数値計算法・数値解析

連立一次方程式：ガウス-ジョルダン法

宮崎大学 工学部

第 6 回

連立一次方程式：ガウス-ジョルダン法

連立方程式をシステマティックに解く

- 方程式の両辺を $c(\neq 0)$ 倍しても同値
- 二つの方程式の両辺同士を足しても同値

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

$$\left(\begin{array}{ccc|c} 2 & 1 & 3 & 13 \\ 1 & 3 & 2 & 13 \\ 3 & 2 & 1 & 10 \end{array}\right)$$

$$\sim \left(\begin{array}{ccc|c} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 2 \\ 0 & 0 & 1 & 3 \end{array}\right)$$

連立一次方程式：ガウス-ジョルダン法

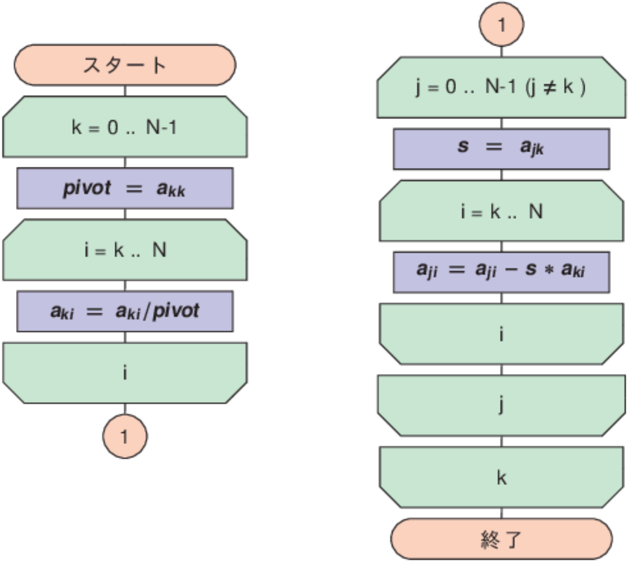
- $N \times (N+1)$ の係数行列 $\{a_{ij}\}$ が与えられる
- 対角要素を 1，それ以外を 0 にする (N+1 列目を除く)

  以下を繰り返す (k = 0 .. N-1)

  1. 対角要素 (k 行 k 列) を ***pivot*** とする：
     $pivot = a_{kk}$
  2. k 行目を ***pivot*** で割る：
     i = k .. N について，$a_{ki} = a_{ki}/pivot$
  3. j 行目 (j ≠ k) について，k 列目が 0 になるようにする：
     j = 0 .. N-1 (j ≠ k) について
     - 3.1 j 行 k 列の要素を ***s*** とする：
       $s = a_{jk}$
     - 3.2 k 行目を s 倍して j 行目から引く：
       $a_{ji} = a_{ji} - s * a_{ki}$ (i = k .. N)

連立一次方程式：ガウス-ジョルダン法

N × (N+1) の係数行列が与えられる



例題 1

以下の連立一次方程式をガウス-ジョルダン法で解け

$$
\begin{cases}
2x + & -4y + & 6z = & 1 \\
-1x + & 7y + & -8z = & 0 \\
1x + & 1y + & -2z = & 3
\end{cases}
$$

例題 2

以下の連立一次方程式をガウス-ジョルダン法で解け

$$\begin{cases} 2x + & 8y + & 2z + & -3w = & 2 \\ 4x + & 6y + & -2z + & -1w = & 1 \\ 2x + & -4y + & -2z + & -1w = & 3 \\ 1x + & -5y + & 2z + & 1w = & -2 \end{cases}$$

ガウス-ジョルダン法のプログラム(1/2)

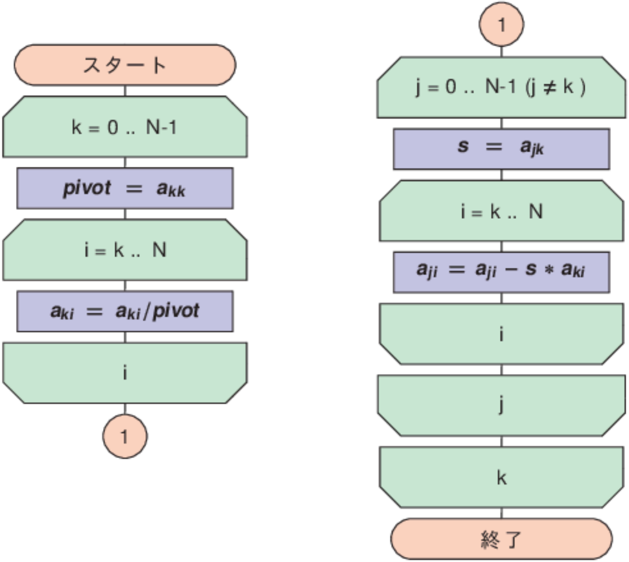
```
#define N  3
int main(int argc, char *argv[])
{
  double coeff[][N+1] = {{2, 1, 3, 13},
                         {1, 3, 2, 13},
                         {3, 2, 1, 10}};

  gauss_jordan(coeff, N);

  for(int i = 0; i < N; i++) {
    printf("x%d␣=␣%.15f\n", i, coeff[i][N]);
  }

  return 0;
}
```

ガウス-ジョルダン法のプログラム (2/2)

```
void gauss_jordan(double a[N][N+1], int size){
  for(int k = 0; k < size; k++) {
    double  pivot = a[k][k];
    for(int i = k; i < size+1; i++){
      a[k][i] /= pivot;
    }
    for(int j = 0; j < size; j++) {
      if (j == k) continue;
      double s = a[j][k];
      for(int i = k; i < size+1; i++) {
        a[j][i] = a[j][i] - s * a[k][i];
    } }
  }
  return;
}
```

連立一次方程式：ガウス-ジョルダン法

N × (N+1) の係数行列が与えられる



方程式の順番

$$\begin{cases} & y+ & z = & 3 \\ x+ & 3y+ & 2z = & 5 \\ 2x+ & & z = & 1 \end{cases}$$

⇒pivot が 0 になり解けない！

方程式の順番を入れ替える
（連立方程式の意味は変わらない）

$$\begin{cases} x+ & 3y+ & 2z = & 5 \\ & y+ & z = & 3 \\ 2x+ & & z = & 1 \end{cases}$$

⇒ 解ける！

変数の並び順

$$
\left\{
\begin{array}{rrrl}
 & y+ & z = & 3 \\
x+ & 3y+ & 2z = & 5 \\
2x+ & & z = & 1
\end{array}
\right.
$$

⇒pivot が 0 になり解けない！

変数の並び順を変えても連立方程式の意味は変わらない

$$
\left\{
\begin{array}{rrrl}
y+ & & z = & 3 \\
3y+ & x+ & 2z = & 5 \\
 & 2x+ & z = & 1
\end{array}
\right.
$$

⇒ 解ける！

ピボット選択

- ガウス-ジョルダン法では pivot での割り算がある
  - pivot が 0 だと計算できない
  - pivot の絶対値が小さいと計算が不安定になる
- 連立方程式の順番を入れ替えても，意味は変わらない

ピボット選択

pivot を選ぶ際に絶対値が最大のものを選ぶ

部分ピボット選択：計算途中で行を入れ替える

完全ピボット選択：計算途中で行と列両方入れ替える