Homework 5 – A Report on Iterative Techniques in Matrix Algebra

Xiaolong Liu, 201900170025
22ed.Dec.2021
Shandong University, Taishan College

Abstract

In this report, we use Gaussian method and Cholesky decomposition and the Jacobi and Gauss-Seidel intertive method to compute x where $Ax = B$.

Contents

1 Introductions **2**

2 Algorithm Description & Mathematical Derivation **2**

2.1 Gaussian method and Cholesky decomposition 2

2.2 Jacobi and Gauss-Seidel intertive method 2

3 Codes and Numerical Experiments **3**

3.1 Gaussian method and Cholesky decomposition 3

3.2 Jacobi and Gauss-Seidel intertive method 5

1 Introductions

In this report, we use Gaussian method and Cholesky decomposition and the Jacobi and Gauss-Seidel intertive method to compute x where $Ax = B$.

2 Algorithm Description & Mathematical Derivation

2.1 Gaussian method and Cholesky decomposition

We omitted these since these are just simply linear algebra.

2.2 Jacobi and Gauss-Seidel intertive method

Theorem 2.2.1 (Jacobi)**.** *Compute* $\boldsymbol{x}$ *with* $\boldsymbol{Ax} = \boldsymbol{b}$ *for* x_i *to obtain*

$$x_i = \sum_{j=1, j\neq i}^{n} \left(-\frac{a_{ij}x_j}{a_{ii}}\right) + \frac{b_i}{a_{ii}}, i = 1, ..., n.$$

For each $k \geq 1$ *and for* $\boldsymbol{x}^{(k)}$ *we have*

$$x_i^{(k)} = \frac{1}{a_{ii}}\left[\sum_{j=1, j\neq i}(-a_{ij}x_j^{(k-1)}) + b_i\right], i = 1, ..., n.$$

Theorem 2.2.2 (Gauss-Seidel)**.** *Compute* $\boldsymbol{x}$ *with* $\boldsymbol{Ax} = \boldsymbol{b}$ *for* x_i *to obtain*

$$x_i = \sum_{j=1, j\neq i}^{n} \left(-\frac{a_{ij}x_j}{a_{ii}}\right) + \frac{b_i}{a_{ii}}, i = 1, ..., n.$$

For each $k \geq 1$ *and for* $\boldsymbol{x}^{(k)}$ *we have*

$$x_i^{(k)} = \frac{1}{a_{ii}}\left[-\sum_{j=1}^{i-1}(a_{ij}x_j) - \sum_{j=i+1}^{n}(a_{ij}x_j^{(k-1)}) + b_i\right], i = 1, ..., n.$$

3 Codes and Numerical Experiments

3.1 Gaussian method and Cholesky decomposition

Problem 3.1.1. *Given a equation $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ where* $\boldsymbol{A} = \begin{pmatrix} 100 & 2 & 4 & 7 \\ 2 & 100 & 5 & 19 \\ 4 & 5 & 100 & 5 \\ 7 & 19 & 5 & 100 \end{pmatrix}$ *and* $\boldsymbol{b} = \begin{pmatrix} 2 \\ 3 \\ 4 \\ 5 \end{pmatrix}$, *we will use the Gaussian method and Cholesky decomposition to compute $\boldsymbol{x}$.*

Solution. Consider the following function:

```
>> format long
function x=GS(A,b)
[n,~] = size(A);
x=zeros(n,1);

for j = 1:n-1
    for i = j+1:n
        mul = A(i,j)/A(j,j);
        A(i,:)= A(i,:)-mul*A(j,:);
        b(i)= b(i)-mul*b(j);
    end
end


for i=n:-1:1
    sum=0;
    for j=n:-1:i+1
        sum=sum+x(j)*A(i,j);
    end
    x(i)=(b(i)-sum)/A(i,i);
end
end
```

And use the follows codes:

```
>> A = [100  2   4   7;
   2   100   5   19;
   4   5   100   5 ;
   7   19   5   100];

b = [2;3;4;5];
>> GS(A,b)

ans =

   0.015119617688382
   0.019639842048166
   0.036243328673106
   0.043397890339006
```

Then $\boldsymbol{x} = \begin{pmatrix} 0.015119617688382 \\ 0.019639842048166 \\ 0.036243328673106 \\ 0.043397890339006 \end{pmatrix}$. Well done.

Now we will use Cholesky decomposition. Consider the follows function:

```
function [x,res]=CD(A,b)
AA=A;[~,n]=size(A);
y=zeros(n,1);x=zeros(n,1);
for k=1:1:n
    A(k,k)=sqrt(A(k,k));
    A(1:k-1,k)=0;
    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
U=A';L=A;
    y(1,1)=b(1,1)/L(1,1);
for i=2:n
    s=0;
    for j=1:i-1
        s=s+y(j,1)*L(i,j);
    end
    y(i,1)=(b(i,1)-s)/L(i,i);
end
x(n,1)=y(n,1)/U(n,n);
for i=n-1:-1:1
    s=0;
    for j=i+1:n
        s=s+U(i,j)*x(j,1);
    end
    x(i,1)=(y(i,1)-s)/U(i,i);
end
res=norm(b-AA*x);
```

Then we have:

```
>> CD(A,b)

ans =

   0.015119617688382
   0.019639842048166
   0.036243328673106
   0.043397890339006
```

So $\boldsymbol{x} = \begin{pmatrix} 0.015119617688382 \\ 0.019639842048166 \\ 0.036243328673106 \\ 0.043397890339006 \end{pmatrix}$. Well done. $\square$

3.2 Jacobi and Gauss-Seidel intertive method

Problem 3.2.1. *We will use the Jacobi and Gauss-Seidel intertive method to compute* $\boldsymbol{x}$ *by construct* $\boldsymbol{A}, \boldsymbol{b}$*.*

Solution. Use the following function to find $\boldsymbol{A}, \boldsymbol{b}$:

```
function [A,b]=DD(n)
   T=zeros(n-1,n-1);
   for i=1:n-1
       T(i,i)=2;
   end
   for i=1:n-2
       T(i,i+1)=-1;
       T(i+1,i)=-1;
   end
   I=eye(n-1,n-1);
   k=(n-1)^2;
   A=zeros(k,k);
   b=ones(k,1);
   j=1;
   for i=1:n-1
       A(j:j+n-2,j:j+n-2)=T+2*I;
       j=j+n-1;
   end
   j=1;
   for i=1:n-2
       A(j:j+n-2,j+n-1:j+2*n-3)=-I;
       A(j+n-1:j+2*n-3,j:j+n-2)=-I;
       j=j+n-1;
   end
end
```

When we use the Gauss Seidel method, we use the following code.

```
function []=GSS(n,eps)
   [A,b]=DD(n);
   D=diag(diag(A));L=-tril(A,-1);
   U=-triu(A,1);x=zeros((n-1)^2,1);k=0;maxit=100000;
   B=(D-L)\U;g=(D-L)\b;
   tic
   while 1
       k=k+1;
       x=B*x+g;
       res=b-A*x;
       if norm(res,2)<eps
           break
       elseif k>=maxit
           fprintf('The times to the maximum:\n');
           break
       else
           continue;
       end
   end
   toc
end
```

Then we have

```
>> GSS(25,1e-4)
```

```
    0.079250
```

The jacobi's method as follows:

```
function []=JB(n,eps)
   [A,b]=DD(n);
   D=diag(diag(A));L=-tril(A,-1);U=-triu(A,1);x=zeros((n-1)^2,1);k=0;
       maxit=100000;
   B=inv(D)*(L+U);
   g=inv(D)*b;
   tic
   while 1
       k=k+1;
       x=B*x+g;
       res=b-A*x;
       if norm(res,2)<eps
           break
       elseif k>=maxit
           fprintf('The times to the maximum:\n');
           break
       else
           continue;
       end
   end
   toc
end
```

Then we have

```
>> JB(25,1e-4)
    0.100129
```

Well done. □

References

[1] Richard L. Burden, J. Douglas Faires, Annette M. Burden, *Numerical Analysis (Tenth Edition)*, Cengage Learning, 2014.