情報処理論2a

5章「連立一次方程式」

（Gauss消去法）

中島康彦

§5. 1 今日の作業ディレクトリを作る

1. % cd ⇒ ホームディレクトリへ移動
2. % mkdir chap17 ⇒ ディレクトリchap17を作成
3. % cd chap17 ⇒ ディレクトリchap17へ移動
4. % netscapeを使ってdata17をchap17へダウンロード
5. % tar xvf data17 ⇒ サンプルデータの複写

```
equation1.c
equation2.c
equation1.in
equation2.in
equation3.in
```

§5. 2 連立一次方程式の求解

連立一次方程式

A00*x0 + A01*x1 + ... + A0p*xp = b0
A10*x0 + A11*x1 + ... + A1p*xp = b1
Ap0*x0 + Ap1*x1 + ... + App*xp = bp

行列による表現 ... Ax = b

$$
\begin{bmatrix} A00 & A01 & \ldots & A0p \\ A10 & A11 & \ldots & A1p \\ A20 & A21 & \ldots & A2p \\ \vdots & \vdots & & \vdots \\ Ap0 & Ap1 & \ldots & App \end{bmatrix}
\begin{bmatrix} x0 \\ x1 \\ x2 \\ \vdots \\ xp \end{bmatrix}
=
\begin{bmatrix} b0 \\ b1 \\ b2 \\ \vdots \\ bp \end{bmatrix}
$$

§5. 3 Gauss消去法(LU分解)

Aを三角行列の積に分解する

L (下三角行列) U (上三角行列)

$$
A = \begin{bmatrix} 1 & 0 & \ldots & 0 & 0 \\ m10 & 1 & \ldots & 0 & 0 \\ m20 & m21 & \ldots & 0 & 0 \\ \vdots & \vdots & & \vdots & \vdots \\ mp0 & mp1 & \ldots & mp,p\text{-}1 & 1 \end{bmatrix}
\begin{bmatrix} A00 & A01 & \ldots & A0p \\ 0 & A11' & \ldots & A1p' \\ 0 & 0 & \ldots & A2p'' \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \ldots & App'''' \end{bmatrix}
$$

A10=m10*A00　A11=m10*A01+　　A11'　…　A1p=m10*A0p+　　A1p'
A20=m20*A00　A21=m20*A01+m21*A11'　…　A2p=m20*A0p+m21*A1p'+A2p''
Ap0=mp0*A00　Ap1=mp0*A01+mp1*A11'　…　App=mp0*A0p+mp1*A1p'+...

§5. 4 LとU すなわち mijとAij'を求める

mijとAij'を求める

m10= A10／A00	A11′=A11	m10＊A01	A12′ =A12	m10＊A02	…	A1p′ =A1p	m10＊A0p
m20= A20／A00	【A21′=A21	m20＊A01】	【A22′ =A22	m20＊A02】	…	【A2p′ =A2p	m20＊A0p】
mp0= Ap0／A00	【Ap1′=Ap1	mp0＊A01】	【Ap2′ =Ap2	mp0＊A02】	…	【App′ =App	mp0＊A0p】
m21= 【A21′】／A11′			A22′′=A22′	m21＊A12′	…	A2p′′=A2p′	m21＊A1p′
mp1= 【Ap1′】／A11′			【Ap2′′=Ap2′	mp1＊A12′】	…	【App′′=App′	mp1＊A1p′】

▶ n元一次方程式の場合 p=n-1

```
for (i=0; i<n; i++) {
  for (j=i+1; j<n; j++) {
    m[j][i] = A[j][i]/A[i][i];
    for (k=i+1; k<n; k++)
      A[j][k] -= m[j][i]*A[i][k];
  }
}
```

§5. 5 Lb’ ＝ bを解く

Ax ＝ LUx ＝ b

まずLb’ ＝ bを解く

L（下三角行列）

$$
\begin{bmatrix} 1 & 0 & \ldots & 0 & 0 \\ m10 & 1 & \ldots & 0 & 0 \\ m20 & m21 & \ldots & 0 & 0 \\ : & : & & : & : \\ mp0 & mp1 & \ldots & mp,p-1 & 1 \end{bmatrix}
\begin{bmatrix} b0' \\ b1' \\ b2' \\ : \\ bp' \end{bmatrix}
=
\begin{bmatrix} b0 \\ b1 \\ b2 \\ : \\ bp \end{bmatrix}
$$

```
for (i=0; i<n; i++) {
  for (j=0; j<i; j++)
    b[i] -= m[i][j]*b[j];
}
```

§5. 6 Ux ＝ b'を解く

次にUx ＝ b'を解く

U（上三角行列）

$$
\begin{bmatrix}
A00 & A01 & \ldots & A0p \\
0 & A11' & \ldots & A1p' \\
0 & 0 & \ldots & A2p' \\
\vdots & \vdots & & \vdots \\
0 & 0 & \ldots & App''
\end{bmatrix}
\begin{bmatrix} x0 \\ x1 \\ x2 \\ \vdots \\ xp \end{bmatrix}
=
\begin{bmatrix} b0' \\ b1' \\ b2' \\ \vdots \\ bp' \end{bmatrix}
$$

```
for (i=n-1; i>=0; i--) {
  for (j=i+1; j<n; j++)
    b[i] -= A[i][j]*x[j];
  x[i] = b[i]/A[i][i];
}
```

§5. 7 除算を減らす

$$
\begin{bmatrix}
1 & 0 & \ldots & 0 & 0 \\
m10 & 1 & \ldots & 0 & 0 \\
m20 & m21 & \ldots & 0 & 0 \\
\vdots & \vdots & & \vdots & \vdots \\
mp0 & mp1 & \ldots & mp,p\!-\!1 & 1
\end{bmatrix}
\begin{bmatrix}
A00 & A01 & \ldots & A0p \\
0 & A11' & \ldots & A1p' \\
0 & 0 & \ldots & A2p'' \\
\vdots & \vdots & & \vdots \\
0 & 0 & \ldots & App''''
\end{bmatrix}
$$

```
for (i=0; i<n; i++) {
  A[i][i] = 1.0/A[i][i];
  for (j=i+1; j<n; j++) {
    m[j][i] = A[j][i]*A[i][i];
    for (k=i+1; k<n; k++)
      A[j][k] -= m[j][i]*A[i][k];
  }
}
for (i=0; i<n; i++) {
  for (j=0; j<i; j++)
    b[i] -= m[i][j]*b[j];
}
for (i=n-1; i>=0; i--) {
  for (j=i+1; j<n; j++)
    b[i] -= A[i][j]*x[j];
  x[i] = b[i]*A[i][i];
}
```

§5. 8 実は配列mは不要

$$
\begin{bmatrix}
A_{00} & A_{01} & \dots & A_{0,p-1} & A_{0p} \\
m_{10} & A_{11}' & & A_{1,p-1}' & A_{1p}' \\
m_{20} & m_{21} & & A_{2,p-1}'' & A_{2p}'' \\
\vdots & \vdots & & & \vdots \\
m_{p0} & m_{p1} & \dots & m_{p,p-1} & A_{pp}'''
\end{bmatrix}
$$

```
for (i=0; i<n; i++) {
  A[i][i] = 1.0/A[i][i];
  for (j=i+1; j<n; j++) {
    A[j][i] *= A[i][i];
    for (k=i+1; k<n; k++)
      A[j][k] -= A[j][i]*A[i][k];
  }
}
for (i=0; i<n; i++) {
  for (j=0; j<i; j++)
    b[i] -= A[i][j]*b[j];
}
for (i=n-1; i>=0; i--) {
  for (j=i+1; j<n; j++)
    b[i] -= A[i][j]*x[j];
  x[i] = b[i]*A[i][i];
}
```

§5. 9 解が求まらない場合がある

正しい解が求まる場合

$$
\begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & 1 & 1 \\ 2 & 2 & 1 & 1 \\ 2 & 2 & 2 & 1 \end{bmatrix}
\begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix}
=
\begin{bmatrix} 10 \\ 11 \\ 13 \\ 16 \end{bmatrix}
\Rightarrow
\begin{bmatrix} 1 & 0 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ 2 & 0 & 1 & 0 \\ 2 & 0 & 0 & 1 \end{bmatrix}
\begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & -1 & -1 & -1 \\ 0 & 0 & -1 & -1 \\ 0 & 0 & 0 & -1 \end{bmatrix}
$$

同じ解でも求まらない場合

$$
\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 2 \\ 1 & 1 & 2 & 2 \\ 1 & 2 & 2 & 2 \end{bmatrix}
\begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix}
=
\begin{bmatrix} 10 \\ 14 \\ 17 \\ 19 \end{bmatrix}
\Rightarrow
\begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & \infty & 1 & 1 \end{bmatrix}
\begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 1/\infty & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & -\infty \end{bmatrix}
$$

行を入れ換えると求まる場合

$$
\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 2 & 2 & 2 \\ 1 & 1 & 2 & 2 \\ 1 & 1 & 1 & 2 \end{bmatrix}
\begin{bmatrix} x_0 \\ x_3 \\ x_2 \\ x_1 \end{bmatrix}
=
\begin{bmatrix} 10 \\ 19 \\ 17 \\ 14 \end{bmatrix}
\Rightarrow
\begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix}
\begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

§5. 10 行の入れ換え(ピボッティング)

直接の原因は0除算にある

- **A[i][i]=1.0/A[i][i];**
- 絶対値最大の**A[k][k]**を探し**i**行と**k**行を交換
- 間接添字**p[i]**の初期値を**i**とし**p[i]**と**p[k]**を交換

```
pmax = 0.0;
k = -1;
for (j=i; j<n; j++) {
  if (pmax < abs(A[p[j]][i])) {
    pmax = abs(A[p[j]][i]);
    k = j;
  }
}
j = p[k]; p[k] = p[i]; p[i] = j;
A[p[i]][i] = 1.0/A[p[i]][i];
```

§5. 11 解が求まらない場合の検出

kが見つからない(絶対値最大の**A[k][k]**が0である)場合，本質的に解けない．

同様に，**A[k][k]**が0に近い場合，誤差が大きくなる．

```
pmax = 0.0;
k = -1;
for (j=i; j<n; j++) {
  if (pmax < abs(A[p[j]][i])) {
    pmax = abs(A[p[j]][i]);
    k = j;
  }
}
if (k == -1) {
  fprintf(stderr, "can't solve\n");
  exit(1);
}
j = p[k]; p[k] = p[i]; p[i] = j;
A[p[i]][i] = 1.0/A[p[i]][i];
```

§5. 12 プログラムおよび入力例

equation1.c … 単純なプログラム
equation2.c … ピボッティングおよび求解不能検出

入力例

	equation1.in	equation2.in	equation3.in
	4 1 1 1 1 10 2 1 1 1 11 2 2 1 1 13 2 2 2 1 16	4 1 1 1 1 10 1 1 1 2 14 1 1 2 2 17 1 2 2 2 19	4 1 1 1 1 10 1 1 1 1 20 1 1 2 2 17 1 2 2 2 19
equation1.c	解ける	解けない(異常終了)	解けない(異常終了)
equation2.c	解ける	解ける	解けない(検出)

§5. 13 コンパイルと実行

1．コンパイルする.

```
% gcc equation1.c -o equation1
```

2．実行

```
% ./equation1 < equation1.in
x0      = 1.0000000000e+00  ... 解x0
x1      = 2.0000000000e+00  ... 解x1
x2      = 3.0000000000e+00  ... 解x2
x3      = 4.0000000000e+00  ... 解x3
res0    = 0.0000000000e+00  ... 残差
res1    = 0.0000000000e+00  ... 残差
res2    = 0.0000000000e+00  ... 残差
res3    = 0.0000000000e+00  ... 残差

% ./equation1 < equation2.in
Floating exception (0除算例外)

% ./equation1 < equation3.in
Floating exception (0除算例外)
```

§5. 13 コンパイルと実行(続き)

1．コンパイルする.

```
% gcc equation2.c -o equation2
```

2．実行

```
% ./equation2 < equation1.in
x0     = 1.0000000000e+00  ... 解x0
x1     = 2.0000000000e+00  ... 解x1
x2     = 3.0000000000e+00  ... 解x2
x3     = 4.0000000000e+00  ... 解x3
res0   = 0.0000000000e+00  ... 残差
res1   = 0.0000000000e+00  ... 残差
res2   = 0.0000000000e+00  ... 残差
res3   = 0.0000000000e+00  ... 残差
% ./equation2 < equation2.in
x0     = 1.0000000000e+00  ... 解x0
x1     = 2.0000000000e+00  ... 解x1
x2     = 3.0000000000e+00  ... 解x2
x3     = 4.0000000000e+00  ... 解x3
res0   = 0.0000000000e+00  ... 残差
res1   = 0.0000000000e+00  ... 残差
res2   = 0.0000000000e+00  ... 残差
res3   = 0.0000000000e+00  ... 残差
% ./equation2 < equation3.in
can't solve
```

§5. 14 例題

A～J席の10種類，AとB，BとC，…，IとJは1000席差(A<B<…<J)，全席の合計は50000席とする．連立一次方程式を表す配列Aを記述せよ．

プログラムを用いてA～J席の各席数を求めよ．

§5. 15 今日の課題

A～T席の20種類
AとB，BとC，…，SとTの比は1：2(A＜B＜…＜T)
全席の合計は50000席とする.

- 連立一次方程式を表す配列Aを記述せよ.

- プログラムを用いてA～T席の各席数を求めよ.

宛先: nakashim@econ.kyoto-u.ac.jp
件名: unix2-学生番号

今日はここまで