Chap Seven

ShangXiaojin

2022 年 12 月 14 日

目录

1 Iterative Techniques for Solving Linear Systems

1.1 Ex

- $$\begin{aligned} 4x_1 + x_2x_3 + x_5 =& 6 \\ -x_1 - 3x_2 + x_3 + x_4 =& 6 \\ 2x_1 + 1x_2 + 5x_3 - x_4 - x_5 =& 6 \\ -x_1 - x_2 - x_3 + 4x_4 =& 6 \\ 2x_2 - x_3 + x_4 + 4x_5 =& 6 \end{aligned}$$

- $$\begin{aligned} 4x_1 - x_3 - x_4 &= 0 \\ -x_1 + 4x_2 - x_3 - x_5 &= 5 \\ -x_2 + 4x_3 - x_6 &= 0 \\ -x_1 - 4x_4 - x_5 &= 6 \\ -x_2 - x_4 + 4x_5 - x_6 &= -2 \\ -x_3 - x_5 + 4x_6 &= 6 \end{aligned}$$

using Jocabi GaussSeidel SOR method with w = 1 and 1.3 to solve this linear systems above it.

1.2 Code

1.2.1 main code

```
A1 =[4 1 1 0 1
    -1 -3 1 1 0
    2 1 5 -1 -1
```

```
    -1 -1 -1 4 0
    0 2 -1 1 4];
b1 = [6 6 6 6 6]';
A2 = [4 -1 0 -1 0 0
    -1 4 -1 0 -1 0
    0 -1 4 0 0 -1
    -1 0 0 -4 -1 0
    0 -1 0 -1 4 -1
    0 0 -1 0 -1 4 ];
b2 = [0 5 0 6 -2 6]';
Jacobi(A1,b1,[0 0 0 0 0]',100,0.001)
GaussSeidel(A1,b1,[0 0 0 0 0]',100 ,0.001)
SOR(A1,b1,[0 0 0 0 0]',100 ,0.001,1)
SOR(A1,b1,[0 0 0 0 0]',100 ,0.001,1.3)
Jacobi(A2,b2,[0 0 0 0 0 0]',100,0.001)
GaussSeidel(A2,b2,[0 0 0 0 0 0]',100,0.001)
SOR(A2,b2,[0 0 0 0 0 0]',100,0.001,1)
SOR(A2,b2,[0 0 0 0 0 0]',100,0.001,1.3)
```

1.2.2 function

- Jacobi

```
function y = Jacobi(A,b,XO,N,TOL)
k = 1;
n = size(b,1);
x = zeros(n,1);
while k<=N
    for i = 1:n
        x(i,1) = b(i,1);
        for j = 1:i-1
            x(i) = x(i) - A(i,j)*XO(j,1);
        end
        for j = i+1:n
            x(i) = x(i) - A(i,j)*XO(j,1);
```

```
        end
        x(i) = x(i)/A(i,i);
    end
    if max(abs(x - XO))<TOL
        y = x;
        break
    else
        k = k+1;
        for i = 1:n
            XO(i,1) = x(i,1);
        end
    end
end
end
```

- GaussSeidel

```
function y = GaussSeidel(A,b,XO,N,TOL)
k = 1;
n = size(b,1);
x = zeros(n,1);
while k<=N
    for i = 1:n
        x(i,1) = b(i,1);
        for j = 1:i-1
            x(i,1) = x(i,1) - A(i,j)*x(j,1);
        end
        for j = i+1:n
            x(i,1) = x(i,1) - A(i,j)* XO(j,1);
        end
        x(i,1) = x(i,1)/A(i,i);

    end
    if max(abs(x-XO))<TOL
```

```
            y = x;
            break
        else
            k = k+1;
            for i = 1:n
                XO(i,1) = x(i,1);
            end
    end
end
end
```

- SOR

```
function y =  SOR(A,b,XO,N,TOL,w)
k = 1;
n = size(b,1);
x = zeros(n,1);
temp = zeros(n,1);
while k <= N
    for i = 1:n
        temp(i,1) = b(i,1);
        for j = 1: i-1
            temp(i) = temp(i) - A(i,j)*x(j,1);
        end
        for j = i+1:n
            temp(i) = temp(i) - A(i,j)*x(j,1);
        end
        temp(i,1) = temp(i,1)/A(i,i);
        x(i,1) = (1-w)*XO(i,1) + temp(i,1);
    end
    if max(abs(x-XO))<TOL
        y = x;
        break
    else
```

```
        k = k+1;
        for i = 1:n
            XO(i,1) = x(i,1);
        end
    end
end
end
```

1.3 Ans

1.3.1 Ex1

	Jacobi	GaussSeidel	SOR with w=1	SOR with w=1.3
x	$\begin{pmatrix} 0.7871 \\ -1.0030 \\ 1.8660 \\ 1.9124 \\ 1.9896 \end{pmatrix}$	$\begin{pmatrix} 0.7867 \\ -1.0027 \\ 1.8663 \\ 1.9126 \\ 1.9898 \end{pmatrix}$	$\begin{pmatrix} 0.7867 \\ -1.0027 \\ 1.8663 \\ 1.9126 \\ 1.9898 \end{pmatrix}$	$\begin{pmatrix} 0.8142 \\ -1.0713 \\ 1.2842 \\ 1.3514 \\ 1.5528 \end{pmatrix}$

1.3.2 Ex2

	Jacobi	GaussSeidel	SOR with w=1	SOR with w=1.3
x	$\begin{pmatrix} -0.0144 \\ 1.4147 \\ 0.7703 \\ -1.4720 \\ -0.0976 \\ 1.6683 \end{pmatrix}$	$\begin{pmatrix} -0.0144 \\ 1.4146 \\ 0.7707 \\ -1.4720 \\ -0.0973 \\ 1.6684 \end{pmatrix}$	$\begin{pmatrix} -0.0144 \\ 1.4146 \\ 0.7707 \\ -1.4720 \\ -0.0973 \\ 1.6684 \end{pmatrix}$	$\begin{pmatrix} -0.0212 \\ 1.0056 \\ 0.4245 \\ -1.1163 \\ -0.1748 \\ 1.2019 \end{pmatrix}$