线性方程组求解——Gauss 消去法

回代, 列主元, 全主元

湘潭大学, 数学与计算科学学院，21 级王艺博

一、问题描述

对于线性方程组

$$Ax = b, \quad A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix}, \quad b = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}$$

A 为非奇异矩阵，求向量 x

二、Gauss 消去法

思路： 系数矩阵 $\xrightarrow{\text{逐步消元}}$ 上三角 $\xrightarrow{\text{回代算法}}$ 解

相对误差估计： x 为真解, $\tilde{x}$ 为近似解

$$\frac{\|x-\tilde{x}\|_\infty}{\|x\|_\infty} \leq \frac{\text{cond}_\infty(A)}{1-\text{cond}_\infty(A)\frac{\|\delta A\|_\infty}{\|A\|_\infty}} \cdot \left[1.01(n^3+3n^2)\rho\mu\right]$$

其中 $\rho = \frac{1}{\|A\|_\infty} \max_{i \leq i,j,k \leq n} \left|a_{ij}^{(k)}\right|$, μ 表示计算机上的单位舍入误差或截断误差

三、算法

3.1 回代算法

情景：$Ax=b \quad a_{ii\neq 0}(i=1,2,\dots,n)$

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

1：$x_n=\frac{b_n}{a_{nn}}$
2：$x_k=\frac{b_k-a_{k,k+1}x_{k+1}-\cdots-a_{kn}x_n}{a_{kk}}$

算法 1 回代算法 (解上三角)

```
function [X] = reg_utm(A,b)
```

Input: 系数矩阵 A, 右端向量 b;
Output: 数值解 X;

$x_n=\frac{b_n}{a_{nn}}$;
for $k=[n-1,n-2,\dots,1]$ **do**

$$t=\begin{bmatrix}a_{k,k+1} & a_{k,k+2} & \cdots & a_{kn}\end{bmatrix}\begin{bmatrix}x_{k+1}\\ x_{k+2}\\ \vdots\\ x_n\end{bmatrix}$$

$x_k=\frac{b_k-t}{a_{kk}}$;
end for
输出 X;

3.2 耿直消元法

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

简记为 $A^{(1)}x=b^{(1)}$, 其中 $A^{(1)}=A, \quad b^{(1)}=b$

若 $a_{11}^{(1)}\neq 0$，令 $l_{i1}=\frac{a_{i1}^{(1)}}{a_{11}^{(1)}}$, 令 $-l_{i1}$ 乘第 1 行加到第 i 行 $(i=2,3,\cdots,n)$ 并保留第 1 个方程. 此时 $A^{(1)}\to A^{(2)}, b^{(1)}\to b^{(2)}$

$a_{ij}^{(2)} = a_{ij}^{(1)} - l_{i1}a_{1j}^{(1)}(i,j = 2,3,\cdots,n) \quad b_i^{(2)} = b_i^{(1)} - l_{i1}b_1^{(1)}(i = 2,3,\cdots,n)$

若 $a_{kk}^{(k)} \neq 0$，令 $l_{ik} = \frac{a_{ik}^{(k)}}{a_{kk}^{(k)}}$，令 $-l_{ik}$ 乘第 k 行加到第 i 行 $(i = k+1, k+2, \cdots, n)$

$$a_{ij}^{(k+1)} = a_{ij}^{(k)} - l_{ik}a_{kj}^{(k)}(i,j = k+1, k+2, \cdots, n) \quad b_i^{(k+1)} = b_i^{(k)} - l_{ik}b_k^{(k)}(i = k+1, k+2, \cdots, n)$$

算法 2 高斯消去法耿直版

```
function [X,Ae,be] = gsem_base(A,be)
```

Input: 系数矩阵 A, 右端列向量 b;

Output: 数值解 X; 消元得到的上三角矩阵 Ae 和对应的右端项 be;

$A1 = \begin{bmatrix} A & b \end{bmatrix}$;

for $i = 1:1:n-1$ **do**

  if $a_{kk}^k = 0$ **then**

    结束循环，不能使用耿直法求解;

  else

    for $k = i+1:1:n$ **do**

      $l_{ik} = \frac{a_{ki}}{a_{ii}}$;

      第 k 行 $= \begin{bmatrix} a_{k,i} & a_{k,i+1} & \cdots & a_{k,n} & b_k \end{bmatrix} - l_{ik} \begin{bmatrix} a_{i,i} & a_{i,i+1} & \cdots & a_{i,n} & b_i \end{bmatrix}$

    end for

  end if

end for

得到 Ae,be

```
function [X] = reg_utm(Ae,be)
```

输出 X;

缺点:

$a_{kk}^{(k)} \neq 0$ 若为 0，不能使用

$\left|a_{kk}^{(k)}\right| \approx 0$ 或相对其他元素较小时，舍入误差增加，可能导致误差剧增

3.3 列主元消去法

进行每一步消元前，先找到列中最大元素所在的那一行，并与原来的行交换位置

3.4 全主元消去法

进行每一步消元前，先找到剩余矩阵中最大的元素，做对应的行交换与列交换，由于进行了列交换，最后要把 X 的位置对应回去

四、北太天元源程序

回代算法

```
function [X] = reg_utm(A,b)
% 解上三角方程组 回代算法
% A : 系数矩阵 必须为上三角
% b : 右端常数
% X : 求得的解向量
% 使用要求: A的对角线处元素不为0
if max(ismember(diag(A),0)) == 1
  fprintf('\n Error:使用reg_utm时 对角线上有0元素, 不能使用该函数\n\n');
else
    n = length(b);
    X = zeros(n,1);
    X(n) = b(n)/A(n,n);
    for k=n-1:-1:1
        t = A(k,[k+1:n])*X(k+1:n);
        X(k) = (b(k)-t)/A(k,k);
    end
end
end
```

将上述代码保存为 reg_utm.m 文件。

Gauss 消去法耿直版

```
function [X,Ae,be] = gsem_base(A,b)
% Gauss消去法耿直版
% A : 系数矩阵
% b : 右端常数 列向量
% X : 求得的解向量
% Ae: 得到的上三角矩阵
% be: 对应的右端项
% 使用要求, A1(k,k)在计算过程中不出现0
% 缺点: 当A1(k,k)较小或近似于0时, 计算误差可能增大很多
A1 = [A,b];% A和b的增广矩阵
n = length(A);
n1 = length(A1);
% 系数矩阵 化为上三角的过程如下
t = 0;
for i = 1:n-1
```

```
    if A1(i,i) != 0
        % 对每一行进行操作
          for k = i+1:1:n
            lik = A1(k,i)/A1(i,i); % 算子
            A1(k,i:n1) = A1(k,i:n1)-lik*A1(i,i:n1);
          end
    else
        t = 1;
        fprintf('出现零元素，不能使用该函数');
        break;
    end
end
% 得到上三角后 进行回代
if t == 0
Ae = A1(:,1:n);
be = A1(:,n1);
X  = reg_utm(A,b);
end
end
```

将上述代码保存为 gsem_base.m 文件。

列主元消去法

```
function [X,Ae,be] = gsem_column(A,b)
% 列主元消去法
% A : 表示系数矩阵
% b : 表示右端常数 [列向量]
% X : 求得的解向量
% Ae: 得到的上三角矩阵
% be: 对应的右端项
% 避免了Gauss消去法中出现0的情况
A1 = [A,b];% A和b的增广矩阵
n = length(A);
n1 = length(A1);
% 系数矩阵 化为上三角的过程如下
for i = 1:n-1
    %[a,s]= max(A1(i:n,i))
    [~,s]= max(A1(i:n,i)); % s表示对应的位置,由于a用不到, 用~替代
```

```
    s = s+i-1; % 由于s是相A对位置, 故有此步
    A1([i,s],:) = A1([s,i],:); % 对应行交换位置
    % 对每一行进行操作
     for k = i+1:1:n
       lik = A1(k,i)/A1(i,i); % 算子
       A1(k,i:n1) = A1(k,i:n1)-lik*A1(i,i:n1);
     end
end
% 得到上三角后 进行回代
Ae = A1(:,1:n);
be = A1(:,n1);
X = reg_utm(Ae,be);
end
```

将上述代码保存为 gsem_column.m 文件。

全主元消去法

```
function [X,Ae,be] = gsem_complete(A,b)
% 全主元消去法
% A : 系数矩阵
% b : 右端常数 [列向量]
% X : 求得的解向量
% Ae: 得到的上三角矩阵
% be: 对应的右端项
% 比列主元更加稳定了
A1 = [A,b];  %A和b的增广矩阵
n = length(A);
n1 = length(A1);
X = zeros(n,1);
t = 1:n; % t 用于修正解的位置与原方程不匹配
% 系数矩阵化为上三角的过程如下
   for i = 1:n-1
        A = A1(i:n,i:n);
       [m,a,b] = max_loc(A); %这里不用A1, 避免误判b中元素
       % a,b是相对的位置
       a = a+i-1;
       b = b+i-1;
       A1([i,a],:) = A1([a,i],:); % 对应行交换位置
```

```
        A1(:,[i,b]) = A1(:,[b,i]); % 对应列交换位置
        [t(i),t(b)]= deal(t(b),t(i));
        % 对每一行进行操作
          for k = i+1:1:n
            lik = A1(k,i)/A1(i,i); % 算子
            A1(k,i:n1) = A1(k,i:n1)-lik*A1(i,i:n1);
         end
    end
% 得到上三角后进行回代
    Ae = A1(:,1:n);
    be = A1(:,n1);
    X([t])= reg_utm(A,be); %使解匹配原方程位置
end
```

将上述代码保存为 gsem_complete.m 文件。

```
function [m,a,b] =max_loc(A)
% 获取矩阵中最大元素的位置
% max: 矩阵中最大值
% [a,b] 最大值在矩阵中的位置
% 判断矩阵类型
    [r,l] = size(A);
    if r == 1  % 行向量
        a = 1;
        [m,b] = max(A);
    elseif l == 1 % 列向量
        b = 1;
        [m,a] = max(A);
    else
        [M,I] = max(A);
        [m,b] = max(M); % b = 最大元素所在的列
        a = I(b); % a = 最大元素所在的行
    end
end
```

将上述代码保存为 max_loc.m 文件。

五、数值算例

例 1

$$\begin{bmatrix} 10 & 1 & 2 & 3 & 4 \\ 1 & 9 & -1 & 2 & -3 \\ 2 & -1 & 7 & 3 & -5 \\ 3 & 2 & 3 & 12 & -1 \\ 4 & -3 & -5 & -1 & 15 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} = \begin{bmatrix} 12 \\ -27 \\ 14 \\ -17 \\ 12 \end{bmatrix}$$

用 *Gauss* 消去法求出解向量 X

```
clc;clear all;
A = [10 1 2 3 4; 1 9 -1 2 -3; 2 -1 7 3 -5; 3 2 3 12 -1; 4 -3 -5 -1 15];
b = [12; -27; 14; -17; 12];

X1 = gsem_complete(A,b)
X2 = gsem_column(A,b)
X3 = gsem_base(A,b)
```

将上述代码保存为 `leqs_test.m`

运行后得到

$$X1 = \begin{bmatrix} 1.0000 \\ -2.0000 \\ 3.0000 \\ -2.0000 \\ 1.0000 \end{bmatrix} X2 = \begin{bmatrix} 1.0000 \\ -2.0000 \\ 3.0000 \\ -2.0000 \\ 1.0000 \end{bmatrix} X3 = \begin{bmatrix} 1.0000 \\ -2.0000 \\ 3.0000 \\ -2.0000 \\ 1.0000 \end{bmatrix}$$