

# 第三章 线性方程组的直接法

殷东生

yindongsheng@tsinghua.edu.cn

清华大学数学科学系

2023年秋季学期

# 数学要理解、动手

I HEAR AND I FORGET,  
I SEE AND I REMEMBER,  
I DO AND I UNDERSTAND.

“不闻不若闻之，  
闻之不若见之，  
见之不若知之，  
知之不若行之”  
荀子《儒孝篇》

ONE MUST LEARN BY DOING  
THE THING;  
FOR THOUGH YOU THINK YOU  
KNOW IT;  
YOU HAVE NO CERTAINTY UNTIL  
YOU TRY.

$$\frac{1}{n} \sin x = ?$$

$$\frac{1}{n} \cancel{\sin} x =$$

$$six = 6$$

After explaining to a student through various lessons and examples that:

$$\lim_{x \rightarrow 8} \frac{1}{x-8} = \infty$$

I tried to check if she really understood that, so I gave her a different example.

This was the result:

$$\lim_{x \rightarrow 5} \frac{1}{x-5} = \infty$$

考虑下三角方程组

$$L\mathbf{y} = \mathbf{b}, \quad \mathbf{b} = [b_1, b_2, \dots, b_n]^T \in \mathbb{R}^n, \quad \mathbf{y} = [y_1, y_2, \dots, y_n]^T.$$

其中  $\mathbf{b}$  已知, 求  $\mathbf{y}$ , 而  $L = [l_{ij}] \in \mathbb{R}^{n \times n}$  为

$$L = \begin{bmatrix} l_{11} & 0 & \cdots & \cdots & 0 \\ l_{21} & l_{22} & 0 & \cdots & \vdots \\ l_{31} & l_{32} & l_{33} & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots & 0 \\ l_{n1} & l_{n2} & l_{n3} & \cdots & l_{nn} \end{bmatrix}, \quad l_{ii} \neq 0$$

第一个方程为

$$l_{11}y_1 = b_1 \Rightarrow y_1 = \frac{b_1}{l_{11}}.$$

第二个方程：  $l_{21}y_1 + l_{22}y_2 = b_2 \Rightarrow y_2 = \frac{1}{l_{22}}(b_2 - l_{21}y_1)$ .

更一般的如果  $y_1, y_2, \dots, y_{i-1}$  已知，则利用第  $i$  个方程有

$$y_i = \frac{1}{l_{ii}} \left( b_i - \sum_{j=1}^{i-1} l_{ij}y_j \right).$$

这个算法称为向前替换方法。写成算法有

```
function FORSOLVE(L,b)
% Find the solution to the system  $Lx = b$ ,
 $x_1 = b_1/l_{11}$ 
for i = 2:n do
    sum = 0.0
    for j = 1:i-1 do
        sum = sum +  $l_{ij}x_j$ 
    end for
     $x(i) = (b(i) - \text{sum})/l_{ii}$ 
end for
return x
end function
```

总的计算量

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

其计算量为  $\mathcal{O}(n^2)$ .

求解上三角方程组

$$U\mathbf{x} = \mathbf{y}, \quad \mathbf{x} = [x_1, x_2, \dots, x_n] \text{未知}$$

而  $U \in \mathbb{R}^{n \times n}$  如下给出

$$U = \begin{bmatrix} u_{11} & u_{12} & u_{13} & \cdots & u_{1n} \\ 0 & u_{22} & u_{23} & \cdots & u_{2n} \\ 0 & 0 & u_{33} & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & u_{nn} \end{bmatrix}$$

若  $u_{ii} \neq 0$ , 则回带(backward substitution)有

$$x_n = \frac{y_n}{u_{nn}}, \quad x_i = \frac{1}{u_{ii}} \left( y_i - \sum_{j=i+1}^n u_{ij} x_j \right), \quad i = n-1, \dots, 1.$$

用直接法求解

$$Ax = b, \quad A \in \mathbb{R}^{n \times n}, \quad b \in \mathbb{R}^n.$$

本质上是找到一个矩阵  $M, \det M \neq 0$  使得  $MA$  是一个上三角矩阵, 这个过程称为**消元**(elimination), 消元后在求解方程组

$$MAx = Mb \quad \text{回带}$$

其中最简单的Gauss 消去法是用一些列的单位下三角阵  $L_i(l_i)$  来把  $A$  化成上三角矩阵。

$$L_{n-1}(l_{n-1}) \cdots L_i(l_i) \cdots L_1(l_1)A = A^{(n)} = U$$

若令  $L = L_1^{-1} \cdots L_{n-1}^{-1}$ , 则有

$$A = LU$$

假设方程为

$$a_{11}^{(1)}x_1 + a_{12}^{(1)}x_2 + a_{13}^{(1)}x_3 + \cdots + a_{1n}^{(1)}x_n = b_1^{(1)}$$

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

$$a_{31}^{(1)}x_1 + a_{32}^{(1)}x_2 + a_{33}^{(1)}x_3 + \cdots + a_{3n}^{(1)}x_n = b_3^{(1)}$$

$$\vdots$$

$$a_{n1}^{(1)}x_1 + a_{n2}^{(1)}x_2 + a_{n3}^{(1)}x_3 + \cdots + a_{nn}^{(1)}x_n = b_n^{(1)}.$$

若  $a_{11}^{(1)} \neq 0$ , 则可将第 1 行的  $-a_{21}^{(1)}/a_{11}^{(1)}$  倍加到第 2 行上去

$$a_{22}^{(2)}x_2 + a_{23}^{(2)}x_3 + \cdots + a_{2n}^{(2)}x_n = b_2^{(2)}$$

同样可将第 1 行的  $-a_{31}^{(1)}/a_{11}^{(1)}$  倍加到第 3 行上去

$$a_{32}^{(2)}x_2 + a_{33}^{(2)}x_3 + \cdots + a_{3n}^{(2)}x_n = b_3^{(2)}$$



如此一直到第  $n$  行

$$a_{n2}^{(2)}x_2 + a_{n3}^{(2)}x_3 + \cdots + a_{nn}^{(2)}x_n = b_n^{(2)}$$

得到与原方程组等价的一个新的方程组

$$\begin{aligned} a_{11}^{(1)}x_1 + a_{12}^{(1)}x_2 + a_{13}^{(1)}x_3 + \cdots + a_{1n}^{(1)}x_n &= b_1^{(1)} \\ a_{22}^{(2)}x_2 + a_{23}^{(2)}x_3 + \cdots + a_{2n}^{(2)}x_n &= b_2^{(2)} \\ &\vdots \\ a_{n2}^{(2)}x_2 + a_{n3}^{(2)}x_3 + \cdots + a_{nn}^{(2)}x_n &= b_n^{(2)}. \end{aligned}$$

如果  $a_{22}^{(2)} \neq 0$ , 重复消去过程有

$$a_{11}^{(1)}x_1 + a_{12}^{(1)}x_2 + a_{13}^{(1)}x_3 + \cdots + a_{1n}^{(1)}x_n = b_1^{(1)}$$

$$a_{22}^{(2)}x_2 + a_{23}^{(2)}x_3 + \cdots + a_{2n}^{(2)}x_n = b_2^{(2)}$$

$$a_{33}^{(3)}x_3 + \cdots + a_{3n}^{(3)}x_n = b_3^{(3)}$$

$$\vdots$$

$$a_{n3}^{(3)}x_3 + \cdots + a_{nn}^{(3)}x_n = b_n^{(3)}.$$

同样, 如果  $a_{33}^{(3)} \neq 0$ , 上述处理过程可以继续, 直到最后一个方程只含有变量  $x_n$ ,

$$\begin{aligned}a_{11}^{(1)}x_1 + a_{12}^{(1)}x_2 + a_{13}^{(1)}x_3 + \cdots + a_{1n}^{(1)}x_n &= b_1^{(1)} \\a_{22}^{(2)}x_2 + a_{23}^{(2)}x_3 + \cdots + a_{2n}^{(2)}x_n &= b_2^{(2)} \\a_{33}^{(3)}x_3 + \cdots + a_{3n}^{(3)}x_n &= b_3^{(3)} \\&\vdots \\a_{nn}^{(n)}x_n &= b_n^{(n)}.\end{aligned}$$

这样便将原问题转化为一个系数矩阵为上三角阵的线性代数方程组的求解问题。

## 算法 (Gauss 消去)

- 1 *For*  $k = 1 : n - 1$  *Do*:
- 2     *For*  $i = k + 1 : n$  *Do*:
- 3          $piv := a_{ik} / a_{kk}$
- 4         *For*  $j := k + 1 : n + 1$  *Do* :
- 5              $a_{ij} := a_{ij} - piv * a_{kj}$
- 6         *End*
- 7     *End*
- 8 *End*

算法总的计算量为

$$T = \sum_{k=1}^{n-1} \sum_{i=k+1}^n \left[ 1 + \sum_{j=k+1}^{n+1} 2 \right] = \sum_{k=1}^{n-1} \sum_{i=k+1}^n [2(n-k) + 3] = ?$$

# 数学家简介 (Johann Carl Friedrich Gauss)

高斯（1777年4月30日－1855年2月23日）德国著名数学家、物理学家、天文学家、大地测量学家。是近代数学奠基者之一，高斯被认为是历史上最重要的数学家之一，并享有“数学王子”之称。高斯和阿基米德、牛顿并列为世界三大数学家。一生成就极为丰硕，以他名字“高斯”命名的成果达110个，属数学家中之最。他对数论、代数、统计、分析、微分几何、大地测量学、地球物理学、力学、静电学、天文学、矩阵理论和光学皆有贡献。



现考虑矩阵形式.记

$$A^{(k)} = \begin{bmatrix} a_{11}^{(1)} & a_{12}^{(1)} & \cdots & a_{1k-1}^{(1)} & a_{1k}^{(1)} & a_{1k+1}^{(1)} & \cdots & a_{1n}^{(1)} \\ 0 & a_{22}^{(2)} & \cdots & a_{2k-1}^{(2)} & a_{2k}^{(2)} & a_{2k+1}^{(2)} & \cdots & a_{2n}^{(2)} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_{k-1k-1}^{(k-1)} & a_{k-1k}^{(k-1)} & a_{k-1k+1}^{(k-1)} & \cdots & a_{k-1n}^{(k-1)} \\ 0 & 0 & \cdots & 0 & a_{kk}^{(k)} & a_{kk+1}^{(k)} & \cdots & a_{kn}^{(k)} \\ 0 & 0 & \cdots & 0 & a_{k+1k}^{(k)} & a_{k+1k+1}^{(k)} & \cdots & a_{k+1n}^{(k)} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & a_{nk}^{(k)} & a_{nk+1}^{(k)} & \cdots & a_{nn}^{(k)} \end{bmatrix}$$

$$b^{(k)} = [b_1^{(1)}, b_2^{(2)}, \cdots, b_{k-1}^{(k-1)}, b_k^{(k)}, b_{k+1}^{(k)}, \cdots, b_n^{(k)}]^T,$$

$$l_k = [0, 0, \cdots, 0, 0, a_{k+1k}^{(k)} / a_{kk}^{(k)}, \cdots, a_{nk}^{(k)} / a_{kk}^{(k)}]^T.$$

则由如上的过程可知

$$\mathbf{A}^{(k+1)} = L_k(-\mathbf{l}_k)\mathbf{A}^{(k)}, \quad \mathbf{b}^{(k+1)} = L_k(-\mathbf{l}_k)\mathbf{b}^{(k)}.$$

其中

$$L_k(-\mathbf{l}_k) = \begin{bmatrix} 1 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & 1 & 0 & \cdots & 0 \\ 0 & \cdots & -a_{k+1k}^{(k)}/a_{kk}^{(k)} & 1 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & -a_{nk}^{(k)}/a_{kk}^{(k)} & 0 & \cdots & 1 \end{bmatrix}$$

则最后有

$$\begin{aligned} \mathbf{A}^{(n)} &= L_{n-1}(-l_{n-1})L_{n-2}(-l_{n-2})\cdots L_1(-l_1)\mathbf{A}^{(1)}, \\ \mathbf{b}^{(n)} &= L_{n-1}(-l_{n-1})L_{n-2}(-l_{n-2})\cdots L_1(-l_1)\mathbf{b}^{(1)}. \end{aligned}$$

由 $L_j^{-1}(l_j) = L_j(-l_j)$ 知

$$\mathbf{A}^{(1)} = L_1(l_1)L_2(l_2)\cdots L_{n-1}(l_{n-1})\mathbf{A}^{(n)}.$$

令 $\mathbf{L} = L_1(l_1)L_2(l_2)\cdots L_{n-1}(l_{n-1})$ ,  $\mathbf{U} = \mathbf{A}^{(n)}$ , 则 $\mathbf{L}$ 为单位下三角阵(即对角线元素为单位1的下三角阵),  $\mathbf{U}$ 为上三角阵。有

$$\mathbf{A} = \mathbf{L}\mathbf{U}$$



## 定理 ( $LU$ 分解)

非奇异矩阵  $A \in \mathbb{R}^{n \times n}$ , 若其顺序主子式

$$\Delta_i \stackrel{\text{def}}{=} \begin{vmatrix} a_{11} & a_{12} & \cdots & a_{1i} \\ a_{21} & a_{22} & \cdots & a_{2i} \\ \vdots & \vdots & \ddots & \vdots \\ a_{i1} & a_{i2} & \cdots & a_{ii} \end{vmatrix}, \quad i = 1, 2, \cdots, n$$

均不为零, 则存在唯一的单位下三角阵  $L$  和上三角阵  $U$ , 使

$$A = LU.$$

上述分解称为矩阵的三角分解, 或  $LU$  分解, 或 *Doolittle* 分解。



$$U = \begin{pmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ 0 & u_{22} & \cdots & u_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & u_{nn} \end{pmatrix}, \det U \neq 0$$

$$D = \mathbf{diag}(u_{11}, u_{22}, \cdots, u_{nn})$$

$$U^* = \begin{pmatrix} 1 & \frac{u_{12}}{u_{11}} & \cdots & \frac{u_{1n}}{u_{11}} \\ 0 & 1 & \cdots & \frac{u_{2n}}{u_{22}} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix}$$

则  $U = DU^*$ 。于是有

$$A = LDU^*.$$

其中  $D$  为对角阵,  $L, U^*$  分别为单位下三角阵和单位上三角阵。

## 定理

非奇异矩阵  $A \in \mathbb{R}^{n \times n}$ , 若其顺序主子式  $\Delta_i, i = 1, 2, \dots, n$  均不为零, 则存在唯一的单位下三角阵  $L$ , 单位上三角阵  $U^*$  和对角阵  $D$ , 使  $A = LDU^*$ 。

由上可知, 若矩阵  $A$  的所有顺序主子式都非零, 则有唯一的三角分解  $A = LU$ , 其中  $L$  为单位下三角阵,  $U$  为上三角阵。

$$\underbrace{\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix}}_A = \underbrace{\begin{pmatrix} 1 & 0 & \cdots & 0 \\ l_{21} & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ l_{n1} & l_{n2} & \cdots & 1 \end{pmatrix}}_L \underbrace{\begin{pmatrix} u_{11} & u_{12} & \cdots & u_{1n} \\ 0 & u_{22} & \cdots & u_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & u_{nn} \end{pmatrix}}_U$$

两矩阵中的未知变量个数为  $n^2$  与矩阵  $A$  的元素个数一致, 因而可以直接由矩阵  $A$  确定  $L$  和  $U$ 。

# Doolittle分解法

$$\begin{aligned}a_{kj} &= \sum_{r=1}^{\min(k,j)} l_{kr}u_{rj}, \\a_{kj} &= \sum_{r=1}^k l_{kr}u_{rj} = \sum_{r=1}^{k-1} l_{kr}u_{rj} + u_{kj}, \quad j \geq k \\u_{kj} &= a_{kj} - \sum_{r=1}^{k-1} l_{kr}u_{rj}. \\a_{jk} &= \sum_{r=1}^k l_{jr}u_{rk} = \sum_{r=1}^{k-1} l_{jr}u_{rk} + l_{jk}u_{kk}, \quad j > k \\l_{jk} &= \frac{1}{u_{kk}} \left( a_{jk} - \sum_{r=1}^{k-1} l_{jr}u_{rk} \right).\end{aligned}$$

## Doolite 分解 Matlab 函数

```
function [A] = lu_ijk (A)
[n,n]=size(A);
for i=1:n
for j=2:i
A(i,j-1)=A(i,j-1)/A(j-1,j-1);
for k=1:j-1,
    A(i,j)=A(i,j)-A(i,k)*A(k,j);
end
end
for j=i+1:n
for k=1:i-1,
    A(i,j)=A(i,j)-A(i,k)*A(k,j);
end
end
end
```

## 例

用 *Doolittle* 分解方法求解如下方程组

$$x_1 - x_2 = 0,$$

$$-2x_1 + 4x_2 - 2x_3 = -1,$$

$$-x_2 + 2x_3 = 1.5.$$

解：由于

$$A = \begin{pmatrix} 1 & -1 & 0 \\ -2 & 4 & -2 \\ 0 & -1 & 2 \end{pmatrix}, \mathbf{b} = \begin{pmatrix} 0 \\ -1 \\ 1.5 \end{pmatrix}$$

首先对系数矩阵做 Doolittle 分解

$$\begin{aligned} & \begin{pmatrix} 1 & -1 & 0 \\ -2 & 4 & -2 \\ 0 & -1 & 2 \end{pmatrix} \rightarrow \begin{pmatrix} \boxed{1} & \boxed{-1} & \boxed{0} \\ -2 & 4 & -2 \\ 0 & -1 & 2 \end{pmatrix} \\ \rightarrow & \begin{pmatrix} \boxed{1} & \boxed{-1} & \boxed{0} \\ \boxed{-2} & 4 & -2 \\ \boxed{0} & -1 & 2 \end{pmatrix} \rightarrow \begin{pmatrix} \boxed{1} & \boxed{-1} & \boxed{0} \\ \boxed{-2} & \boxed{4} & \boxed{-2} \\ \boxed{0} & -1 & 2 \end{pmatrix} \\ \rightarrow & \begin{pmatrix} \boxed{1} & \boxed{-1} & \boxed{0} \\ \boxed{-2} & \boxed{4} & \boxed{-2} \\ \boxed{0} & \boxed{-1} & 2 \end{pmatrix} \rightarrow \begin{pmatrix} \boxed{1} & \boxed{-1} & \boxed{0} \\ \boxed{-2} & \boxed{4} & \boxed{-2} \\ \boxed{0} & \boxed{-1} & \boxed{2} \end{pmatrix} \end{aligned}$$

其次求解  $\mathbf{Ly} = \mathbf{b}$ , 得到  $\mathbf{y} = (0, -1, 1)^T$ ; 最后求解  $\mathbf{Ux} = \mathbf{y}$ , 得到  $\mathbf{x} = (1/2, 1/2, 1)^T$ 。

## Matlab 求矩阵 $A$ 的 LU 分解的命令为

```
>> A = [ 1    2    3  
        4    5    6  
        7    8    0 ];  
>> [L,U]=lu(A)
```

L =

0.1429	1.0000	0
0.5714	0.5000	1.0000
1.0000	0	0

U =

7.0000	8.0000	0
0	0.8571	3.0000
0	0	4.5000



方程组  $A\mathbf{x} = \mathbf{d}$ , 其中  $\mathbf{d} = (d_1, \dots, d_n)^T$ , 而  $A$  为 **三对角矩阵**(tridiagonal matrix)

$$A = \begin{bmatrix} b_1 & c_1 & & & \\ a_2 & b_2 & c_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & \ddots \\ & & & a_n & b_n \end{bmatrix}$$

若 $A$ 的顺序主子式非零, 则可以用 **Doolittle** 分解方法求解。三对角矩阵是**稀疏矩阵** (sparse matrix), 因为共有

$$n^2 - (n + 2(n - 1)) = n^2 - 3n + 2$$

个零元素。

## 一维Poisson方程

$$-\frac{d^2 u(x)}{dx^2} = f(x), \quad 0 < x < 1,$$

其中  $f$  是给定的函数。  $u(x)$  还必须满足边界条件

$$u(0) = u(1) = 0.$$

首先把区间  $[0, 1]$  等分  $N + 1$ , 有  $N + 2$  个离散的点

$$x_i = ih, \quad h = \frac{1}{N + 1}, \quad 0 \leq i \leq N + 1.$$

记  $u_i = u(x_i)$ ,  $f_i = f(x_i)$ , 用中心差分近似

$$-\frac{d^2 u(x)}{dx^2} \Big|_{x=x_i} = \frac{2u_i - u_{i-1} - u_{i+1}}{h^2} - \tau_i$$



其中  $\tau_i$  称为截断误差,

$$\tau_i \approx O(h^2 \cdot \|\frac{d^4 u}{dx^4}\|_\infty).$$

则Poisson方程在  $x = x_i$  处可写为

$$-u_{i-1} + 2u_i - u_{i+1} = h^2 f_i + h^2 \tau_i, \quad 1 \leq i \leq N.$$

由边界条件知  $u_0 = u_{N+1} = 0$ , 因而有下面的方程组

$$T_N \cdot \begin{bmatrix} u_1 \\ \vdots \\ \vdots \\ u_N \end{bmatrix} \equiv \begin{bmatrix} 2 & -1 & & 0 \\ -1 & \ddots & \ddots & \\ & \ddots & \ddots & -1 \\ 0 & & -1 & 2 \end{bmatrix} \begin{bmatrix} u_1 \\ \vdots \\ \vdots \\ u_N \end{bmatrix} = h^2 \begin{bmatrix} f_1 \\ \vdots \\ \vdots \\ f_N \end{bmatrix} + h^2 \begin{bmatrix} \tau_1 \\ \vdots \\ \vdots \\ \tau_N \end{bmatrix}$$

三对角矩阵可以稀疏存储:

$$\mathbf{a} = [a_2, a_3, \dots, a_n]^T,$$

$$\mathbf{b} = [b_1, b_2, \dots, b_{n-1}, b_n]^T,$$

$$\mathbf{c} = [c_1, c_2, \dots, c_{n-1}]^T$$

对于三对角阵  $L$  和  $U$  有下面的特殊形式:

$$L = \begin{bmatrix} 1 & & & & \\ l_2 & 1 & & & \\ & l_3 & 1 & & \\ & & \ddots & \ddots & \\ & & & l_n & 1 \end{bmatrix}, U = \begin{bmatrix} u_1 & c_1 & & & \\ & u_2 & c_2 & & \\ & & \ddots & \ddots & \\ & & & \ddots & c_{n-1} \\ & & & & u_n \end{bmatrix}$$

$$\Rightarrow \begin{cases} u_1 = b_1, \\ l_i = \frac{a_i}{u_{i-1}}, i = 2, 3, \dots, n \\ u_i = b_i - l_i c_{i-1}, i = 2, 3, \dots, n \end{cases}$$

可求得  $L$  和  $U$ , 求解  $Ax = d$  转化为  $Ly = d$  和  $Ux = y$ ,

$$\begin{cases} y_1 = d_1, \\ y_i = d_i - l_i y_{i-1}, i = 2, 3, \dots, n \\ x_n = \frac{y_n}{u_n}, \\ x_i = \frac{y_i - c_i x_{i+1}}{u_i}, i = n-1, n-2, \dots, 1. \end{cases}$$

上面的计算过程称为求解三对角方程组的**追赶法**。

用追赶法求解三对角方程组的计算量为  $\mathcal{O}(n)$ 。

```
function THOMAS(a,b,c,rhs)
% The function solves a tridiagonal system of linear equations Ax = rhs
% using the linear Thomas algorithm. a is the lower diagonal, b the
% diagonal, and c the upper diagonal.

% Begin elimination steps, resulting in a bidiagonal matrix
% with 1s on its diagonal.
c1 = c1/b1
rhs1 = rhs1/b1
for i = 2:n-1 do
    ci = ci / (bi - ai-1ci-1)
    rhsi = (rhsi - ai-1rhsi-1) / (bi - ai-1ci-1)
end for
rhsn = (rhsn - an-1rhsn-1) / (bn - an-1cn-1)
% Now perform back substitution
xn = rhsn
for i = n-1:-1:1 do
    xi = rhsi - ciXi+1
end for
return x
end function
```

```
>> a = randn(4999,1);
>> b = randn(5000,1);
>> c = randn(4999,1);
>> rhs = randn(5000,1);
>> tic;x1 = thomas(a,b,c,rhs);toc;
Elapsed time is 0.032754 seconds.
>> T = trid(a,b,c);
>> tic;x2 = T\rhs;toc;
Elapsed time is 0.386797 seconds.
```

# 循环三对角阵

周期样条插值等问题遇到的循环三对角方程  $Ax = d$ ,

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

若  $A$  的顺序主子式非零, 则可以用Doolittle分解方法求解。



此循环三对角阵  $L$  和  $U$  有下面的特殊形式：

$$L = \begin{bmatrix} 1 & & & & \\ l_2 & 1 & & & \\ & l_3 & 1 & & \\ & & \ddots & \ddots & \\ \sigma_1 & \sigma_2 & \cdots & \sigma_{n-1} + l_n & 1 \end{bmatrix}, \quad U = \begin{bmatrix} u_1 & c_1 & & & \rho_1 \\ & u_2 & c_2 & & \rho_2 \\ & & \ddots & \ddots & \vdots \\ & & & \ddots & c_{n-1} + \rho_{n-1} \\ & & & & u_n \end{bmatrix}$$

## 数学家简介 (L. H. Thomas)

*L. H. Thomas*(1903-1992) 著名的物理学家，上世纪50年代在IBM的Watson实验室工作，其最著名的工作是 *Thomas-Fermi electron gas model*，是密度泛函理论的基础。

由矩阵乘法有

$$\begin{cases} u_1 = b_1, \rho_1 = a_n, \sigma_1 = \frac{c_n}{u_1}, \\ l_i = \frac{a_i}{u_{i-1}}, u_i = b_i - l_i c_{i-1}, \rho_i = -l_i \rho_{i-1}, \\ \sigma_i = -\frac{\sigma_{i-1} c_{i-1}}{u_{i-1}}, \quad i = 1, s, \dots, n-1, \\ l_n = \frac{a_n}{u_{n-1}}, \\ u_n = b_1 - (l_n + \sigma_{n-1})(c_{n-1} + \rho_{n-1}) - \sum_{i=1}^{n-2} \sigma_i \rho_i \end{cases}$$

可求得  $L$  和  $U$  的所有元素, 解原方程组  $Ax = d$  分为两步求解  $Ly = d$  和  $Ux = y$ ,

$$\begin{cases} y_1 = d_1, \\ y_i = d_i - l_i y_{i-1}, i = 2, 3, \dots, n \\ y_n = d_n - \sum_{i=1}^{n-2} \sigma_i y_i - (l_n + \sigma_{n-1}) y_{n-1}, \\ \begin{cases} x_n = \frac{y_n}{u_n}, \\ x_i = \frac{y_i - c_i x_{i+1} - \rho_i x_n}{u_i}, i = n-1, n-2, \dots, 1. \end{cases} \end{cases}$$

**Gauss**消去法如果  $a_{kk}^{(k)} = 0$ ，则消去法不能继续下去。即便系数矩阵  $A$  非奇异，理论上方程组存在唯一解亦是如此。此外，即便在消去过程中， $a_{kk}^{(k)} \neq 0$ ，但是如果  $|a_{kk}^{(k)}| \ll 1$ ，则会产生大数除小数的情况，导致数值解的精度下降，有时更会完全失真。

$$x_2 + x_3 = 2$$

$$x_1 - x_2 + x_3 = 1$$

$$x_1 + x_2 - x_3 = 1$$

其准确解为  $\mathbf{x} = (1, 1, 1)^T$ 。但系数矩阵

$$\begin{pmatrix} 0 & 1 & 1 \\ 1 & -1 & 1 \\ 1 & 1 & -1 \end{pmatrix}$$

**Doolittle** 分解第一步即不能进行。

# 例

用3位十进制浮点运算求解

$$1.00 \times 10^{-5}x_1 + 1.00x_2 = 1.00,$$

$$1.00x_1 + 1.00x_2 = 2.00.$$

解：这个方程组的准确解用3位十进制浮点数近似为  $(1.00, 1.00)^T$ ；

$$\begin{pmatrix} 1.00 \times 10^{-5} & 1.00 \\ 1.00 & 1.00 \end{pmatrix} \rightarrow \begin{pmatrix} \boxed{1.00 \times 10^{-5}} & \boxed{1.00} \\ \boxed{1.00 \times 10^5} & 1.00 \end{pmatrix} \\ \rightarrow \begin{pmatrix} \boxed{1.00 \times 10^{-5}} & \boxed{1.00} \\ \boxed{1.00 \times 10^5} & \boxed{-1.00 \times 10^5} \end{pmatrix}$$

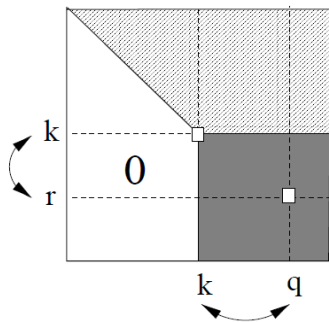
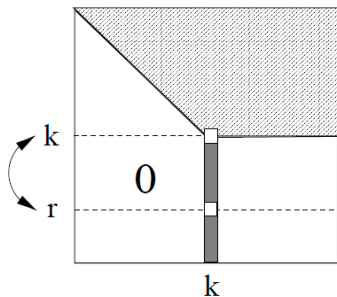
第一步求解  $\mathbf{Ly} = \mathbf{b}$ ，得到  $\mathbf{y} = (1.00, -1.00 \times 10^5)^T$ 。第二步求解  $\mathbf{Ux} = \mathbf{y}$ ，得到  $\mathbf{x} = (0.00, 1.00)^T$ 。

# 选主元(pivoting)

如上所述困难有一自然的解决方案：即在第  $k$  步消去之前，选择

$$a_{kk}^{(k)}, a_{k+1k}^{(k)}, \dots, a_{nk}^{(k)}$$

模值最大者，将其所在的行与第  $k$  行交换，然后进行消去过程。



设其所在的行数为  $i_k \geq k$ , 则 加入如上处理过程后有

$$A^{(k+1)} = L_k(-l_k)I_{i_k k}A^{(k)}.$$

这里  $I_{ij}$  是将单位矩阵的第  $i, j$  行对换得到的矩阵,  $I_{ij}A$  是将  $A$  的第  $i, j$  行对换所得的矩阵。记  $U = A^{(n)}$ , 则

$$U = L_{n-1}(-l_{n-1})I_{i_{n-1}n-1} \cdots L_2(-l_2)I_{i_2 2}L_1(-l_1)I_{i_1 1}A$$

## 定理

设  $A$  非奇异, 则存在排列阵  $P$ , 单位下三角阵  $L$  和上三角阵  $U$ , 使得

$$PA = LU.$$







```

function [L,U,P,Q] = LUpivotot(A,n)
P=eye(n); Q=P; Minv=P;
for k=1:n-1
    [Pk,Qk]=pivot(A,k,n);    A=Pk*A*Qk;
    [Mk,Mkinv]=MGauss(A,k,n);
    A=Mk*A;    P=Pk*P;    Q=Q*Qk;
    Minv=Minv*Pk*Mkinv;
end
U=triu(A);    L=P*Minv;

```

```

function [Mk,Mkinv]=MGauss(A,k,n)
Mk=eye(n);
for i=k+1:n,    Mk(i,k)=-A(i,k)/A(k,k);    end
Mkinv=2*eye(n)-Mk;

```

```

function [Pk,Qk]=pivot(A,k,n)
[y,i]=max(abs(A(k:n,k:n)));    [piv,jpiv]=max(y);
ipiv=i(jpiv);    jpiv=jpiv+k-1;    ipiv=ipiv+k-1;
Pk=eye(n); Pk(ipiv,ipiv)=0; Pk(k,k)=0; Pk(k,ipiv)=1; Pk(ipiv,k)=1;
Qk=eye(n); Qk(jpiv,jpiv)=0; Qk(k,k)=0; Qk(k,jpiv)=1; Qk(jpiv,k)=1;

```

# Matlab 求 LU 分解的命令

```
A = [ 1    2    3  
      4    5    6  
      7    8    0 ];
```

```
[L, U, P]=lu(A)
```

```
L =
```

```
1.0000    0    0  
0.1429    1.0000    0  
0.5714    0.5000    1.0000
```

```
U =
```

```
7.0000    8.0000    0  
0    0.8571    3.0000  
0    0    4.5000
```

```
P =
```

```
0    0    1  
1    0    0  
0    1    0
```

## 定理

非奇异矩阵  $A \in \mathbb{R}^{n \times n}$ , 若其顺序主子式  $\Delta_i, i = 1, 2, \dots, n$  均不为零, 则存在唯一的单位下三角阵  $L$ , 单位上三角阵  $U^*$  和对角阵  $D$ , 使得

$$A = LDU^*.$$

## 推论

设  $A \in \mathbb{R}^{n \times n}$ ,  $A^T = A$ ,  $A$  的顺序主子式  $\Delta_i, i = 1, 2, \dots, n$  都非零, 则存在唯一的单位下三角阵  $L$  和对角阵  $D$ , 使得

$$A = LDL^T.$$

## 定理 (Cholesky分解)

设  $A \in \mathbb{R}^{n \times n}$  为对称正定阵, 则存在唯一的对角元素为正的下三角阵  $L$ , 使得

$$A = LL^T.$$

上述分解称为对称正定阵的 **Cholesky分解**。

利用  $A$  的 Cholesky 分解式来求  $Ax = b$  的方法称为 **Cholesky方法** 或 **平方根法**。

## 数学家简介 (Cholesky)

**Cholesky(1875-1918)** 法国军官和数学家, 在北非和克里特岛进行大地测量。首先用 **Cholesky** 方法求解地图测量中的最小二乘问题的法方程。



# Cholesky 方法

$$a_{ij} = \sum_{k=1}^n l_{ik} l_{jk} = \sum_{k=1}^j l_{ik} l_{jk}, i \geq j,$$

$$a_{jj} = \sum_{k=1}^j l_{jk}^2 = \sum_{k=1}^{j-1} l_{jk}^2 + l_{jj}^2,$$

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

$$a_{ij} = \sum_{k=1}^j l_{ik} l_{jk} = \sum_{k=1}^{j-1} l_{ik} l_{jk} + l_{ij} l_{jj}, i > j$$

$$l_{ij} = \frac{1}{l_{jj}} \left( a_{ij} - \sum_{k=1}^{j-1} l_{ik} l_{jk} \right).$$

```
function [A] = chol2 (A)
[n,n]=size(A);
for k=1:n-1
A(k,k)=sqrt(A(k,k));
A(k+1:n,k)=A(k+1:n,k)/A(k,k);
for j=k+1:n,
    A(j,n,j)=A(j:n,j)-A(j:n,k)*A(j,k);
end
end
A(n,n)=sqrt(A(n,n));
```

$A = LL^T$  分解后, 求解  $LL^T x = b$ , 依次求解  $Ly = b$  和  $L^T x = y$ 。由于

$$a_{ii} = \sum_{k=1}^i l_{ik}^2,$$

可以推出  $|l_{ik}| \leq \sqrt{a_{ii}}, k = 1, 2, \dots, i$ , 所以 Cholesky 方法中, 中间量得到控制, 不会产生"大数吃掉小数"的情况。

Matlab求  $A$  的Cholesky分解的函数为  $\text{chol}(A)$ 。

```
>> A = [2 2 18; 2 24 -18; 18 -18 234];  
>> b = [65; -62; 819];  
>> R = chol(A);  
>> y = R' \ b;  
>> x = R \ y  
x =  
    17.0000  
    -2.5000  
     2.0000
```

## 注记

Cholesky 分解的算法可以用来检验一个对称矩阵  $A$  是否正定。



## 注记

矩阵  $A \in \mathbb{R}^{n \times n}$  的行列式可以用  $LU$  分解  $PA = LU$  来计算

$$\det A = \det(P) \cdot \det(LU) = \operatorname{sgn}(\pi_0) \cdots u_{11} \cdots u_{nn}$$

但是在计算行列式的时候要非常小心； 因为

$$\det(\alpha A) = \alpha^n \det A$$

而这个性质导致在实际计算的时候由于舍入误差的影响会把一个“小”的行列式变得任意“大”。实际上在上面的这个变换下唯一不变的量是 Boole 量

$$\det A = 0, \text{ 或者 } \det A \neq 0.$$

而这个量是衡量一个线性方程组可解性的一个理论特征，但是可解性理论并不是基于行列式的。

# 数值求逆

可利用 $LU$ 分解计算矩阵的逆。设  $X$  是  $A \in \mathbb{R}^{n \times n}$  的逆矩阵, 则  $X$  的列向量是下列线性方程组的解:

$$Ax_i = e_i, \quad i = 1, \dots, n.$$

若

$$PA = LU,$$

而  $P$  是选主元的排列阵, 则为了求逆需要求  $2n$  个三角形形式的方程组

$$Ly_i = Pe_i, \quad i = 1, \dots, n$$

$$Ux_i = y_i, \quad i = 1, \dots, n$$

通过求解方程组求逆矩阵的计算量?

在用直接法求解  $Ax = b$  的过程中, 由于舍入误差的存在, 必然会导致结果产生误差。因而有必要对可能产生的误差作一估计。通常假设在数值处理的过程中计算都是精确的。

例

方程组

$$\begin{pmatrix} 3 & 1 \\ 3.0001 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 4 \\ 4.0001 \end{pmatrix}$$

的精确解是  $(1, 1)^T$ 。若  $A$  和  $b$  作如下的微小变化,

$$\begin{pmatrix} 3 & 1 \\ 2.9999 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 4 \\ 4.0002 \end{pmatrix}.$$

其精确解是  $(-2, 10)^T$ ; 数据的微小变化产生解的巨大变化。

方程组

$$\begin{pmatrix} 10 & 7 & 8 & 7 \\ 7 & 5 & 6 & 5 \\ 8 & 6 & 10 & 9 \\ 7 & 5 & 9 & 10 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 32 \\ 23 \\ 33 \\ 31 \end{pmatrix}$$

其精确解为  $(1, 1, 1, 1)^T$ 。对右端向量做微小的修改

$$\begin{pmatrix} 10 & 7 & 8 & 7 \\ 7 & 5 & 6 & 5 \\ 8 & 6 & 10 & 9 \\ 7 & 5 & 9 & 10 \end{pmatrix} \begin{pmatrix} x_1 + \delta x_1 \\ x_2 + \delta x_2 \\ x_3 + \delta x_3 \\ x_4 + \delta x_4 \end{pmatrix} = \begin{pmatrix} 32.1 \\ 22.9 \\ 33.1 \\ 30.9 \end{pmatrix}$$

其精确解为  $(9.2, -12.6, 4.5, -1.1)^T$ 。

若对系数矩阵做微小的修改

$$\begin{pmatrix} 10 & 7 & 8.1 & 7.2 \\ 7.08 & 5.04 & 6 & 5 \\ 8 & 5.98 & 9.89 & 9 \\ 6.99 & 4.99 & 9 & 9.98 \end{pmatrix} \begin{pmatrix} x_1 + \Delta x_1 \\ x_2 + \Delta x_2 \\ x_3 + \Delta x_3 \\ x_4 + \Delta x_4 \end{pmatrix} = \begin{pmatrix} 32 \\ 23 \\ 33 \\ 31 \end{pmatrix}$$

其精确解为  $\mathbf{x} + \Delta \mathbf{x} = (-81, 137, -34, 22)^T$ 。

### 定义 (条件数)

设  $A \in \mathbb{R}^{n \times n}$  为可逆阵,  $\|\cdot\|$  为一种从属的矩阵范数, 则

$$\text{cond}(A) = \|A\| \cdot \|A^{-1}\|$$

称为  $A$  的**条件数**。



矩阵的条件数的概念是由 Turing 引入的：  
Quart. J. Mech. Appl. Math. 1, 287-308, 1948.

## 数学家简介 (Alan Mathison Turing)

图灵 (1912-1954)，英国数学家、逻辑学家，被称为计算机之父，人工智能之父。图灵在科学、特别在数理逻辑和计算机科学方面，取得了举世瞩目的成就，他的一些科学成果，构成了现代计算机技术的基础：如可计算理论，判定问题、人工智能、图灵实验等。世界级的长跑健将，著名的同性恋者，受到英国政府的迫害，职业生涯尽毁并自杀身亡。



记

$$\text{cond}_2(A) = \|A\|_2 \cdot \|A^{-1}\|_2.$$

类似定义  $\text{cond}_\infty(A)$  和  $\text{cond}_1(A)$ 。

## 定理

设  $A \in \mathbb{R}^{n \times n}$ ,  $A$  非奇异,  $\mathbf{x}$  及  $\mathbf{x} + \delta\mathbf{x}$  满足

$$A\mathbf{x} = \mathbf{b}, A(\mathbf{x} + \delta\mathbf{x}) = \mathbf{b} + \delta\mathbf{b},$$

且  $\mathbf{b} \neq 0$ , 则

$$\frac{\|\delta\mathbf{x}\|}{\|\mathbf{x}\|} \leq \text{cond}(A) \frac{\|\delta\mathbf{b}\|}{\|\mathbf{b}\|}$$



## 定理

设  $A \in \mathbb{R}^{n \times n}$ ,  $\det(A) \neq 0$ ,  $\mathbf{x}$  和  $\mathbf{x} + \delta\mathbf{x}$  分别满足方程组

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

$$(A + \delta A)(\mathbf{x} + \delta\mathbf{x}) = \mathbf{b} + \delta\mathbf{b}$$

其中  $\mathbf{b} \neq \mathbf{0}$ , 而且  $\delta A$  充分小, 使

$$\frac{\|\delta A\|}{\|A\|} < \frac{1}{\text{cond}(A)}$$

则有

$$\frac{\|\delta\mathbf{x}\|}{\|\mathbf{x}\|} \leq \frac{\text{cond}(A)}{1 - \|A^{-1}\| \cdot \|\delta A\|} \left( \frac{\|\delta A\|}{\|A\|} + \frac{\|\delta\mathbf{b}\|}{\|\mathbf{b}\|} \right)$$





设  $A \in \mathbb{R}^{n \times n}$ ,  $A$  非奇异, 则有

①  $\text{cond}(A) \geq 1$ ,  $\text{cond}(A) = \text{cond}(A^{-1})$ ,

$$\text{cond}(\alpha A) = \text{cond}(A), \forall \alpha \in \mathbb{R}, \alpha \neq 0,$$

② 若  $A$  为正交阵, 则

$$\text{cond}_2(A) = 1$$

③ 若  $U$  为正交阵, 则

$$\text{cond}_2(A) = \text{cond}_2(AU) = \text{cond}_2(UA)$$

④ 设  $\lambda_1, \lambda_n$  分别是按模最大和最小的特征值, 则

$$\text{cond}(A) \geq \frac{|\lambda_1|}{|\lambda_n|}, \quad \text{若 } A = A^T \implies \text{cond}_2(A) = \frac{|\lambda_1|}{|\lambda_n|}$$



# 矩阵的条件数是否依赖于范数的选取?

$$A = \begin{pmatrix} 1 & 0 & 0 & 0 & \cdots & 0 \\ 1 & 1 & 0 & 0 & \cdots & 0 \\ 1 & 0 & 1 & 0 & \cdots & 0 \\ 1 & 0 & 0 & 1 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 1 & 0 & 0 & 0 & \cdots & 1 \end{pmatrix}$$

计算  $A$  的1-条件数和 $\infty$ -条件数:

## Matlab 求矩阵条件数的命令为 cond。

```
>> A = diag(ones(10,1)) + diag(ones(9,1),1);  
>> cond(A)
```

```
ans =  
13.2320
```

```
>> cond(A,'inf')
```

```
ans =  
20
```

```
>> cond(A,1)
```

```
ans =  
20
```

```
>> cond(A, 'fro')
```

```
ans =  
32.3265
```

$$A = \begin{bmatrix} 1 & 1 & 0 & 0 & \dots & 0 \\ 0 & 1 & 1 & 0 & \dots & 0 \\ 0 & \vdots & 1 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \dots & 0 \\ \vdots & \vdots & \vdots & \dots & 1 & 1 \\ 0 & 0 & 0 & \dots & \dots & 1 \end{bmatrix}$$

# 病态矩阵的误差?

## 定理

设  $A \in \mathbb{R}^{n \times n}$ ,  $b \in \mathbb{R}^n$ , 且  $A$  非奇异,  $b \neq 0$ ,  $x$  是方程组  $Ax = b$  的精确解,  $\tilde{x}$  是近似解,  $r = b - A\tilde{x}$ , 称为  $\tilde{x}$  的残量, 则有

$$\frac{1}{\text{cond}(A)} \frac{\|r\|}{\|b\|} \leq \frac{\|\tilde{x} - x\|}{\|x\|} \leq \text{cond}(A) \frac{\|r\|}{\|b\|}$$



## 例

考虑线性方程组

$$A\mathbf{x} = \begin{bmatrix} 0.913 & 0.659 \\ 0.457 & 0.330 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0.254 \\ 0.127 \end{bmatrix} = \mathbf{b}$$

考虑两个近似解

$$\tilde{\mathbf{x}}_1 = \begin{bmatrix} -0.0827 \\ 0.5 \end{bmatrix} \quad \text{和} \quad \tilde{\mathbf{x}}_2 = \begin{bmatrix} 0.999 \\ -1.001 \end{bmatrix}$$

它们的残差范数分别为

$$\|\mathbf{r}_1\|_1 = 2.1 \times 10^{-4} \quad \text{及} \quad \|\mathbf{r}_2\|_1 = 2.4 \times 10^{-2}.$$

## 注记

对病态矩阵，由于条件数很大，小的残差并不能说明解的误差小。

例：系数矩阵的所有特征值为

$\lambda_1 \approx 30.2887$ ,  $\lambda_2 \approx 3.858$ ,  $\lambda_3 \approx 0.8431$ ,  $\lambda_4 \approx 0.01015$ , 故而

$$\text{cond}_2(A) = \frac{\lambda_1}{\lambda_4} \approx 2984$$

若  $\mathbf{b} = (32, 23, 33, 31)^T$ ,  $\delta \mathbf{b} = (0.1, -0.1, 0.1, -0.1)^T$ ,  $\delta \mathbf{x} = (8.2, -13.6, 3.5, -2.1)^T$ , 所以 实际的相对误差为

$$\frac{\|\delta \mathbf{x}\|_2}{\|\mathbf{x}\|_2} \approx 8.198$$

而用上述定理估计的相对误差界为

$$\frac{\|\delta \mathbf{x}\|_2}{\|\mathbf{x}\|_2} \leq 2984 \frac{\|\delta \mathbf{b}\|_2}{\|\mathbf{b}\|_2} = 9.943.$$

相对误差放大了两千多倍。

例

*Hilbert*矩阵是一个著名的病态矩阵，对称正定，

$$H_n = \begin{pmatrix} 1 & \frac{1}{2} & \cdots & \frac{1}{n} \\ \frac{1}{2} & \frac{1}{3} & \cdots & \frac{1}{n+1} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{n} & \frac{1}{n+1} & \cdots & \frac{1}{2n-1} \end{pmatrix}$$

$n$	$\lambda_{\max}$	$\lambda_{\min}$	$\kappa_2(H_n)$
5	1.6	$3.3 \times 10^{-6}$	$4.8 \times 10^5$
10	1.8	$1.1 \times 10^{-13}$	$1.6 \times 10^{13}$
15	1.8	$3.0 \times 10^{-21}$	$6.1 \times 10^{20}$
20	1.9	$7.8 \times 10^{-29}$	$2.5 \times 10^{28}$
25	2.0	$1.9 \times 10^{-36}$	$1.0 \times 10^{36}$

- ① 矩阵的条件数与矩阵的行列式值没有必然的联系。如

$$D_n = \text{diag}(10^{-1}, 10^{-1}, \dots, 10^{-1}) \in \mathbb{R}^{n \times n}$$

其条件数  $\text{cond}_p(D_n) = 1, p = 1, 2, \infty$ 。所以  $D_n$  是一个非常好的矩阵，但是  $\det D_n = 10^{-n}$ 。

- ② 对于病态的线性方程组，求解时必须非常小心，否则得不到满意的数值解。

方程组

$$H_4 \mathbf{x} = \left( \frac{25}{12}, \frac{77}{60}, \frac{57}{60}, \frac{319}{420} \right)^T$$

的准确解是  $\mathbf{x} = (1, 1, 1, 1)^T$ 。如果我们用3位和5位十进制舍入运算的消去法求解，得到的解分别是

$$(0.988, 1.42, -0.428, 2.10)^T$$

$$(1.0000, 0.99950, 1.0017, 0.99900)^T.$$

## 定理

$A \in \mathbb{R}^{n \times n}$  非奇异, 则

$$\min \left\{ \frac{\|\delta A\|_2}{\|A\|_2} : A + \delta A \text{ 奇异} \right\} = \frac{1}{\text{cond}_2(A)}$$

## 注记

粗略地说, 如果用主元素法或平方根法( $A$  对称正定阵)解方程组  $Ax = b$ , 设  $A$  和  $b$  的元素准确到  $s$  位数字, 且  $\text{cond}(A) \approx 10^t$ , 其中  $t < s$ , 则计算解向量的分量大约有  $s - t$  位数字的准确度。

# 预处理

- 对原方程作某些预处理, 可望降低系数矩阵的条件数。因为  $\text{cond}(\alpha A) = \text{cond}(A)$ , 所以不能通过将每一个方程乘上相同的常数  $\alpha$  来达到这个目标。

可考虑将矩阵的每一行和每一列分别乘上不同的常数。亦即找可逆的对角阵  $D_1$  和  $D_2$ , 将方程组  $Ax = b$  化为

$$D_1 A D_2 y = D_1 b, x = D_2 y$$

这称为矩阵的平衡问题。理论上最好选择对角阵  $\bar{D}_1$  和  $\bar{D}_2$  满足

$$\text{cond}(\bar{D}_1 A \bar{D}_2) = \min \text{cond}(D_1 A D_2)$$

其中的  $\min$  是对所有  $D_1$  和  $D_2$  属于可逆对角阵集合取的。

# 预处理矩阵的平衡问题

$D_1$  是平衡每个方程, 而  $D_2$  是平衡未知数的。令  $D_2 = I$ , 称为行平衡。 $D_1$  的选择可使  $D_1 A$  每一行的  $\infty$ -范数大体相等, 这可避免消元过程中小数与大数的相加。

如果令  $D_1 = I$ , 这时候称为列平衡,  $D_2$  的选择可使  $AD_2$  的每一列的  $\infty$ -范数大体相等。

从行平衡和列平衡的想法出发, 要求矩阵  $A$  的所有行和列的范数大致相等。但是为了求  $D_1$  和  $D_2$  的元素, 需要求解未知量为  $2n - 2$  的非线性方程组, 所需计算量比原问题本身的计算量要多。

## 预处理示例

方程组

$$\begin{pmatrix} 10 & 100000 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 100000 \\ 2 \end{pmatrix}$$

的准确解为  $x_1 = 1.0010001 \dots$ ,  $x_2 = 0.99989998 \dots$ 。

引入  $D_1 = \text{diag}(10^{-5}, 1)$ , 平衡后方程为

$$\begin{pmatrix} 0.0001 & 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$$

如果都用 3 位十进制的列主元消去法求解, 原方程组解得

$\mathbf{x} = (0.00, 1.00)^T$ , 而后一方程组解得  $\mathbf{x} = (1.00, 1.00)^T$ 。前一方程系数矩阵的 2-条件数为  $1 \times 10^5$ , 而后一方程系数矩阵的 2-条件数为 2.6184。