

# 方程组求解

St Maxwell

2018 年 10 月 1 日

## 1 Gauss 消元法

$$\begin{cases} 2x - 2y - z = -2 \\ 4x + y - 2z = 1 \\ -2x + y - z = -3 \end{cases}$$

对于以上的线性方程组，可以将其写成（增广）矩阵的形式：

$$\left( \begin{array}{ccc|c} 2 & -2 & -1 & -2 \\ 4 & 1 & -2 & 1 \\ -2 & 1 & -1 & -3 \end{array} \right)$$

使用 Gauss 消元法，首先通过矩阵初等变换进行消去，将左边变成上三角矩阵。

$$\left( \begin{array}{ccc|c} 2 & -2 & -1 & -2 \\ 0 & 5 & 0 & 5 \\ 0 & 0 & -2 & -4 \end{array} \right)$$

之后进行回代，并将主元约化为 1。

$$\left( \begin{array}{ccc|c} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 2 \end{array} \right)$$

由此得到方程组的解。

$$\begin{cases} x = 1 \\ y = 1 \\ z = 2 \end{cases}$$

## 2 fortran 代码细节与 LAPACK 包

代码中的增广矩阵是使用 `reshape` 函数生成的。

```
real(kind=8) :: mat(4,3) = reshape( (/ 2,-2, -1,-2, &
                                     4, 1, -2, 1, &
                                     -2, 1, -1,-3 /), (/ 4,3 /) )
```

所以矩阵元（数组元素）的下标和通常是相反的。例如，上面的增广矩阵中  $a_{21} = 4$ ；而在数组中，它对应于 `mat(1,2)`。

要说明这一点的主要原因是，这个数组的定义方式与 LAPACK 包的 `dgesv` 子程序对输入数组的要求是相反的。

```
PROGRAM Main
  IMPLICIT NONE
  REAL    :: A(3,3), b(3)
  INTEGER :: i, j, pivot(3), ok

  ** Initialize A(:, :)
  ** Initialize b(:)

  CALL SGESV(3, 1, A, 3, pivot, b, 3, ok)

  PRINT *, b    !! Print solution
END PROGRAM
```

以上代码中的  $A_{(i,j)}$  即对应矩阵元  $A_{ij}$ 。此时若依然使用 `reshape` 函数，其矩阵元应按列依次定义。

正式的项目中应该还是考虑使用 LAPACK 包，所以以后定义矩阵就选择第二种方式，而且也符合数学上的习惯。