数值分析编程实验三

2018011324

计82

尤艺霖

实验题目：（截取自下发文件）

实验三

第三章上机题6: 编程生成 Hilbert 矩阵 $\mathbf{H}_n$ (见例 3. 4) , 以及 n 维向量 $\mathbf{b}=\mathbf{H}_n\mathbf{x}$, 其中 $\mathbf{x}$ 为所有分量都是 1 的向量. 用 Cholesky 分解算法求解方程 $\mathbf{H}_n\mathbf{x}=\mathbf{b}$， 得到近似解 $\hat{\mathbf{x}}$, 计算残差 $\mathbf{r}=\mathbf{b}-\mathbf{H}_n\hat{\mathbf{x}}$ 和误差 $\Delta\mathbf{x}=\hat{\mathbf{x}}-\mathbf{x}$ 的 ∞-范数.

(1) 设 $n=10$, 计算 $||\mathbf{r}||_\infty$ 、$||\Delta\mathbf{x}||_\infty$.

(2) 在右端项上施加 10^{-7} 的扰动然后解方程组, 观察残差和误差的变化情况.

(3) 改变 n 的值为 8 和 12 , 求解相应的方程, 观察 $||\mathbf{r}||_\infty$ 、$||\Delta\mathbf{x}||_\infty$ 的变化情况. 通过这个实验说明了什么问题?

思路：

按照书上的cholesky分解和后面的解方程思路完成算法即可，第二题的扰动我默认是每一项都加了扰动。

以及存在一个问题：第二问的扰动是否应该加入残差的计算。我的理解是计算残差时不应该加上扰动，因为扰动相对于残差来说太大了，会影响对残差本身的观察。

实验结果：

n	残差的无穷范数	误差的无穷范数	扰动后残差的无穷范数	扰动后误差的无穷范数
10	2.2204×10^{-16}	4.4459×10^{-4}	4.4409×10^{-16}	0.7007
8	2.2204×10^{-16}	4.1154×10^{-7}	2.2204×10^{-16}	0.02162
12	4.4409×10^{-16}	0.3358	4.4409×10^{-16}	23.6202

从实验结果中可以观察到在引入扰动之后，误差的无穷范数明显增大了若干个数量级，残差的无穷范数变化不大。

而这个H矩阵在上课的时候是老师一再强调的一个病态矩阵。因而这个问题中误差对扰动很敏感。

而随着n的增大，也可以观察到误差的无穷范数明显增大，而残差的无穷范数变化不大。

代码：

代码置于src目录下problem36.m中：

```
format long;
n=10;%这里可以改变
```

```
x=ones(n,1);
oldH=ones(n);
for i=1:n
    for j=1:n
        oldH(i,j)=1/(i+j-1);%计算H矩阵
    end
end
H=oldH;
b=oldH*x;
for i=1:n%cholesky分解
    for j=1:(i-1)
        H(i,i)=H(i,i)-H(i,j)*H(i,j);
    end
    H(i,i)=sqrt(H(i,i));
    for j=(i+1):n
        for k=1:(i-1)
            H(j,i)=H(j,i)-H(i,k)*H(j,k);
        end
        H(j,i)=H(j,i)/H(i,i);
    end
end
tx=zeros(n,1);
mx=zeros(n,1);
for i=1:n%第一次前代
    mx(i,1)=b(i,1);
    for j=1:(i-1)
        mx(i,1)=mx(i,1)-H(i,j)*mx(j,1);
    end
    mx(i,1)=mx(i,1)/H(i,i);
end
for i=n:-1:1%第二次回代
    tx(i,1)=mx(i,1);
    for j=n:-1:(i+1)
        tx(i,1)=tx(i,1)-H(j,i)*tx(j,1);
    end
    tx(i,1)=tx(i,1)/H(i,i);
end
dx=max(abs(tx-x));
r=max(abs(b-oldH*tx));
dx
r
diff=ones(n,1)/(1e7);%扰动
H=oldH;
b=oldH*x+diff;
for i=1:n%与上面的一样
    for j=1:(i-1)
        H(i,i)=H(i,i)-H(i,j)*H(i,j);
    end
    H(i,i)=sqrt(H(i,i));
    for j=(i+1):n
        for k=1:(i-1)
            H(j,i)=H(j,i)-H(i,k)*H(j,k);
        end
```

```
        H(j,i)=H(j,i)/H(i,i);
    end
end
tx=zeros(n,1);
mx=zeros(n,1);
for i=1:n
    mx(i,1)=b(i,1);
    for j=1:(i-1)
        mx(i,1)=mx(i,1)-H(i,j)*mx(j,1);
    end
    mx(i,1)=mx(i,1)/H(i,i);
end
for i=n:-1:1
    tx(i,1)=mx(i,1);
    for j=n:-1:(i+1)
        tx(i,1)=tx(i,1)-H(j,i)*tx(j,1);
    end
    tx(i,1)=tx(i,1)/H(i,i);
end
dx2=max(abs(tx-x));
r2=max(abs(b-oldH*tx));
dx2
r2
```