数值代数实验报告 4

马天开

2023 年 12 月 1 日

1 问题描述

1.1 考虑两点边值问题

$$\begin{cases}\epsilon\dfrac{d^2y}{dx^2}+\dfrac{dy}{dx}=a, 0<a<1,\\ y(0)=0, y(1)=1\end{cases}$$

容易知道它的精确解为

$$y=\frac{1-a}{1-e^{-\frac{1}{\epsilon}}}1-e^{-\frac{x}{\epsilon}}+ax$$

为了把微分方程离散化，把 [0, 1] 区间 n 等分，令 $h=\frac{1}{n}, x_i=ih, i=1,\cdots,n-1$, 得到差分方程

$$\epsilon\frac{y_{i-1}-2y_i+y_{i+1}}{h^2}+\frac{y_{i+1}-y_i}{h}=a$$

简化为

$$(\epsilon+h)y_{i+1}-(2\epsilon+h)y_i+\epsilon y_{i-1}=ah^2$$

离散化后得到线性方程组 Ay = b, 其中

$$A=\begin{bmatrix}-(2\epsilon+h) & (\epsilon+h) & 0 & 0 & \cdots & 0\\ \epsilon & -(2\epsilon+h) & (\epsilon+h) & 0 & \cdots & 0\\ 0 & \epsilon & -(2\epsilon+h) & (\epsilon+h) & \cdots & 0\\ \vdots & & \ddots & \ddots & \ddots & \\ 0 & \cdots & 0 & \epsilon & -(2\epsilon+h) & (\epsilon+h)\\ 0 & 0 & \cdots & 0 & \epsilon & -(2\epsilon+h)\end{bmatrix}$$

注意 A 为 n-1 阶矩阵, 将线性方程组与差分方程进行对比得出正确的 b 向量（尤其注意第一行和最后一行）。

对 $\epsilon=1, a=1/2, n=100$, 分别用 Jacobi 迭代法,G-S 迭代法和 SOR 迭代法求线性方程组的解，要求 4 位有效数字，然后比较迭代次数，运行时间与精确解的误差。迭代法终止条件为 $||x^{(k+1)}-x^{(k)}||<10^{-6}$。

对 $\epsilon=0.1, 0.01, 0.0001$, 考虑同样的问题。要求输出计算结果，收敛所需要的迭代次数和运行时间。

1.2 考虑偏微分方程

$$-\Delta u + g(x,y)u = f(x,y) \quad (x,y) \in [0,1]\times[0,1]$$

在 [0,1]×[0,1] 边界上 u = 1. 沿 x 方向和 y 方向均匀剖分 N 等份，令 h = 1/N, 并设应用中心差分离散化后得到差分方程的代数方程组为

$$-u_{i-1,j} - u_{i,j-1} + (4 + h^2 g(ih,jh))u_{i,j} - u_{i+1,j} - u_{i,j+1} = h^2 f(ih,jh)$$

取 $g(x,y) = \exp(xy)$, $f(x,y) = x + y$, 分别用 Jacobi 迭代法,G-S 迭代法和 SOR 迭代法求解上述代数方程组，要求输出解的最小分量，并比较 N = 20, 40, 60 时收敛所需要的迭代次数和运行时间，迭代终止条件为 $||u^{(k+1)} - u^{(k)}||_2 < 10^{-7}$。

要求仿照下面写的 Jacobi 迭代格式的推导过程推导处 G-S 迭代和 SOR 迭代的格式，在用 SOR 迭代法求解的过程中，请对不同的 N 使用合适的松弛因子 ω, 并在程序输出中打印松弛因子的值。观察运行结果后选取合适的。(代码中不需要体现选取过程，只需给出即可)。

注意本题中的三个迭代法的算法需要重新写，不能用矩阵的通用算法!!!

2 算法说明

1、Jacobi 迭代法

考虑其迭代格式 $Dx_{(k+1)} = (L + U)x_k + b$。对第 i 行有

$$D_i i x_i^{(k+1)} = \sum_{i,j} (L+U)_{ij} x_j^{(k)} + b_i$$

将 D, L, U 还原成代数方程组的 Jacobi 迭代式

$$D_{ii} x_i^{(k+1)} = L_{i1} x_1^{(k)} + \cdots + L_{ii-1} x_{i-1}^{(k)} + U_{ii+1} x_{i+1}^{(k)} + \cdots U_{in} x_n^{(k)} b_i$$

即只有与向量 b 的下标相同的位置替换成 x^{k+1}。

由此类比推广至矩阵（或者可以直接将矩阵拉直成向量），知代数方程组 (1) 的 Jacobi 迭代格式为

$$(4 + h^2 g(ih,jh))u_{i,j}^{(k+1)} = u_{i-1,j}^{(k)} + u_{i,j-1}^{(k)} + u_{i+1,j}^{(k)} + u_{i,j+1}^{(k)} + h^2 f(ih,jh)$$

2、G-S 迭代法

$$(4 + h^2 g(ih,jh))u_{i,j}^{(k+1)} = u_{i-1,j}^{(k+1)} + u_{i,j-1}^{(k+1)} + u_{i+1,j}^{(k)} + u_{i,j+1}^{(k)} + h^2 f(ih,jh)$$

3、SOR 迭代法

$$u_{i,j}^{(k+1)} = \frac{\omega}{4 + h^2 g(ih,jh)} (u_{i-1,j}^{(k+1)} + u_{i,j-1}^{(k+1)} + u_{i+1,j}^{(k)} + u_{i,j+1}^{(k)} + h^2 f(ih,jh)) + (1-\omega) u_{i,j}^{(k)}$$

3 程序介绍

并没有什么特别值得介绍的，也许。这次抢了点 ddl 有的地方可能乱一点，见谅。

4 运行结果

```
------ Q 4.1 ------
eps = 1
[0,0.0128705,0.0256626,0.0383772,0.0510151,0.063577,0.0760636,0.0884757,0.100814,0.113079,0.12527

Jacobi Iteration:
11118 iterations
93637  s
diff = 0.0970001
[0.0128434,0.0256091,0.038298,0.0509107,0.0634482,0.0759111,0.0883001,0.100616,0.11286,0.125032,0

Gauss-Seidel Iteration:
6234 iterations
71759  s
diff = 0.0978536
[0.0128486,0.0256196,0.0383136,0.0509315,0.063474,0.0759418,0.0883357,0.100656,0.112905,0.125081,

SOR Iteration (omega = 1.6):
4187 iterations
59507  s
diff = 0.0981855
[0.0128508,0.0256239,0.03832,0.0509399,0.0634844,0.0759542,0.0883501,0.100673,0.112923,0.125102,0

eps = 0.1
[0,0.0525835,0.100639,0.144597,0.184847,0.221744,0.255604,0.286719,0.315348,0.341729,0.366075,0.3

Jacobi Iteration:
5465 iterations
46316  s
diff = 0.112392
[0.0504433,0.0967554,0.139312,0.178454,0.214493,0.247711,0.278363,0.306683,0.332884,0.357158,0.37

Gauss-Seidel Iteration:
2969 iterations
34354  s
diff = 0.112556
[0.0504503,0.0967688,0.139331,0.178479,0.214522,0.247744,0.2784,0.306723,0.332927,0.357203,0.3797

SOR Iteration (omega = 1.5):
2061 iterations
```

```
29169  s
diff = 0.112612
[0.0504528,0.0967736,0.139338,0.178488,0.214533,0.247755,0.278413,0.306738,0.332942,0.357219,0.37

eps = 0.01
[0,0.32106,0.442332,0.490106,0.510842,0.521631,0.528761,0.534544,0.539832,0.544938,0.549977,0.554

Jacobi Iteration:
508 iterations
5042  s
diff = 0.267383
[0.254995,0.384993,0.452492,0.488742,0.509368,0.522181,0.531088,0.538042,0.54402,0.549508,0.55475

Gauss-Seidel Iteration:
307 iterations
3574  s
diff = 0.267387
[0.254997,0.384996,0.452496,0.488746,0.509372,0.522185,0.531092,0.538045,0.544022,0.549511,0.5547

SOR Iteration (omega = 1.4):
217 iterations
2541  s
diff = 0.267388
[0.254998,0.384997,0.452497,0.488747,0.509373,0.522186,0.531092,0.538046,0.544022,0.549511,0.5547

eps = 0.0001
[0,0.505,0.51,0.515,0.52,0.525,0.53,0.535,0.54,0.545,0.55,0.555,0.56,0.565,0.57,0.575,0.58,0.585,

Jacobi Iteration:
113 iterations
1830  s
diff = 0.502493
[0.500049,0.509951,0.514999,0.52,0.525,0.53,0.535,0.54,0.545,0.55,0.555,0.56,0.565,0.57,0.575,0.5

Gauss-Seidel Iteration:
106 iterations
1224  s
diff = 0.502493
[0.500049,0.509951,0.515,0.52,0.525,0.53,0.535,0.54,0.545,0.55,0.555,0.56,0.565,0.57,0.575,0.58,0

SOR Iteration (omega = 1):
```

```
106 iterations
1250  s
diff = 0.502493
[0.500049,0.509951,0.515,0.52,0.525,0.53,0.535,0.54,0.545,0.55,0.555,0.56,0.565,0.57,0.575,0.58,0
------ Q 4.2 ------

N = 20

Jacobi Iteration:
1207 iterations
3880  s
0.974077

Gauss-Seidel Iteration:
633 iterations
3351  s
0.974077

SOR Iteration (omega=1.8):
88 iterations
524  s
0.974077

N = 40

Jacobi Iteration:
4410 iterations
60575  s
0.976126

Gauss-Seidel Iteration:
2316 iterations
55730  s
0.976127

SOR Iteration (omega=1.9):
186 iterations
4880  s
0.976127

N = 60
```

```
Jacobi Iteration:
9494 iterations
262795  s
0.976783

Gauss-Seidel Iteration:
4993 iterations
222763  s
0.976784

SOR Iteration (omega=1.9):
205 iterations
11014  s
0.976785
```

5 结果分析

第一问画了个图，虽然没什么用，但是还是放上来了。

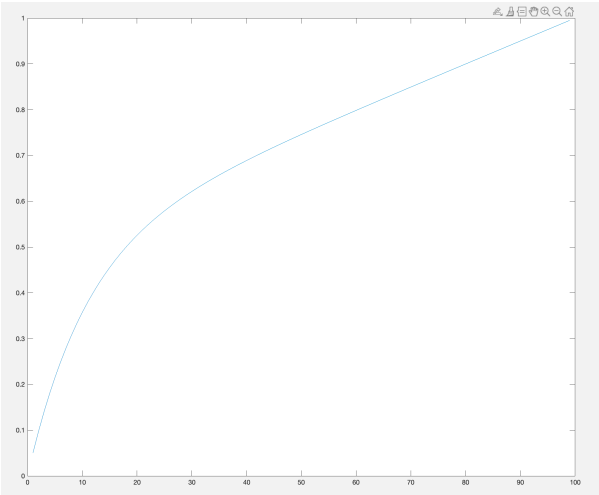

图 1: 计算数据

对比解析解绘制的图像：

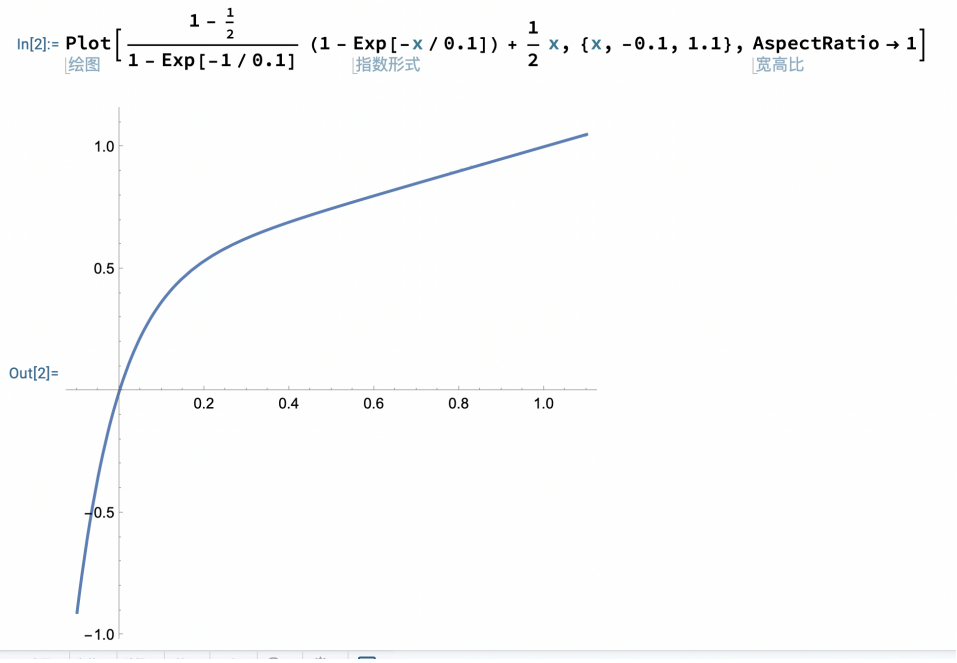

图 2: 解析解