

实验报告5

PB15081576 蔡文韬

实验目的

分别编写Jacobi迭代及Gauss-Seidel迭代的通用程序。

实验分析

按照书上的公式是实现本次实验的核心代码：

Jacob迭代：

```
def CalRg(A, B):
    for i in range(len(A)):
        b=A[i][i]
        A[i][i]=0
        for j in range(len(A[0])):
            A[i][j]=A[i][j]/(-b)
        B[i]=B[i]/b
    return A,B # 计算R, g

def Linf(x1, x2):
    return sum([abs(x1[i]-x2[i]) for i in range(len(x1))])

def Jacob(A, B, e, x1):
    R,g=CalRg(A,B)
    i = 0
    x2 = x1
    while i==0 or Linf(x1,x2)>e :
        x1=x2
        x2=dot(R, x1)+g # dot: 矩阵乘法
        i+=1
    return x2,i
```

Gauss-Seidel迭代：

```
def GaussSeidel(A,B,e,x1):
    i = 0
    R,g=CalRg(A,B)
    x2=dot(R, x1)+g
    while Linf(x1,x2)>e:
        x1=x2
        x2=dot(R, x1)+g
```

```
        i+=1
    return x2,i
```

实验结果

Jacob: 迭代38次

x1
-2.892330029875e-01
3.454354885048e-01
-7.128112966542e-01
-2.206086019795e-01
-4.304000556149e-01
1.543084810206e-01
-5.782201273198e-02
2.010538605875e-01
2.902288017534e-01

G-S : 迭代17次

x1
-2.892320250343e-01
3.454371074225e-01
-7.128109765954e-01
-2.206079582467e-01
-4.303998750673e-01
1.543097530973e-01
-5.782213188002e-02
2.010540243670e-01
2.902288422246e-01

实验小结

结果分析：

- Gauss-Seidel的迭代次数明显低于Jacob的，如果希望收敛快，则选择Gauss-Seidel.

- Gauss-Seidel的优势在于：计算 x^{k+1} 时，使用直接更新的方法，既节省储存单元，又能使迭代加速
- G-S迭代收敛更快的结论不是恒成立的。在有的条件下，Jacob收敛，G-S却发散。

收获：

- 迭代法适用于大型稀疏线性方程组。
- G-S迭代在一般情况下比Jacob收敛更快。