科学计算引论作业(三)

谢悦晋　U202210333

Sept 28th, 2023

3.3 用改进的Cholesky分解法求解方程组

$$\begin{pmatrix} 5 & -4 & 1 & 0 \\ -4 & 6 & -4 & 1 \\ 1 & -4 & 6 & -4 \\ 0 & 1 & -4 & 5 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} 2 \\ -1 \\ -1 \\ 2 \end{pmatrix}$$

解：

Step 1. 计算 $d_1, \boldsymbol{L}$ 的第 1 列元素

$$d_1 = a_{11}, \quad l_{j1} = a_{j1}/d_1, \quad j = 2, 3, \cdots, 4.$$

$$\Rightarrow d_1 = 5, \begin{pmatrix} l_{11} \\ l_{21} \\ l_{31} \\ l_{41} \end{pmatrix} = \begin{pmatrix} -1 \\ -1 \\ 2 \end{pmatrix}$$

Step 2. 若 $\boldsymbol{D}, \boldsymbol{L}$ 的前 $j-1$ 列元素已计算,则计算 $\boldsymbol{D}, \boldsymbol{L}$ 的第 j 列元素($\boldsymbol{D}, \boldsymbol{L}$的结果在后面)

$$\begin{aligned} d_j &= a_{jj} - \sum_{k=1}^{j-1} l_{jk} v_{jk}, \quad v_{jk} = l_{jk} d_k, \\ l_{ij} &= \left(a_{ij} - \sum_{k=1}^{j-1} l_{ik} v_{jk} \right) / d_j, \quad i = j+1, j+2, \cdots, 4. \end{aligned}$$

Step 3. 求解下列系数矩阵为下三角形矩阵和上三角形矩阵的方程组

$$LY = b, \quad DL^{\mathrm{T}} X = Y.$$

获得原方程组的解 X.

```
Cholesky分解的下三角矩阵L:
[[ 1.          0.          0.          0.        ]
 [-0.8         1.          0.          0.        ]
 [ 0.2        -1.14285714  1.          0.        ]
 [ 0.          0.35714286 -1.33333333  1.        ]]
对角矩阵D:
[[5.         0.         0.         0.        ]
 [0.         2.8        0.         0.        ]
 [0.         0.         2.14285714 0.        ]
 [0.         0.         0.         0.83333333]]
线性方程组的解x:
[1. 1. 1. 1.]
```

Figure 1: 代码运行图

完整的Python代码如下：

```python
import numpy as np
def Cholesky_plus(matrix):
    w = matrix.shape[0]
    L = np.zeros((w,w))
    for i in range(w):
        L[i,i] = 1
    D = np.zeros((w,w))
    for i in range(w):
        D[i,i] = matrix[i,i] - np.dot(np.dot(L[i,:i],D[:i,:i]),L[i,:i].T)
        for j in range(i+1,w):
            L[j,i] = (matrix[j,i] - np.dot(np.dot(L[j,:i],D[:i,:i]),L[i,:i].T))/D[i,i]
    return L,D
def backward_substitution(L, D, b):
    y = np.dot(np.linalg.inv(L), b)
    x = np.dot(np.linalg.inv(np.dot(D, L.T)), y)
    return x
A = np.array([[5, -4, 1, 0], [-4, 6, -4, 1], [1, -4, 6, -4],[0, 1, -4, 5]])
b = np.array([2, -1, -1, 2])
L, D = Cholesky_plus(A)
x = backward_substitution(L, D, b)
print("Cholesky分解的下三角矩阵L:")
print(L)
print("对角矩阵D:")
print(D)
print("线性方程组的解x:")
print(x)
```
在 11ms 的 2023.10.05 12:33:30 执行

Figure 2: Source Code

3.4 用追赶法求解三对角型方程组

$$\begin{pmatrix} 2 & -1 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & -1 & 2 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

求$Ax = f$等价于求$\begin{cases} Ly = f \\ Ux = y \end{cases}$ 其中$f = (b_1, f_2, \ldots, f_n)^T$,故有:

$$\begin{bmatrix} 1 & & & & \\ p_2 & 1 & & & \\ & p_3 & 1 & & \\ & & \ddots & \ddots & \\ & & & p_n & 1 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ \vdots \\ f_n \end{bmatrix}$$

解得$\begin{cases} y_1 = f_1 \\ y_i = f_i - p_i y_{i-1} \quad (i = 2, \ldots, n) \end{cases}$

再由

$$\begin{bmatrix} q_1 & c_1 & & & \\ & q_2 & c_2 & & \\ & & \ddots & \ddots & \\ & & & q_{n-1} & c_{n-1} \\ & & & & q_n \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{n-1} \\ x_n \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_{n-1} \\ y_n \end{bmatrix}$$

解得 $\begin{cases} x_n = \dfrac{y_n}{q_n} \\ x_i = \dfrac{y_i - c_i x_{i+1}}{q_i} \quad (i = n-1, \ldots, 1) \end{cases}$

这是追赶法的主要过程，下面给出Python代码实现及运行结果：

```python
import numpy as np
def tridiagonal_solver(a, b, c, f):
    n = a.shape[0]
    y = np.zeros(n)
    B = np.zeros(n)
    x = np.zeros(n)
    y[0] = f[0] / b[0]
    d = b[0]
    for i in range(1, n):
        B[i - 1] = c[i - 1] / d
        d = b[i] - a[i] * B[i - 1]
        y[i] = (f[i] - a[i] * y[i - 1]) / d
    x[n-1] = y[n-1]

    for i in range(n-2, -1, -1):
        x[i] = y[i] - B[i] * x[i + 1]
    return x

a = np.array([0, -1, -1, -1, -1])  # 下对角线元素
b = np.array([2, 2, 2, 2, 2])  # 主对角线元素
c = np.array([-1, -1, -1, -1, 0])  # 上对角线元素
f = np.array([1, 0, 0, 0, 0])  # 右侧常数项
x = tridiagonal_solver(a, b, c, f)
print("线性方程组的解x:")
print(x)
```
在 6ms 的 2023.10.05 13:21:00 执行

Figure 3: Source Code

```
线性方程组的解x:
[0.83333333 0.66666667 0.5        0.33333333 0.16666667]
```

Figure 4: Result