科学計算研究室 Python ゼミ

～ 3. Jordan 法～

2021-02-24 福田　浩

1　原理

1.1　計算の内容

なんてことはない．中学校の数学で習った「消去法」．

$$\begin{cases} 2x & +y & +3z & = 13 \\ x & +3y & +2z & = 13 \\ 3x & +2y & +z & = 10 \end{cases} \tag{1}$$

第 1 式の x の係数で第 1 式を割る (規格化する)．

$$\begin{cases} x & +0.5y & +1.5z & = 6.5 \\ x & +3y & +2z & = 13 \\ 3x & +2y & +z & = 10 \end{cases} \tag{2}$$

第 2 式-(第 2 式の x の係数) ×第 1 式,
第 3 式-(第 3 式の x の係数) ×第 1 式,
で x を消去する．

$$\begin{cases} x & +0.5y & +1.5z & = 6.5 \\ & +2.5y & +0.5z & = 6.5 \\ & +0.5y & -3.5z & = -9.5 \end{cases} \tag{3}$$

第 2 式の y の係数で第 2 式を割る (規格化する)．

$$\begin{cases} x & +0.5y & +1.5z & = 6.5 \\ & y & +0.2z & = 2.6 \\ & +0.5y & -3.5z & = -9.5 \end{cases} \tag{4}$$

第 1 式-(第 1 式の y の係数) ×第 2 式,
第 3 式-(第 3 式の y の係数) ×第 2 式,
で y を消去する．

$$\begin{cases} x & & +1.4z & = 5.2 \\ & y & +0.2z & = 2.6 \\ & & -3.6z & = -10.8 \end{cases} \tag{5}$$

第 3 式の z の係数で第 3 式を割る (規格化する)．

$$\begin{cases} x & & +1.4z & = 5.2 \\ & y & +0.2z & = 2.6 \\ & & z & = 3 \end{cases} \tag{6}$$

第 1 式-(第 1 式の z の係数) ×第 3 式,
第 2 式-(第 2 式の z の係数) ×第 3 式,
で z を消去する．

$$\begin{cases} x & & & = 1 \\ & y & & = 2 \\ & & z & = 3 \end{cases} \tag{7}$$

1.2 アルゴリズム

行列表記したときの対角要素 `a[i][i]`[1]に着目する.

1. 第 1 ピボットで第 1 式を規格化する
2. 第 i 式の第 j 要素 `a[i][j]` $-=$ 第 i 式の第 1 要素 $\times$ 第 1 式の第 j 要素
3. 第 2 ピボットで第 2 式を規格化する
4. 第 i 式の第 j 要素 `a[i][j]` $-=$ 第 i 式の第 2 要素 $\times$ 第 2 式の第 j 要素
   $\vdots$
5. 第 n ピボットで第 n 式を規格化する
6. 第 i 式の第 j 要素 `a[i][j]` $-=$ 第 i 式の第 n 要素 $\times$ 第 n 式の第 j 要素

2 課題

Jordan 法を用いて連立方程式の解を求めるプログラムを Python で書き，動作を検証せよ.

- 連立方程式の元数と行列表示にしたときの係数を，標準入力から入力する機能を備えること
- 連立方程式の解を，標準出力に出力する機能を備えること
- 以下の連立方程式を解け

$$\begin{cases} 3x & +y & +z & = 10 \\ x & +5y & +2z & = 21 \\ x & +2y & +5z & = 30 \end{cases} \tag{8}$$

- 以下の連立方程式を解け

$$\begin{cases} 0.51x_1 & +0.95x_2 & +0.80x_3 & +0.28x_4 & +0.41x_5 & = 16.7 \\ 0.39x_1 & +0.25x_2 & +0.43x_3 & +0.28x_4 & +0.88x_5 & = 9.8 \\ 0.55x_1 & +0.91x_2 & +0.12x_3 & +0.23x_4 & +0.31x_5 & = 10.4 \\ 0.26x_1 & +0.66x_2 & +0.95x_3 & +0.52x_4 & +0.57x_5 & = 17.7 \\ 0.83x_1 & +0.73x_2 & +0.62x_3 & +0.16x_4 & +0.77x_5 & = 14.1 \end{cases} \tag{9}$$

[1] ピボット (pivot) と呼ぶ

2.1　参考： C++のソースコード例

連立方程式の元数と行列表記にしたときの各要素を標準入力から読み取り，その解を Jordan 法により求めて，標準出力に出力する．アルゴリズムの参考例であり，エラー処理は実装していない．

```
#include <iostream>
#include <math.h>
#define EPS 0.0001
using namespace std;

int main(void){
    double a[10][111];
    double pivot, del;
    int n, i, j, k;

    cin >> n;
    for (int i=0;i<n;i++){
        for (int j=0;j<n+1;j++) cin >> a[i][j];
    }
    for (i=0; i<n;i++){
        pivot = a[i][i];
        if (fabs(pivot) > EPS){
            for (j=i; j<n+1; j++) a[i][j] /= pivot;
            for (k=0; k<n; k++){
                del = a[k][i];
                for (j=i; j<n+1; j++) if(k!=i) a[k][j] -= del*a[i][j];
            }
        }
        else{
            cout << "Pivot: " << pivot << "is too small." << endl;
            return 1;
        }
    }
    for (i=0;i<n;i++) cout << a[i][n] << endl;
    return 0;
}
```

2.2　更なる検討

C++ソースコードにあるように，pivot が極端に小さくなると，解が不安定になるため注意が必要である．このような状態を一般に ill-condition と呼ぶ (ill-condition そのものはもっと広い概念である)．例えば以下の 2 つの連立方程式を解き，その解の違いを確認せよ．

$$\begin{cases} 99x & +98y = 197 \\ 100x & +99y = 199 \end{cases} \tag{10}$$

$$\begin{cases} 98.99x & +98y = 197 \\ 100x & +99y = 199 \end{cases} \tag{11}$$