarrow \quad LY=b, \quad UX=Y$$

$$LY=\begin{pmatrix} 1 & & & & \\ l_{21} & 1 & & & \\ l_{31} & l_{32} & 1 & & \\ & \cdots & \cdots & & \\ l_{n1} & l_{n2} & & l_{n(n-1)} & 1 \end{pmatrix}\begin{pmatrix} y_1 \\ y_2 \\ \\ \\ y_n \end{pmatrix}=\begin{pmatrix} b_1 \\ b_2 \\ \\ \\ b_n \end{pmatrix}=b$$

$$\mathrm{UX}=\begin{pmatrix} u_{11} & u_{12} & \cdots & u_{1\mathrm{n}} \\ & u_{22} & \cdots & u_{2\mathrm{n}} \\ & & & \\ & & \cdots \quad \cdots & \cdots \\ & & & u_{\mathrm{nn}} \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \\ \\ \\ x_n \end{pmatrix}=\begin{pmatrix} y_1 \\ y_2 \\ \\ \\ y_n \end{pmatrix}=Y$$

一般计算公式

$$A=\begin{pmatrix} a_{11} & \cdots & a_{1k} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots & & \vdots \\ a_{k1} & \cdots & a_{kk} & \cdots & a_{kn} \\ \vdots & & \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nk} & \cdots & a_{nn} \end{pmatrix} = \begin{pmatrix} 1 & & & & \\ \vdots & \ddots & & & \\ l_{k1} & \cdots & 1 & & \\ \vdots & & \vdots & \ddots & \\ l_{n1} & \cdots & l_{nk} & \cdots & 1 \end{pmatrix} \cdot \begin{pmatrix} u_{11} & \cdots & u_{1k} & \cdots & u_{1n} \\ & \ddots & \vdots & & \vdots \\ & & u_{kk} & \cdots & u_{kn} \\ & & & \ddots & \vdots \\ & & & & u_{nn} \end{pmatrix}$$

$$u_{1j}=a_{1j}\ ,\quad j=1,\cdots,n$$

$$l_{i1}=a_{i1}/u_{11}\ ,\quad i=2,\cdots,n$$

对 $k=2,\ 3,\ \cdots,\ n$ 计算

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

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

计算量与 Gauss 消去法相同.

例1. 用LU分解法解方程组

$$\begin{pmatrix} 2 & 10 & 0 & -3 \\ -3 & -4 & -12 & 13 \\ 1 & 2 & 3 & -4 \\ 4 & 14 & 9 & -13 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 10 \\ 5 \\ -2 \\ 7 \end{pmatrix}$$

$$u_{1j} = a_{1j}$$

$$l_{i1} = \frac{a_{i1}}{u_{11}}$$

$$A = \left(\begin{array}{cccc|c} 2 & 10 & 0 & -3 & 10 \\ -3 & -4 & -12 & 13 & 5 \\ 1 & 2 & 3 & -4 & -2 \\ 4 & 14 & 9 & -13 & 7 \end{array}\right) \xrightarrow{r=1} \left(\begin{array}{c|ccc|c} 2 & 10 & 0 & -3 & 10 \\ \hline -3 & -4 & -12 & 13 & 5 \\ 1 & 2 & 3 & -4 & -2 \\ 4 & 14 & 9 & -13 & 7 \end{array}\right)$$

$$\xrightarrow{r=1}\left(\begin{array}{cccc|c} 2 & 10 & 0 & -3 & 10 \\ -\frac{3}{2} & -4 & -12 & 13 & 5 \\ \frac{1}{2} & 2 & 3 & -4 & -2 \\ 2 & 14 & 9 & -13 & 7 \end{array}\right)$$

$$u_{rj} = a_{rj} - \sum_{k=1}^{r-1} l_{rk} u_{kj}$$

$$\xrightarrow{r=2}\left(\begin{array}{cccc|c} 2 & 10 & 0 & -3 & 10 \\ -\frac{3}{2} & 11 & -12 & \frac{17}{2} & 20 \\ \frac{1}{2} & -\frac{3}{11} & 3 & -4 & -2 \\ 2 & -\frac{6}{11} & 9 & -13 & 7 \end{array}\right)$$

$$l_{ir} = \frac{a_{ir} - \sum_{k=1}^{r-1} l_{ik} u_{kr}}{u_{rr}}$$

$$\xrightarrow{r=2}\left(\begin{array}{cccc|c} 2 & 10 & 0 & -3 & 10 \\ -\dfrac{3}{2} & 11 & -12 & \dfrac{17}{2} & 20 \\ \dfrac{1}{2} & -\dfrac{3}{11} & 3 & -4 & -2 \\ 2 & -\dfrac{6}{11} & 9 & -13 & 7 \end{array}\right)$$

$$\xrightarrow{r=3}\left(\begin{array}{cccc|c} 2 & 10 & 0 & -3 & 10 \\ -\dfrac{3}{2} & 11 & -12 & \dfrac{17}{2} & 20 \\ \dfrac{1}{2} & -\dfrac{3}{11} & -\dfrac{3}{11} & -\dfrac{2}{11} & -\dfrac{17}{11} \\ 2 & -\dfrac{6}{11} & -9 & -13 & 7 \end{array}\right)$$

$$u_{rj} = a_{rj} - \sum_{k=1}^{r-1} l_{rk} u_{kj}$$

$$l_{ir} = \frac{a_{ir} - \sum_{k=1}^{r-1} l_{ik} u_{kr}}{u_{rr}}$$

$$U$$

$$\xrightarrow{r=4}\left(\begin{array}{cccc|c} 2 & 10 & 0 & -3 & 10 \\ -\frac{3}{2} & 11 & -12 & \frac{17}{2} & 20 \\ \frac{1}{2} & -\frac{3}{11} & -\frac{3}{11} & -\frac{2}{11} & -\frac{17}{11} \\ 2 & -\frac{6}{11} & -9 & -4 & -16 \end{array}\right) \quad y$$

$$L$$

$$\xrightarrow{解Ux=y}\left(\begin{array}{cccc|c} 2 & 10 & 0 & -3 & 1 \\ -\frac{3}{2} & 11 & -12 & \frac{17}{2} & 2 \\ \frac{1}{2} & -\frac{3}{11} & -\frac{3}{11} & -\frac{2}{11} & 3 \\ 2 & -\frac{6}{11} & -9 & -4 & 4 \end{array}\right) \quad x$$

所以

$$x=\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix}=\begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \end{pmatrix}$$

Given A

For i=1…n-1

For k=i+1…n

$a_{ki}=a_{ki}/a_{ii}$

for j=i+1…n

$a_{kj}=a_{kj}-a_{ki}a_{ij}$

end

End

End

$$L=\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix}$$

$$\rightarrow \begin{bmatrix} u_{11} & u_{12} & u_{13} & u_{14} \\ l_{21} & u_{22} & u_{23} & u_{24} \\ l_{31} & a^{(2)}_{32} & a^{(2)}_{33} & a^{(2)}_{34} \\ l_{41} & a^{(2)}_{42} & a^{(2)}_{43} & a^{(2)}_{44} \end{bmatrix}$$

$$\rightarrow \begin{bmatrix} u_{11} & u_{12} & u_{13} & u_{14} \\ l_{21} & u_{22} & u_{23} & u_{24} \\ l_{31} & l_{32} & u_{33} & u_{34} \\ l_{41} & l_{42} & a^{(3)}_{43} & a^{(3)}_{44} \end{bmatrix}$$

$$\rightarrow \begin{bmatrix} u_{11} & u_{12} & u_{13} & u_{14} \\ l_{21} & u_{22} & u_{23} & u_{24} \\ l_{31} & l_{32} & u_{33} & u_{34} \\ l_{41} & l_{42} & l_{43} & u_{44} \end{bmatrix}$$

LU分解的matlab程序

```
function [L,U] = luNopiv(A,ptol)
% luNopiv  LU factorization without pivoting
% Synopsis:  [L,U] = luNopiv(A)
%            [L,U] = luNopiv(A,ptol)
% Input:     A    = coefficient matrix
% ptol = (optional) tolerance for detection of zero pivot
%                 Default:  ptol = 50*eps
% Output:    L,U = lower triangular matrix, L, and upper triangular
%                 matrix, U, such that A = L*U
if nargin<3, ptol = 50*eps;  end
%  Default tolerance for zero pivot
[m,n] = size(A);
if m~=n,  error('A matrix needs to be square');  end
```

LU分解的matlab程序

```
for i = 1:n-1                          %  loop over pivot rows
  pivot = A(i,i);
  if abs(pivot)<ptol, error('zero pivot encountered');  end
  for k = i+1:n                        %  row k is eliminated next
     A(k,i) = A(k,i)/pivot;      %  compute and store multiplier
     A(k,i+1:n) = A(k,i+1:n) - A(k,i)*A(i,i+1:n);
            %  row ops to eliminate A(k,i)
  end
end
L = eye(size(A)) + tril(A,-1);  %  extract L and U
U = triu(A);
```

高斯消元法	LU分解法
处理方程对应增广矩阵	处理方程对应系数矩阵

- **在LU分解法中，如果只使用系数矩阵，需要记录对系数矩阵的行操作，以便在求解的过程中对向量B做同样的操作。**

三、平方根法

若 $A^T = A$，则称为对称矩阵

若A正定矩阵， 则有

(1)对$\forall x \neq \theta$, $x^T A x > 0$;

(2)A非奇异,A^{-1}正定,且$a_{ii} > 0(i = 1,2,\cdots n)$;

(3)A的顺序主子式均大于0;

(4)A的特征值都大于0.

$$A=\begin{bmatrix}1 & & & & \\ l_{21} & 1 & & & \\ l_{31} & l_{32} & 1 & & \\ \vdots & \vdots & \vdots & \ddots & \\ l_{n1} & l_{n2} & \cdots & l_{nn-1} & 1\end{bmatrix}\begin{bmatrix}u_{11} & u_{12} & u_{13} & \cdots & u_{1n} \\ & u_{22} & u_{23} & \cdots & u_{2n} \\ & & u_{33} & \cdots & u_{3n} \\ & & & \ddots & \cdots \\ & & & & u_{nn}\end{bmatrix}$$

记 A_k（$1\le k\le n$）为A的 k 阶顺序主子阵，则det（A_k）为A的k阶顺序主子式。由上式，利用矩阵分块运算规则，容易验证

$$\det(A_k)=u_{11}u_{22}\dots u_{kk}$$

那么由 det（A_k）＞0，可知 $u_{kk}>0$，$k=1$，2，…，n

这时，将上面的矩阵表示为：

$$A=\begin{bmatrix}1 &&&& \\ l_{21} & 1 &&& \\ l_{31} & l_{32} & 1 && \\ \vdots & \vdots & \vdots & \ddots & \\ l_{n1} & l_{n2} & \cdots & l_{nn-1} & 1\end{bmatrix}\begin{bmatrix}u_{11} & u_{12} & u_{13} & \cdots & u_{1n} \\ & u_{22} & u_{23} & \cdots & u_{2n} \\ && u_{33} & \cdots & u_{3n} \\ &&& \ddots & \cdots \\ &&&& u_{nn}\end{bmatrix}$$

$$=\begin{bmatrix}1 &&&& \\ l_{21} & 1 &&& \\ l_{31} & l_{32} & 1 && \\ \vdots & \vdots & \vdots & \ddots & \\ l_{n1} & l_{n2} & \cdots & l_{nn-1} & 1\end{bmatrix}\begin{bmatrix}u_{11} &&&& \\ & u_{22} &&& \\ && u_{33} && \\ &&& \ddots & \\ &&&& u_{nn}\end{bmatrix}\begin{bmatrix}1 & \frac{u_{12}}{u_{11}} & \frac{u_{13}}{u_{11}} & \cdots & \frac{u_{1n}}{u_{11}} \\ & 1 & \frac{u_{23}}{u_{22}} & \cdots & \frac{u_{2n}}{u_{2}} \\ && 1 & \cdots & \frac{u_{3n}}{u_{33}} \\ &&& \ddots & \cdots \\ &&&& 1\end{bmatrix}$$

$= LDM$

即： $A=LDM$ ， 其中 $DM=U$， $M=D^{-1}U$。

当$A=A^T$为对称矩阵时，根据 $A=LDM$, 得到

$$A^T=M^T D L^T$$

再根据矩阵三角分解的唯一性,可知 $M=L^T$。于是

$$A=LDL^T$$

令 $D^{\frac{1}{2}}=\mathrm{diag}(\sqrt{u_{11}},\sqrt{u_{22}},\cdots,\sqrt{u_{nn}})$ $\quad G=LD^{\frac{1}{2}}$

则有 $A=LD^{\frac{1}{2}}D^{\frac{1}{2}}L^T=LD^{\frac{1}{2}}(LD^{\frac{1}{2}})^T=GG^T$

$$=\begin{bmatrix} g_{11} & & & & \\ g_{21} & g_{22} & & & \\ g_{31} & g_{32} & g_{33} & & \\ \vdots & \vdots & \vdots & \ddots & \\ g_{n1} & g_{n2} & \cdots & g_{nn-1} & g_{nn} \end{bmatrix}\begin{bmatrix} g_{11} & g_{12} & g_{13} & \cdots & g_{1n} \\ & g_{22} & g_{23} & \cdots & g_{2n} \\ & & g_{33} & \cdots & g_{3n} \\ & & & \ddots & \cdots \\ & & & & g_{nn} \end{bmatrix}$$

如果对称正定矩阵A具有如下分解 $A=GG^T$，其中G为下三角矩阵，则称其为对称正定矩阵的 **Cholesky**（乔列斯基）分解。

为表示方便，可以记

$$A = LL^T = \begin{bmatrix} l_{11} & & & & \\ l_{21} & l_{22} & & & \\ l_{31} & l_{32} & l_{33} & & \\ \vdots & \vdots & \vdots & \ddots & \\ l_{n1} & l_{n2} & \cdots & l_{nn-1} & l_{nn} \end{bmatrix} \begin{bmatrix} l_{11} & l_{12} & l_{13} & \cdots & l_{1n} \\ & l_{22} & l_{23} & \cdots & l_{2n} \\ & & l_{33} & \cdots & l_{3n} \\ & & & \ddots & \cdots \\ & & & & l_{nn} \end{bmatrix}$$

给定对称正定方程组 $Ax=b$，对 A 进行 **Cholesky** 分解$A=LL^T$，则原方程组等价于 $LL^Tx=b$

即： $LL^Tx=b$，等价于

$$Ly=b$$
$$L^Tx=y$$

解此方程组即可得到原方程组的解x，这就是求解方程组的平方根法。

下面，我们通过比较矩阵的对应元素给出对称正定矩阵的平方根分解法。已知

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} = \begin{bmatrix} l_{11} & & & & \\ l_{21} & l_{22} & & & \\ l_{31} & l_{32} & l_{33} & & \\ \vdots & \vdots & \vdots & \ddots & \\ l_{n1} & l_{n2} & \cdots & l_{nn-1} & l_{nn} \end{bmatrix} \begin{bmatrix} l_{11} & l_{12} & l_{13} & \cdots & l_{1n} \\ & l_{22} & l_{23} & \cdots & l_{2n} \\ & & l_{33} & \cdots & l_{3n} \\ & & & \ddots & \cdots \\ & & & & l_{nn} \end{bmatrix}$$

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} = \begin{bmatrix} l_{11} & & & & \\ l_{21} & l_{22} & & & \\ l_{31} & l_{32} & l_{33} & & \\ \vdots & \vdots & \vdots & \ddots & \\ l_{n1} & l_{n2} & \cdots & l_{nn-1} & l_{nn} \end{bmatrix} \begin{bmatrix} l_{11} & l_{12} & l_{13} & \cdots & l_{1n} \\ & l_{22} & l_{23} & \cdots & l_{2n} \\ & & l_{33} & \cdots & l_{3n} \\ & & & \ddots & \cdots \\ & & & & l_{nn} \end{bmatrix}$$

比较对应元素：

$$a_{ij} = \left(l_{i1}, l_{i2}, \cdots, l_{ii}, 0, \cdots 0 \right) \begin{bmatrix} l_{1j} \\ l_{2j} \\ \vdots \\ l_{jj} \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

$$a_{ij} = \left(l_{i1}, l_{i2}, \cdots, l_{ii}, 0, \cdots 0\right) \begin{bmatrix} l_{j1} \\ l_{j2} \\ \vdots \\ l_{jj} \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

当 $i = j$ 时 $a_{jj} = l_{j1}^2 + l_{j2}^2 + \cdots + l_{jj}^2$

当 $j \le i$ 时 $a_{ij} = l_{i1} l_{j1} + l_{i2} l_{j2} + \cdots + l_{ij} l_{jj}$

解得 $$l_{jj} = (a_{jj} - \sum_{k=1}^{j-1} l_{jk}^2)^{\frac{1}{2}}, \quad j = 1, 2, \cdots, n$$

$$l_{ij} = (a_{ij} - \sum_{k=1}^{j-1} l_{ik} l_{jk}) / l_{jj}, \quad i = j, j+1, \cdots, n$$

关于方程组 $Ax=b$ ，如果对系数矩阵进行了平方根分解 $A=LL^T$，则将方程组化为： $Ly=b$, $L^Tx=y$

$$\begin{bmatrix} l_{11} & & & & \\ l_{21} & l_{22} & & & \\ l_{31} & l_{32} & l_{33} & & \\ \vdots & \vdots & \vdots & \ddots & \\ l_{n1} & l_{n2} & l_{n3} & \cdots & l_{nn} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ b_n \end{bmatrix} \qquad \begin{bmatrix} l_{11} & l_{21} & l_{31} & \vdots & l_{n1} \\ & l_{22} & l_{32} & \vdots & l_{n2} \\ & & l_{33} & \vdots & l_{n3} \\ & & & \ddots & \vdots \\ & & & & l_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{bmatrix}$$

解得

$$y_j = \left(b_j - \sum_{k=1}^{j-1} l_{jk} y_k \right) \Big/ l_{jj}, \quad j = 1,2,\cdots,n$$

$$x_j = \left(y_j - \sum_{k=j+1}^{n} l_{kj} y_k \right) \Big/ l_{jj}, \quad j = n, n-1, \cdots, 1$$

于是，关于系数矩阵是对称正定矩阵的线性方程组$Ax=b$的求解，分两步进行：

第一步：系数矩阵的平方根分解

$$l_{jj} = \left(a_{jj} - \sum_{k=1}^{j-1} l_{jk}^2\right)^{\frac{1}{2}}, \quad j = 1,2,\cdots,n$$

$$l_{ij} = \left(a_{ij} - \sum_{k=1}^{j-1} l_{ik} l_{jk}\right) / l_{jj}, \quad i = j, j+1,\cdots,n$$

第二步：解等价方程组

$$y_j = \left(b_j - \sum_{k=1}^{j-1} l_{jk} y_k\right) \Big/ l_{jj}, \quad j = 1,2,\cdots,n$$

$$x_j = \left(y_j - \sum_{k=j+1}^{n} l_{kj} y_k\right) \Big/ l_{jj}, \quad j = n, n-1,\cdots,1$$

例5-6 用平方根法求解对称正定方程组

$$\begin{bmatrix} 4 & -1 & 1 \\ -1 & 4.25 & 2.75 \\ 1 & 2.75 & 3.5 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 4 \\ 6 \\ 7.25 \end{bmatrix}$$

解：首先进行A 的Cholesky 分解

$$l_{jj} = (a_{jj} - \sum_{k=1}^{j-1} l_{jk}^2)^{\frac{1}{2}}, \quad j = 1,2,\cdots,n$$

$$l_{ij} = (a_{ij} - \sum_{k=1}^{j-1} l_{ik} l_{jk}) / l_{jj}, \quad i = j, j+1,\cdots,n$$

$$A = LL^T = \begin{bmatrix} 2 & & \\ -0.5 & 2 & \\ 0.5 & 1.5 & 1 \end{bmatrix}\begin{bmatrix} 2 & -0.5 & 0.5 \\ & 2 & 1.5 \\ & & 1 \end{bmatrix}$$

求解$Ly=b$:

$$\begin{bmatrix} 2 & 0 & 0 \\ -0.5 & 2 & 0 \\ 0.5 & 1.5 & 1 \end{bmatrix}\begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}=\begin{bmatrix} 4 \\ 6 \\ 7.25 \end{bmatrix}$$

得 $y_1=2$，$y_2=3.5$，$y_3=1$

再求解 $L^Tx=y$:

$$\begin{bmatrix} 2 & -0.5 & 0.5 \\ 0 & 2 & 1.5 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}=\begin{bmatrix} 2 \\ 3.5 \\ 1 \end{bmatrix}$$

得 $x_1=1$，$x_2=1$，$x_3=1$

关于对称正定方程组

$$\begin{bmatrix} 4 & -1 & 1 \\ -1 & 4.25 & 2.75 \\ 1 & 2.75 & 3.5 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}=\begin{bmatrix} 4 \\ 6 \\ 7.25 \end{bmatrix}$$

也可以用一般的三角分解法求解，这时由

$$\left[\begin{array}{ccc|c} 4 & -1 & 1 & 4 \\ -1 & 4.25 & 2.75 & 6 \\ 1 & 2.75 & 3.5 & 7.25 \end{array}\right] \left[\begin{array}{ccc|c} 4 & -1 & 1 & 4 \\ -0.25 & 4 & 3 & 7 \\ 0.25 & 0.75 & 1 & 1 \end{array}\right]$$

求得 $x_1=1$，$x_2=1$，$x_3=1$

四、追赶法

追赶法是专门用于求解三对角方程组的。这类方程组经常出现于用差分方法或有限元方法求解二阶常微分方程边值问题、热传导问题及三次样条函数插值等问题，三对角方程组$Ax=b$的系数矩阵具有如下形式：

$$A=\begin{bmatrix} b_1 & c_1 & & & \\ a_2 & b_2 & c_2 & & \\ & a_3 & b_3 & \ddots & \\ & & \ddots & \ddots & c_{n-1} \\ & & & a_n & b_n \end{bmatrix}$$

设A为一个三对角矩阵，那么它的顺序主子式均不为零的一个充分条件是：

$$|b_1|>|c_1|>0, |b_n|>|a_n|>0$$

$$|b_i|\ge|c_i|+|a_i|, c_i a_i \ne 0, i=2,\cdots,n-1$$

在此条件下，可对A进行三角分解，设

$$A=\begin{bmatrix} \alpha_1 & & & & \\ \gamma_2 & \alpha_2 & & & \\ & \gamma_3 & \alpha_3 & & \\ & & & \ddots & \\ & & & \gamma_n & \alpha_n \end{bmatrix}\begin{bmatrix} 1 & \beta_1 & & & \\ & 1 & \beta_2 & & \\ & & \ddots & \ddots & \\ & & & \ddots & \beta_{n-1} \\ & & & & 1 \end{bmatrix}$$

比较矩阵的对应元素，根据矩阵乘法规则，可得到

$$\begin{bmatrix} b_1 & c_1 & & & \\ a_2 & b_2 & c_2 & & \\ & a_3 & b_3 & \ddots & \\ & & \ddots & \ddots & c_{n-1} \\ & & & a_n & b_n \end{bmatrix} = \begin{bmatrix} \alpha_1 & & & & \\ \gamma_2 & \alpha_2 & & & \\ & \gamma_3 & \alpha_3 & & \\ & & & \ddots & \\ & & & \gamma_n & \alpha_n \end{bmatrix} \begin{bmatrix} 1 & \beta_1 & & & \\ & 1 & \beta_2 & & \\ & & \ddots & \ddots & \\ & & & \ddots & \beta_{n-1} \\ & & & & 1 \end{bmatrix}$$

$$c_i = a_{i\,i+1} = (0,\cdots,0,\gamma_i,\alpha_i,0,\cdots,0)\begin{array}{r} \\ \\ \\ \\ i\text{行} \\ \\ \\ \\ \end{array}\begin{bmatrix} 0 \\ \vdots \\ 0 \\ \beta_i \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}\begin{array}{l} \\ \\ \\ i+1\text{列} \\ \\ \\ \\ \\ \end{array} = \alpha_i\beta_i, \quad i = 1,2,\cdots,n-1$$

$$
\begin{bmatrix}
b_1 & c_1 & & & \\
a_2 & b_2 & c_2 & & \\
 & a_3 & b_3 & \ddots & \\
 & & \ddots & \ddots & c_{n-1} \\
 & & & a_n & b_n
\end{bmatrix}
=
\begin{bmatrix}
\alpha_1 & & & & \\
\gamma_2 & \alpha_2 & & & \\
 & \gamma_3 & \alpha_3 & & \\
 & & & \ddots & \\
 & & & \gamma_n & \alpha_n
\end{bmatrix}
\begin{bmatrix}
1 & \beta_1 & & & \\
 & 1 & \beta_2 & & \\
 & & \ddots & \ddots & \\
 & & & \ddots & \beta_{n-1} \\
 & & & & 1
\end{bmatrix}
$$

$$
b_i = a_{ii} = \left(0,\cdots,0,\gamma_i,\alpha_i,0,\cdots,0\right)
\begin{bmatrix}
0 \\ \vdots \\ 0 \\ \beta_{i-1} \\ 1 \\ 0 \\ \vdots \\ 0
\end{bmatrix}
= \gamma_i \beta_{i-1} + \alpha_i, \quad i = 1,2,\cdots,n
$$

i 列

i行

$$
\begin{bmatrix}
b_1 & c_1 & & & \\
a_2 & b_2 & c_2 & & \\
 & a_3 & b_3 & \ddots & \\
 & & \ddots & \ddots & c_{n-1} \\
 & & & a_n & b_n
\end{bmatrix}
=
\begin{bmatrix}
\alpha_1 & & & & \\
\gamma_2 & \alpha_2 & & & \\
 & \gamma_3 & \alpha_3 & & \\
 & & & \ddots & \\
 & & & \gamma_n & \alpha_n
\end{bmatrix}
\begin{bmatrix}
1 & \beta_1 & & & \\
 & 1 & \beta_2 & & \\
 & & \ddots & \ddots & \\
 & & & \ddots & \beta_{n-1} \\
 & & & & 1
\end{bmatrix}
$$

$$
a_i = a_{i\,i-1} = \underbrace{\left(0,\cdots,0,\gamma_i,\alpha_i,0,\cdots,0\right)}_{i\text{行}}
\underbrace{\begin{bmatrix} 0 \\ \vdots \\ 0 \\ \beta_{i-2} \\ 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}}_{i\text{-}1\text{列}}
\begin{matrix} \\ \\ \\ \\ i\text{ -}1\text{列} \\ \\ \\ \\ \end{matrix}
= \gamma_i, \quad i = 2,3,\cdots,n
$$

于是，由以上结果：

$$c_i = \alpha_i \beta_i, \quad i = 1,2,\cdots,n$$

$$b_i = \gamma_i \beta_{i-1} + \alpha_i, \quad i = 1,2,\cdots,n$$

$$a_i = \gamma_i, \quad i = 2,3,\cdots,n$$

分别得到：

$$\begin{cases} \gamma_i = a_i, \quad i = 2,3,\cdots,n \\ \alpha_1 = b_1 \qquad \alpha_i = b_i - \gamma_i \beta_{i-1}, \quad i = 2,3,\cdots,n \\ \beta_i = \dfrac{c_i}{\alpha_i}, \quad i = 1,2,\cdots,n-1 \end{cases}$$

也就是说，用这一组公式可以对三对角矩阵进行三角分解：

$$\begin{bmatrix} b_1 & c_1 & & & \\ a_2 & b_2 & c_2 & & \\ & a_3 & b_3 & \ddots & \\ & & \ddots & \ddots & c_{n-1} \\ & & & a_n & b_n \end{bmatrix} = \begin{bmatrix} \alpha_1 & & & & \\ \gamma_2 & \alpha_2 & & & \\ & \gamma_3 & \alpha_3 & & \\ & & & \ddots & \\ & & & \gamma_n & \alpha_n \end{bmatrix} \begin{bmatrix} 1 & \beta_1 & & & \\ & 1 & \beta_2 & & \\ & & \ddots & \ddots & \\ & & & \ddots & \beta_{n-1} \\ & & & & 1 \end{bmatrix}$$

$$\begin{cases} \gamma_i = a_i, \quad i = 2,3,\cdots,n \\ \alpha_1 = b_1 \qquad \alpha_i = b_i - \gamma_i \beta_{i-1}, \quad i = 2,3,\cdots,n \\ \beta_i = \dfrac{c_i}{\alpha_i}, \quad i = 1,2,\cdots,n-1 \end{cases}$$

对于三对角方程组$Ax=b$，设A的三角分解为$A=LU$，则原方程组等价于

$$Ly=b, \quad Ux=y$$

由$Ly=b$，即
$$\begin{bmatrix} \alpha_1 & & & & \\ \gamma_2 & \alpha_2 & & & \\ & \gamma_3 & \alpha_3 & & \\ & & & \ddots & \\ & & & \gamma_n & \alpha_n \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ b_n \end{bmatrix}$$

解得
$$y_1 = b_1 / \alpha_1$$
$$y_i = (b_i - \gamma_i y_{i-1}) / \alpha_i, \quad i = 2, \cdots, n$$

由$Ux=y$，即
$$\begin{bmatrix} 1 & \beta_1 & & & \\ & 1 & \beta_2 & & \\ & & \ddots & \ddots & \\ & & & \ddots & \beta_{n-1} \\ & & & & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{bmatrix}$$

解得
$$x_n = y_n$$
$$x_i = y_i - \beta_i x_{i+1}, \quad i = n-1, \cdots, 1$$

于是，对于三对角矩阵方程组 ***Ax=b***，如下的两组公式便构成了构成了解三对角方程组的追赶法：

$$\begin{cases} \gamma_i = a_i, \quad i = 2,3,\cdots,n \\ \alpha_1 = b_1 \qquad \alpha_i = b_i - \gamma_i \beta_{i-1}, \quad i = 2,3,\cdots,n \\ \beta_i = \dfrac{c_i}{\alpha_i}, \quad i = 1,2,\cdots,n-1 \end{cases}$$

$$\begin{cases} y_1 = b_1/\alpha_1, y_i = (b_i - \gamma_i y_{i-1})/\alpha_i, \quad i = 2,\cdots,n \\ x_n = y_n, x_i = y_i - \beta_i x_{i+1}, \quad i = n-1,\cdots,1 \end{cases}$$

例5-7　用追赶法求解三对角方程组

$$\begin{bmatrix} 2 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 2 \end{bmatrix} \begin{bmatrix} x_2 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

解：首先进行系数矩阵的三角分解

$$\begin{bmatrix} b_1 & c_1 & & & \\ a_2 & b_2 & c_2 & & \\ & a_3 & b_3 & \ddots & \\ & & \ddots & \ddots & c_{n-1} \\ & & & a_n & b_n \end{bmatrix} = \begin{bmatrix} \alpha_1 & & & & \\ \gamma_2 & \alpha_2 & & & \\ & \gamma_3 & \alpha_3 & & \\ & & & \ddots & \\ & & & \gamma_n & \alpha_n \end{bmatrix} \begin{bmatrix} 1 & \beta_1 & & & \\ & 1 & \beta_2 & & \\ & & \ddots & \ddots & \\ & & & \ddots & \beta_{n-1} \\ & & & & 1 \end{bmatrix}$$

$$\begin{cases} \gamma_i = a_i, & i = 2,3,\cdots,n \\ \alpha_1 = b_1 \quad \alpha_i = b_i - \gamma_i \beta_{i-1}, & i = 2,3,\cdots,n \\ \beta_i = \dfrac{c_i}{\alpha_i}, & i = 1,2,\cdots,n-1 \end{cases}$$

$$\begin{bmatrix} 2 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 2 \end{bmatrix} = \begin{bmatrix} 2 & 0 & 0 & 0 \\ -1 & \frac{3}{2} & 0 & 0 \\ 0 & -1 & \frac{4}{3} & 0 \\ 0 & 0 & -1 & \frac{5}{4} \end{bmatrix} \begin{bmatrix} 1 & -\frac{1}{2} & 0 & 0 \\ 0 & 1 & -\frac{2}{3} & 0 \\ 0 & 0 & 1 & -\frac{3}{4} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

求解方程组$Ly=b$，即

$$\begin{bmatrix} 2 & 0 & 0 & 0 \\ -1 & \frac{3}{2} & 0 & 0 \\ 0 & -1 & \frac{4}{3} & 0 \\ 0 & 0 & -1 & \frac{5}{4} \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

得 到 $y_1=1/2$，$y_2=1/3$，$y_3=1/4$，$y_4=1$

再求解方程组 $Ux=y$，即

$$\begin{bmatrix} 1 & -\frac{1}{2} & 0 & 0 \\ 0 & 1 & -\frac{2}{3} & 0 \\ 0 & 0 & 1 & -\frac{3}{4} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} \frac{1}{2} \\ \frac{1}{3} \\ \frac{1}{4} \\ 1 \end{bmatrix}$$

得 到 $x_1=1$，$x_2=1$，$x_3=1$，$x_4=1$

上机实验

- 上机作业：
  - ➢ **编写具有按列选主元策略的高斯消元法函数**
  - ➢ **编写具有按列选主元策略的LU分解函数**
- 实验报告要求：
  - ➢ **有函数使用说明**
  - ➢ **画出流程图**
  - ➢ **给出程序代码**

§2.3 向量范数和矩阵范数

1．向量的范数

定义1：设$X \in R^n$，$\|X\|$ 表示定义在R^n上的一个实值函数，称之为X的范数，它具有下列性质：

（1）**非负性：即对一切$X \in R^n$，$X \neq 0, \|X\| > 0$**

（2）**齐次性：即对任何实数$a \in R$，$X \in R^n$，**

$$\|aX\| = |a| \cdot \|X\|$$

（3）**三角不等式：即对任意两个向量X、$Y \in R^n$，恒有**

$$\|X + Y\| \leq \|X\| + \|Y\|$$

三个常用的范数:

设$X=(x_1, x_2,\ldots, x_n)^T$，则有

（1） $\|X\|_1=|x_1|+|x_2|+\cdots|x_n|$

（2） $\|X\|_2=\sqrt{X^T X}=\sqrt{x_1^2+x_2^2+\cdots+x_n^2}$

（3） $\|X\|_\infty=\max\limits_{1\le i\le n}|x_i|$

定义：设向量x,y属于R^n，则称$\|x-y\|$为x和y之间的距离。

这里$\|\cdot\|$可以是R^n上任何一种向量范数。

2．矩阵的范数

定义3：设A为n 阶方阵，R^n中已定义了向量范数$\|\cdot\|$，

则称 $\max\limits_{\|x\|=1}\|AX\|$ 为矩阵A的范数或模，

记为 $\|A\|$。

$$\|A\| = \max_{\|x\|=1}\|AX\|$$

$$\|A\|_\infty = \max_{\|x\|_\infty=1}\|Ax\|_\infty = \max_{1\ll i\ll n}\sum_{j=1}^{n}|a_{ij}|$$

$$\|A\|_1 = \max_{\|x\|_1=1}\|Ax\|_1 = \max_{1\ll i\ll n}\sum_{i=1}^{n}|a_{ij}|$$

$$\|A\|_2 = \max_{\|x\|_2=1}\|Ax\|_2 = \sqrt{\rho(A^TA)}$$

$\rho(B)$**为矩阵B的谱半径**

矩阵范数的基本性质：

（1）当$A=0$时，$\|A\|=0$，当$A\neq 0$时，$\|A\|>0$

（2）对任意实数和任意A，有 $\|kA\|=|k|\|A\|$

（3）对任意两个n阶矩阵A、B有

$$\|A+B\|\leq\|A\|+\|B\|$$

（4）对任意向量$X\in R^n$，和任意矩阵A，有

$$\|AX\|\leq\|A\|\|X\|$$

（5）对任意两个n阶矩阵A、B，有

$$\|AB\|\leq\|A\|\cdot\|B\|$$

§ 2.4 线性方程组的迭代法求解

直接法得到的解是理论上准确的，但是我们可以看得出，它们的计算量都是n^3数量级，存储量为n^2量级，这在n比较小的时候还比较合适。

按照2000年的国际标准规定：

n<500为小矩阵

在用直接法时就会耗费大量的时间和存储单元。因此我们有必要引入一类新的方法：迭代法。

对方程组 $Ax=b$ 做等价变换 $x=Gx+g$

如：令 $A=M-N$，则

$$Ax=b \Rightarrow (M-N)x=b \Rightarrow Mx=b+Nx \Rightarrow x=M^{-1}Nx+M^{-1}b$$

则，我们可以构造序列 $x^{(k+1)}=G\,x^{(k)}+g$

若 $x^{(k)} \to x^* \Rightarrow x^*=G\,x^*+g \Rightarrow Ax^*=b$

同时：
$$x^{(k+1)}-x^*=Gx^{(k)}-Gx^*=G(x^{(k)}-x^*)$$
$$=\cdots=G^{k+1}(x^{(0)}-x^*)$$

所以，序列收敛 $\Leftrightarrow G^k \to 0$　　与初值的选取无关

迭代过程

$$x_{k+1} = Bx_k + g, \qquad (2)$$

B称为迭代矩阵。

给定初值 x_0,就得到向量序列

$$x_0, x_1 \cdots, x_n \cdots,$$

定义：若 $\lim\limits_{n\to\infty} x_n = x^*$， 称逐次逼近法收敛，否则，称逐次逼近法不收敛或发散。

迭代法的收敛性

定理3.2 (充分条件)对方程组 $x=Bx+f$, 若$\|B\|<1$,则

（1）方程组有唯一解x^*;

（2）对任意初始向量$x^{(0)} \in \mathbf{R}^n$, 迭代格式收敛于x^*，且有

$$\|x^{(k+1)} - x^*\| \le \|B\| \, \|(x^{(k)} - x^*)\|$$

（3） $$\left\|x^{(k)} - x^*\right\| \le \frac{\|B\|}{1-\|B\|}\left\|x^{(k)} - x^{(k-1)}\right\|, (k = 1, 2, \cdots)$$

（4） $$\left\|x^{(k)} - x^*\right\| \le \frac{\|B\|^k}{1-\|B\|}\left\|x^{(1)} - x^{(0)}\right\|, (k = 1, 2, \cdots)$$

证　（1）要证方程组x=Bx+f有唯一解，

即证明齐次方程组 x=Bx 只有零解。

（用反证法）设齐次方程有非零解x_1。

$$x_1=Bx_1$$

取范数

$$\|x_1\| = \|Bx_1\| \le \|B\|\|x_1\|$$

$$\Rightarrow \quad 1 \le \|B\|$$

与已知条件矛盾。故假设不成立，原方程组存在唯一解。

迭代法

（2） $x^{(k+1)} = \mathrm{B}\, x^{(k)} + f$ 与 $x^* = \mathrm{B}\, x^* + f$相减，得

$$x^{(k+1)} - x^* = B(x^{(k)} - x^*)$$

即 $\|x^{(k+1)} - x^*\| \le \|\mathrm{B}\|\,\|(x^{(k)} - x^*)\|$

$$\|x^{(k)} - x^*\| \le \|\mathrm{B}\|^k\,\|(x(0) - x^*)\|$$

$$\|\mathrm{B}\| < 1 \Longrightarrow \lim_{k\to\infty} \|B\|^k \|x^{(0)} - x^*\| = 0$$

$$\Longrightarrow \lim_{k\to\infty} \|x^{(k)} - x^*\| = 0$$

所以 $\lim_{k\to\infty} x^{(k)} = x^*$

迭代法

（3）　由 $x^{(k)}-x^* = x^{(k+1)}-x^* + x^{(k)} - x^{(k+1)}$

$$\Rightarrow \| x^{(k)}-x^*\| \le \| x^{(k+1)}-x^*\| + \|x^{(k)} - x^{(k+1)}\|$$

$$\|x^{(k+1)}-x^*\| \le \|B\|\,\|(x^{(k)}-x^*)\|$$

$$\|x^{(k+1)} - x^{(k)}\| \le \|B\|\,\|(x^{(k)} - x^{(k-1)})\|$$

$$\Rightarrow \|x^{(k)}-x^*\| \le \|B\|\,\|x^{(k)}-x^*\| + \|B\|\,\|x^{(k)} - x^{(k-1)}\|$$

即 $$\left\|x^{(k)}-x^*\right\| \le \frac{\|B\|}{1-\|B\|}\left\|x^{(k)}-x^{(k-1)}\right\|,(k=1,2,\cdots)$$

利用两次迭代结果之差，来估计迭代的误差。

- 迭代法的收敛与迭代矩阵的特征值有关。

定义: 设A为n阶方阵，λ_i $(i = 1,2,\ldots,n)$为A的特征值，称特征值模的最大值为矩阵A的谱半径，记为:

$$\rho(A) = \max_{1 \le i \le n}\{|\lambda_i|\}$$

$(\lambda_1, \lambda_2, \cdots, \lambda_n)$称为矩阵$A$的谱。

由特征值的定义知，矩阵$A^k = AA\cdots A$的谱是 $(\lambda_1^k, \lambda_2^k, \cdots, \lambda_n^k)(k = 1, 2, \cdots)$,因而 $\rho(A^k) = [\rho(A)]^k$

- 矩阵的谱半径与范数之间有如下关系： 设x为对应于特征值λ的A的特征向量，则由：

$$\lambda x = Ax \underset{\text{相容性}}{\Rightarrow} \|\lambda x\| = \|Ax\| \le \|A\|\|x\|$$

$$\Rightarrow |\lambda|\|x\| \le \|A\|\|x\| \underset{\|x\|\neq 0}{\Rightarrow} |\lambda| \le \|A\|$$

A的谱半径不超过A的任一种范数。即：

$$\rho(A) \le \|A\|$$

并且对于任意正数ε，存在一种矩阵范数很接近$\rho(A)$，使得成立：

$$\|A\| \le \rho(A) + \varepsilon$$

对任意n 阶方阵A，一般不存在矩阵范数使$\rho(A)=\|A\|$，

但若A为对称矩阵，则有：

$$\rho(A)=\|A\|$$

迭代法收敛与否只决定于迭代矩阵的谱半径，与初始值和方程右端的常数项无关！

迭代法的收敛条件

定理3.6：对任意初始向量$x^{(0)}$和右 g，由迭代格式：

$$x^{(k+1)}=Ax^{(k)}+g \qquad (k=0,1,2,\ldots)$$

产生的向量序列x^k收敛 $\xleftrightarrow{\text{充分必要条件}}$ $\rho(A)<1$

$\rho(A)$ 难计算，而$\|A\|_\infty$、$\|A\|_1$计算容易，

证明必要性

- 设存在n维向量x^*，使得 $\lim_{k\to\infty} x^{(k)} = x^*$

则x^*满足 $x^* = Mx^* + g$，根据迭代公式有

$$x^{(k+1)} - x^* = Mx^{(k)} - Mx^* = M(x^{(k)} - x^*)$$

$$= \cdots = M^{k+1}(x^{(0)} - x^*)$$

，于是有 $\lim_{k\to\infty} M^{k+1}(x^{(0)} - x^*) = \lim_{k\to\infty}(x^k - x^*) = 0$

因为x(0)是任意不为0的n维向量 $\Rightarrow$ $\lim_{k\to\infty} M^k = 0$

$\Rightarrow \lim_{k\to\infty} \|M^k\| = 0 \xrightarrow{\rho(A) \le \|A\|}$

$$0 \le [\rho(M^k)] = [\rho(M)]^k \le \|M^k\|$$

$\Rightarrow \lim_{k\to\infty}[\rho(M)]^k = 0 \Rightarrow \boxed{\rho(M) < 1}$

证明充分性

- 若 $\rho(M)<1$ 则λ=1不是M的特征值，因而有 $|I-M|\neq 0$ 于是对任意n维向量g，方程组 $(I-M)x=g$ 有唯一解，记为x*，即

$x^*=Mx^*+g$ ，并且 $\lim_{k\to\infty} M^k=0$

$\|M\|\leq \rho(M)+\varepsilon$ $\Rightarrow$ $\|M\|<1$

不妨设 $\varepsilon=(1-\rho(M))/2$

同时 $\|M^k\|\leq \|M\|^k$ $\Rightarrow$ $\lim_{k\to\infty}\|M^k\|\to 0$

$\Rightarrow$ $\lim_{k\to\infty} M^k\to 0$

一、简单迭代法(基本迭代法)

设线性方程组(1)的一般形式为

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

设 $a_{ii} \neq 0 \quad (i = 1,2,\cdots,n)$,则可从上式解出 x_i

$$x_1 = \frac{1}{a_{11}}[b_1 - (a_{12}x_2 + \cdots + a_{1n}x_n)]$$

$$x_2 = \frac{1}{a_{22}}[b_2 - (a_{21}x_1 + a_{23}x_3 + \cdots + a_{2n}x_n)]$$

依此类推,线性方程组(1)可化为

$$\begin{cases} x_1 = \dfrac{1}{a_{11}}\Big(b_1 - \sum\limits_{\substack{j=1\\ j\neq 1}}^{n} a_{1j}x_j\Big) = g_1 - \sum\limits_{\substack{j=1\\ j\neq 1}}^{n} b_{1j}x_j & b_{11} = 0 \\ x_2 = \dfrac{1}{a_{22}}\Big(b_2 - \sum\limits_{\substack{j=1\\ j\neq 2}}^{n} a_{2j}x_j\Big) = g_2 - \sum\limits_{\substack{j=1\\ j\neq 2}}^{n} b_{2j}x_j & b_{22} = 0 \\ \cdots\cdots\cdots\cdots \\ x_i = \dfrac{1}{a_{ii}}\Big(b_i - \sum\limits_{\substack{j=1\\ j\neq i}}^{n} a_{ij}x_j\Big) = g_i - \sum\limits_{\substack{j=1\\ j\neq i}}^{n} b_{ij}x_j & b_{ii} = 0 \\ \cdots\cdots\cdots\cdots \end{cases}$$

$$B = \begin{bmatrix} 0 & b_{12} & b_{13} & \cdots & b_{1n-1} & b_{1n} \\ b_{21} & 0 & b_{23} & \cdots & b_{2n-1} & b_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ b_{n1} & b_{n2} & b_{n3} & \cdots & b_{nn-1} & 0 \end{bmatrix} \qquad g = \begin{bmatrix} g_1 \\ g_2 \\ \vdots \\ g_n \end{bmatrix}$$

线性方程组(1)可化为

$$B=\begin{bmatrix} 0 & b_{12} & b_{13} & \cdots & b_{1n-1} & b_{1n} \\ b_{21} & 0 & b_{23} & \cdots & b_{2n-1} & b_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ b_{n1} & b_{n2} & b_{n3} & \cdots & b_{nn-1} & 0 \end{bmatrix} \qquad g=\begin{bmatrix} g_1 \\ g_2 \\ \vdots \\ g_n \end{bmatrix}$$

雅可比迭代公式

$$x = Bx + g$$

利用迭代式 $x^{(k+1)} = Bx^{(k)} + g$

如果利用系数矩阵A来表示

设 $D = diag(a_{11}, a_{22}, \cdots, a_{nn})$

$$B = I - D^{-1}A$$

例1. 用Jacobi迭代法求解方程组,误差不超过1e-4

$$\begin{pmatrix} 8 & -3 & 2 \\ 4 & 11 & -1 \\ 2 & 1 & 4 \end{pmatrix}\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 20 \\ 33 \\ 12 \end{pmatrix}$$

解:

$$A = \begin{pmatrix} 8 & -3 & 2 \\ 4 & 11 & -1 \\ 2 & 1 & 4 \end{pmatrix}$$

$$D = \begin{pmatrix} 8 & 0 & 0 \\ 0 & 11 & 0 \\ 0 & 0 & 4 \end{pmatrix} \qquad B_J = I - D^{-1}A = \begin{pmatrix} 0 & \frac{3}{8} & -\frac{1}{4} \\ -\frac{4}{11} & 0 & \frac{1}{11} \\ -\frac{1}{2} & -\frac{1}{4} & 0 \end{pmatrix}$$

$$f = D^{-1}b = \begin{pmatrix} 2.5 \\ 3 \\ 3 \end{pmatrix}$$

取初值 $x^{(0)} = [0 \quad 0 \quad 0]^T$，使用 $Jacobi$ 迭代法

$$x^{(k+1)} = B_J x^{(k)} + f \qquad (k = 0,1,2,\cdots n,\cdots)$$

$$x^{(1)} = B_J x^{(0)} + f$$

$$= \begin{pmatrix} 0 & \frac{3}{8} & -\frac{1}{4} \\ -\frac{4}{11} & 0 & \frac{1}{11} \\ -\frac{1}{2} & -\frac{1}{4} & 0 \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 2.5 \\ 3 \\ 3 \end{pmatrix} = \begin{pmatrix} 2.5 \\ 3 \\ 3 \end{pmatrix}$$

$$\left\| x^{(1)} - x^{(0)} \right\| = 4.924$$

$$x^{(2)} = B_J x^{(1)} + f$$

$$= \begin{pmatrix} 0 & \frac{3}{8} & -\frac{1}{4} \\ -\frac{4}{11} & 0 & \frac{1}{11} \\ -\frac{1}{2} & -\frac{1}{4} & 0 \end{pmatrix} \cdot \begin{pmatrix} 2.5 \\ 3 \\ 3 \end{pmatrix} + \begin{pmatrix} 2.5 \\ 3 \\ 3 \end{pmatrix}$$

$$= [2.875, 2.3636, 1]^T$$

$$\left\| x^{(2)} - x^{(1)} \right\| = 2.1320$$

$$x^{(3)} = B_J x^{(2)} + f = \begin{pmatrix} 0 & \frac{3}{8} & -\frac{1}{4} \\ -\frac{4}{11} & 0 & \frac{1}{11} \\ -\frac{1}{2} & -\frac{1}{4} & 0 \end{pmatrix} \cdot \begin{pmatrix} 2.875 \\ 2.3636 \\ 1 \end{pmatrix} + \begin{pmatrix} 2.5 \\ 3 \\ 3 \end{pmatrix}$$

$$= [3.1364, 2.0455, 0.9716]^T \qquad \left\| x^{(3)} - x^{(2)} \right\| = 0.4127$$

依此类推

x4 =	3.0241	1.9478	0.9205	d =	0.1573
x5 =	3.0003	1.9840	1.0010	d =	0.0914
x6 =	2.9938	2.0000	1.0038	d =	0.0175
x7 =	2.9990	2.0026	1.0031	d =	0.0059
x8 =	3.0002	2.0006	0.9998	d =	0.0040
x9 =	3.0003	1.9999	0.9997	d =	7.3612e-004
x10 =	3.0000	1.9999	0.9999	d =	2.8918e-004
x11 =	3.0000	2.0000	1.0000	d =	1.7669e-004
x12 =	3.0000	2.0000	1.0000	d =	3.0647e-005

迭代次数为12次

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 3.0000 \\ 2.0000 \\ 1.0000 \end{pmatrix}$$

```
function x = Jacobi(A,b, x0, ptol)
% Synopsis:  x = Jacobi(A,b,x0)
%            x = Jacobi(A,b,x0,ptol)
% Input:     A,b  = coefficient matrix and right hand side vector
%            ptol = (optional) tolerance for Jacobi
%                   Default:  ptol =1e-8
% Output:    x = solution vector, if solution exists
if nargin<4, ptol = 1e-8;  end
[m,n] = size(A);
if m~=n,  error('A matrix needs to be square');  end
for i = 1:n               %  loop over rows
  pivot = A (i,i);
  if abs(pivot)<50*eps, error('zero pivot encountered');  end
  A(i,:)=-A(i,:)./pivot; A(i,i)=0;g(i)=b(i)/pivot;
end
```

```
% ---
x1=A*x0+g'; i=1;
while ((norm(x1-x0,2)>= ptol) && (i<100))
    x0=x1; x1=A*x0+g'; i=i+1;
end
x=x1;
if i<100
  [ 'toleranc
end
```

```
ans =
tolerance encountered at 21th step
X =
    3.0000
    2.0000
    1.0000
```

X=Jacobi([8 -3 2;4 11 -1;2 1 4],[20;33;12],[0;0;0])

分析Jacobi迭代法的迭代过程

$$x_1^{(k+1)} = x_1^{(k)} + \frac{1}{a_{11}}(b_1 - \sum_{j=1}^{n} a_{1j}x_j^{(k)})$$

$$x_2^{(k+1)} = x_2^{(k)} + \frac{1}{a_{22}}(b_2 - \sum_{j=1}^{n} a_{2j}x_j^{(k)})$$

发现在$x_i^{(k+1)}$之前，$x_1^{(k+1)}, x_2^{(k+1)}, \cdots, x_{i-1}^{(k+1)}$已经求出

但当求$x_i^{(k+1)}$时，仍用$x_1^{(k)}, x_2^{(k)}, \cdots, x_{i-1}^{(k)}$进行迭代

能否求$x_i^{(k+1)}$时，利用$x_1^{(k+1)}, x_2^{(k+1)}, \cdots, x_{i-1}^{(k+1)}$进行迭代呢？

$$B_J = I - D^{-1}A$$

令

$$L = \begin{pmatrix} 0 & 0 & \cdots & 0 \\ -a_{21} & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ -a_{n1} & -a_{n2} & \cdots & 0 \end{pmatrix}$$

$$U = \begin{pmatrix} 0 & -a_{12} & \cdots & -a_{1n} \\ 0 & 0 & \cdots & -a_{2n} \\ \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{pmatrix}$$

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

考虑迭代式

$$x^{(k+1)} = B_J x^{(k)} + g \qquad (k = 0,1,2,\cdots)$$

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

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

注 意 到 L 的 形 式(下 三 角,不 含 对 角 线)

将上式改为

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

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

$D-L$即为A的下三角(包括对角线)部分

当$D-L$可逆时

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

设$B_G = (D-L)^{-1}U,\ f_G = (D-L)^{-1}b$,得

$$x^{(k+1)} = B_G x^{(k)} + f_G$$

$$(k = 0,1,2,\cdots)$$

上式称为Gauss-Seidel迭代法,简称G-S法

利用(8)式展开Gauss-Seidel迭代法也可表示成

$$x_1^{(k+1)} = \qquad -\frac{1}{a_{11}}\sum_{j=2}^{n} a_{1j}x_j^{(k)} + \frac{1}{a_{11}}b_1$$

$$x_2^{(k+1)} = -\frac{1}{a_{22}}\sum_{j=1}^{1} a_{2j}x_j^{(k+1)} - \frac{1}{a_{22}}\sum_{j=3}^{n} a_{2j}x_j^{(k)} + \frac{1}{a_{22}}b_2$$

$$x_3^{(k+1)} = -\frac{1}{a_{33}}\sum_{j=1}^{2} a_{3j}x_j^{(k+1)} - \frac{1}{a_{33}}\sum_{j=4}^{n} a_{3j}x_j^{(k)} + \frac{1}{a_{33}}b_3$$

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

$$x_n^{(k+1)} = -\frac{1}{a_{nn}}\sum_{j=1}^{n-1} a_{nj}x_j^{(k+1)} \qquad + \frac{1}{a_{nn}}b_n$$

例2. 用Gauss-Seidel迭代法求解例1.

解: 取初值 $x^{(0)}=[0,0,0]^T$, 使用 $Gauss-Seidel$ 迭代法

$$x_1^{(k+1)} = -\frac{1}{a_{11}}\sum_{j=2}^{3} a_{1j}x_j^{(k)} + \frac{1}{a_{11}}b_1 = -\frac{1}{8}(-3x_2^{(k)} + 2x_3^{(k)}) + 2.5$$

$$x_2^{(k+1)} = -\frac{1}{a_{22}}\sum_{j=1}^{1} a_{2j}x_j^{(k+1)} - \frac{1}{a_{22}}\sum_{j=3}^{3} a_{2j}x_j^{(k)} + \frac{1}{a_{22}}b_2$$

$$= -\frac{4}{11}x_1^{(k+1)} + \frac{1}{11}x_3^{(k)} + 3$$

$$x_3^{(k+1)} = -\frac{1}{a_{33}}\sum_{j=1}^{2} a_{3j}x_j^{(k+1)} + \frac{1}{a_3}b_3 = -\frac{1}{4}(2x_1^{(k+1)} + 1x_2^{(k+1)}) + 3$$

通过迭代,至第7步得到满足精度的解x7

x1 =2.5000	2.0909	1.2273	d =3.4825
x2 =2.9773	2.0289	1.0041	d =0.5305
x3 =3.0098	1.9968	0.9959	d =0.0465
x4 =2.9998	1.9997	1.0002	d =0.0112
x5 =2.9998	2.0001	1.0001	d =3.9735e-004
x6 =3.0000	2.0000	1.0000	d =1.9555e-004
x7 =3.0000	2.0000	1.0000	d =1.1576e-005

从例1和例2可以看出,Gauss-Seidel迭代法的收敛速度比Jacobi迭代法要高

Jacobi迭代法和Gauss-Seidel迭代法统称为简单迭代法

Gauss-Seidel迭代法的计算过程如下

1.输入$A=(a_{ij}), b=(b_1,\cdots,b_n)$,维数$n, x^{(0)}=(x_1^{(0)}, x_2^{(0)},\cdots,x_n^{(0)})$,
ε,最大容许迭代次数N.

2.置$k=1$.

3.计算 $$x_1=(b_1-\sum_{j=2}^{n}a_{1j}x_j^{(0)})/a_{11}$$

$$x_i=(b_i-\sum_{j=1}^{i-1}a_{ij}x_j-\sum_{j=i+1}^{n}a_{ij}x_j^{(0)})/a_{ii}\quad(i=2,\cdots,n-1)$$

$$x_n=(b_n-\sum_{j=1}^{n-1}a_{nj}x_j)/a_{nn}$$

4.若$\left\|x-x^{(0)}\right\|<\varepsilon$,输出$x$,停机；否则转5。

5.若$k<N$,置$k+1\Rightarrow k, x_i\Rightarrow x_i^{(0)}(i=1,2,\cdots,n)$,转3；
否则，输出失败信息，停机。

```
function x = Gau_Seidel(A,b, x0, ptol)
% Synopsis:  x = Gau_Seidel(A,b,x0)
%            x = Gau_Seidel (A,b,x0,ptol)
% Input:     A,b  = coefficient matrix and right hand side vector
%            ptol = (optional) tolerance for Jacobi
%                   Default:  ptol =1e-8
% Output:    x = solution vector, if solution exists
if nargin<4, ptol = 1e-8;  end
[m,n] = size(A);
if m~=n,  error('A matrix needs to be square');  end
D=diag(diag(A)); L=-tril(A,-1);U=-triu(A,1);
Bg=(D-L)\U;fg=(D-L)\b';
```

```
% ---
x1=Bg*x0+fg; i=1;
while ((norm(x1-x0,2)>= ptol) && (i<100))
    x0=x1; x1=Bg*x0+fg; i=i+1;
end
x=x1;
if i<100
  [ 'toleranc
end
```

```
ans =
tolerance encountered at 11th step
X =
    3.0000
    2.0000
    1.0000
```

X=Gau_Seidel([8 -3 2;4 11 -1;2 1 4],[20;33;12],[0;0;0])

松弛法

记$\Delta x=(\Delta x_1,\Delta x_2,\cdots,\Delta x_n)^T=x^{(k+1)}-x^{(k)}$，其中$x^{(k+1)}$由$Gauss-Seidel$ 迭代公式得到。于是有

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

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

可以把Δx看作$Gauss-Seidel$迭代的修正项，即第k次近似解$x^{(k)}$以此项修正后得到新的近似解

$$x^{(k+1)}=x^{(k)}+\Delta x$$

问题：能否通过修正Δx的值实现迭代的加速收敛呢?

松弛法（续1）

松弛法是将Δx乘上一个参数因子ω作为修正项而得到新的近似解，具体公式为

$$x_i^{(k+1)} = x_i^k + \omega \Delta x_i$$

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

$$(i = 1, 2, \cdots, n)$$

按上式计算$Ax = b$的近似解序列的方法称为松弛法，ω称为松弛因子。当$\omega < 1$时称为低松弛；$\omega = 1$是$Gauss-Seidel$迭代；$\omega > 1$时称为超松弛法（SOR法）。

松弛法（续2）

松弛法迭代公式的矩阵表示：

$$
\begin{aligned}
x^{(k+1)} &= x^{(k)} + \omega\Delta x \\
&= x^{(k)} + \omega(D^{-1}Lx^{(k+1)} + D^{-1}Ux^{(k)} + D^{-1}b - x^{(k)}) \\
&= (1-\omega)x^{(k)} + \omega D^{-1}Lx^{(k+1)} + \omega D^{-1}Ux^{(k)} + \omega D^{-1}b
\end{aligned}
$$

$\left|I - \omega D^{-1}L\right| = 1$ 故（$I - \omega D^{-1}L)^{-1}$与$(D - \omega L)^{-1}$存在，

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

令 $B_S = (D - \omega L)^{-1}[(1-\omega)D + \omega U]$ $f_S = (D - \omega L)^{-1}\omega b$

$$\boxed{x^{(k+1)} = B_s x^{(k)} + f_s}$$

松弛法计算过程如下

1.输入$A=(a_{ij}), b=(b_1,\cdots,b_n)$,维数$n, x^{(0)}=(x_1^{(0)},x_2^{(0)},\cdots,x_n^{(0)})$,
ε,最大容许迭代次数N，参数ω.

2.置$k=1$.

3.计算 $$x_1=(1-\omega)x_1^{(0)}+\omega(b_1-\sum_{j=2}^{n}a_{1j}x_j^{(0)})/a_{11}$$

$$x_i=(1-\omega)x_i^{(0)}+\omega(b_i-\sum_{j=1}^{i-1}a_{ij}x_j-\sum_{j=i+1}^{n}a_{ij}x_j^{(0)})/a_{ii}$$

$$(i=2,\cdots,n-1)$$

$$x_n=(1-\omega)x_n^{(0)}+\omega(b_n-\sum_{j=1}^{n-1}a_{nj}x_j)/a_{nn}$$

4.若$\left\|x-x^{(0)}\right\|<\varepsilon$,输出$x$,停机；否则转5。

5.若$k<N$,置$k+1\Rightarrow k, x_i\Rightarrow x_i^{(0)}(i=1,2,\cdots,n)$,转3；
否则，输出失败信息，停机。

```
function x = SOR(A,b, x0, omega,ptol)
% Synopsis:  x = SOR(A,b,x0)
%            x = SOR (A,b,x0,ptol)
% Input:     A,b  = coefficient matrix and right hand side vector
%     ptol = (optional) tolerance for Seidel, omega=SOR para
%                  Default:  ptol =1e-8
% Output:    x = solution vector, if solution exists
if nargin<5, ptol = 1e-8;  end
[m,n] = size(A);
if m~=n,  error('A matrix needs to be square');  end
D=diag(diag(A)); L=-tril(A,-1);U=-triu(A,1);
Bs=(D-omega*L)\((1-omega)*D+omega*U);
fs=(D-omega*L)\(omega*b);
```

```
% ---
x1=Bs*x0+fs; i=1;
while ((norm(x1-x0,2)>= ptol) && (i<100))
    x0=x1; x1=Bs*x0+fs; i=i+1;
end
x=x1;
if i<100
  [ 'toleranc
end
```

```
ans =
tolerance encountered at 11th step
X =
    3.0000
    2.0000
    1.0000
```

X=SOR([8 -3 2;4 11 -1;2 1 4],[20;33;12],[0;0;0],1)

松弛法例子

例：取 $\omega = 1.4, x^{(0)} = (1,1,1)^T$，用超松弛法解方程组

$$\begin{cases} 2x_1 - x_2 = 1 \end{cases}$$

ans =

tolerance encountered at 27th step

X =

1.2000

1.4000

1.6000

ans =

tolerance encountered at 15th step

X =

1.2000

1.4000

1.6000

$$\Rightarrow \begin{cases} x_2^{(k+1)} \\ x_3^{(k+1)} \end{cases}$$

X=SOR([2 -1 0;-1 2 -1;0 -1 2],[1;0;1.8],[1;1;1],1)

X=SOR([2 -1 0;-1 2 -1;0 -1 2],[1;0;1.8],[1;1;1],1.25)

定理 若SOR方法收敛，则$0<\omega<2$.

证 设SOR方法收敛，则$\rho(\mathbf{M})<1$, 所以

$$|\det(\mathbf{M})| = |\lambda_1\lambda_2\ldots\lambda_n|<1$$

而

$$\det(M) = \det[(\mathbf{D}-\omega\mathbf{L})^{-1}((1-\omega)\mathbf{D}+\omega\mathbf{U})]$$

$$=1/(a_{11}a_{22}\ldots a_{nn})(1-\omega)^n a_{11}a_{22}\ldots a_{nn}$$

$$=(1-\omega)^n$$

于是

$$|1-\omega|<1, \quad \text{或} \quad 0<\omega<2$$

直接用矩阵A判定敛散性

定义： 若 n阶方阵$A=(a_{ij})$满足 $|a_{ii}| \geq \sum_{\substack{j=1 \\ j \neq i}}^{n} |a_{ij}| \quad (i=1,2,\ldots,n)$

且至少有一个 i 值，使上式中不等号严格成立，则称A为<u>弱对角占优阵</u>。若对所有 i，上式中不等号均严格成立，则称A为<u>严格对角占优阵</u>。

设有线性方程组$Ax=b$，下列结论成立：

1. 若A为严格对角占优阵，则*Jacobi*迭代法与*G-S*法均收敛；
2. 若A为严格对角占优阵，$0<\omega\leq 1$，则松驰法收敛；
3. 若A为对称正定矩阵，$0<\omega<2$，则松驰法收敛；

即，若A为对称正定阵，松驰法收敛$\Leftrightarrow 0<\omega<2$。