岡本安晴　2001.2.1;2001.8

行列の計算―２

固有値・固有ベクトル・特異値・特異値分解・一般逆行列

スペクトル分解

　例えば、（２，２）型の行列

$$\begin{pmatrix} 5 & 3 \\ 3 & 5 \end{pmatrix} \tag{1}$$

に対して次式が成り立つ。

$$\begin{pmatrix} 5 & 3 \\ 3 & 5 \end{pmatrix}\begin{pmatrix} 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 8 \\ 8 \end{pmatrix} = 8 \times \begin{pmatrix} 1 \\ 1 \end{pmatrix}、 \tag{2}$$

$$\begin{pmatrix} 5 & 3 \\ 3 & 5 \end{pmatrix}\begin{pmatrix} 1 \\ -1 \end{pmatrix} = \begin{pmatrix} 2 \\ -2 \end{pmatrix} = 2 \times \begin{pmatrix} 1 \\ -1 \end{pmatrix} \tag{3}$$

　一般に、行列 A と数（スカラ） l および０でないベクトル$\mathbf{u}$の間に次式

$$A\mathbf{u} = l\mathbf{u} \tag{4}$$

が成り立つとき、数 l を行列 A の固有値、ベクトル$\mathbf{u}$を行列 A の固有値 l に対する固有ベクトルという(1)。

　（１）式の行列の場合、（２）式より数８が固有値、ベクトル$(1\ \ 1)^t$が固有値８に対する固有ベクトルであることがわかる。（３）式からは、２が固有値、$(1\ \ -1)^t$が固有値２に対する固有ベクトルであることがわかる。

　（４）式の関係は、固有ベクトルをa 倍したベクトル$a\mathbf{u}$に対しても成り立つ。

$$A(a\mathbf{u}) = l\,(a\mathbf{u})$$

　すなわち、$\mathbf{u}$が固有値 l に対する固有ベクトルであるとき、$\mathbf{u}$を任意の数a によってa 倍したベクトル$a\mathbf{u}$も固有値 l に対する固有ベクトルになっている。このことから、固有ベクトルとして長さ１のものをとることがある。（１）式の行列の場合、長さ１の固有ベクトルは、固有値８に対しては$\left(1/\sqrt{2}\ \ \ 1/\sqrt{2}\right)^t$、固有値２に対しては$\left(1/\sqrt{2}\ \ \ -1/\sqrt{2}\right)^t$とすることができる。これらの長さ１の固有ベクトルによって（１）式の行列を次のように表わすことができる。

$$\begin{pmatrix} 5 & 3 \\ 3 & 5 \end{pmatrix} = 8 \times \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix} + 2 \times \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{-1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix} \qquad \text{（4 a）}$$

したがって、（1）式の行列に(1,2)型の行列（列ベクトル）$\mathbf{x}$を右から掛けたものは次のように表せる。

$$\begin{pmatrix} 5 & 3 \\ 3 & 5 \end{pmatrix}\mathbf{x} = 8 \times \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix}\mathbf{x} + 2 \times \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{-1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix}\mathbf{x} \qquad \text{（5）}$$

（5）式の右辺において、ベクトル$\begin{pmatrix} 1/\sqrt{2} & 1/\sqrt{2} \end{pmatrix}$および$\begin{pmatrix} 1/\sqrt{2} & -1/\sqrt{2} \end{pmatrix}$の長さが１なので、積$\begin{pmatrix} 1/\sqrt{2} & 1/\sqrt{2} \end{pmatrix}\mathbf{x}$と$\begin{pmatrix} 1/\sqrt{2} & -1/\sqrt{2} \end{pmatrix}\mathbf{x}$は、ベクトル$\mathbf{x}$のベクトル$\begin{pmatrix} 1/\sqrt{2} & 1/\sqrt{2} \end{pmatrix}$および$\begin{pmatrix} 1/\sqrt{2} & -1/\sqrt{2} \end{pmatrix}$への正射影の長さを表わしている（図１）。

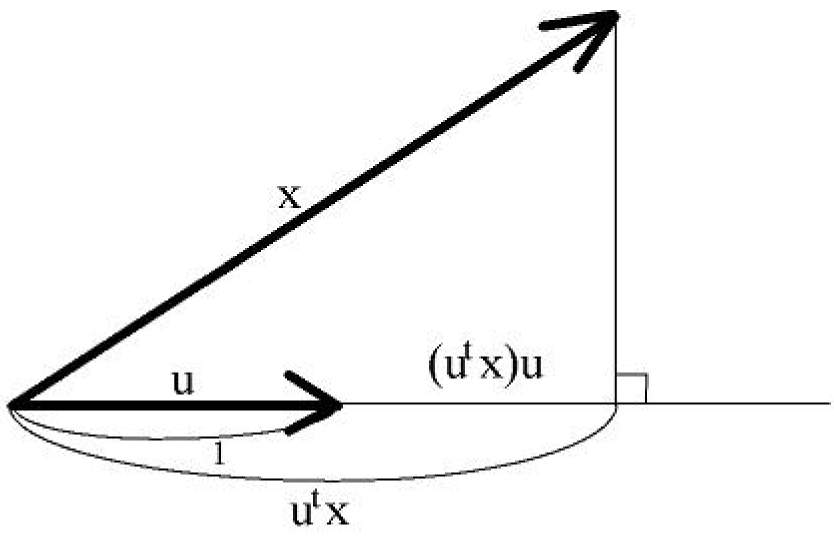


図１　長さ１のベクトル$\mathbf{u}$へのベクトル$\mathbf{x}$の正射影$\mathbf{u}\mathbf{u}^t\mathbf{x} = (\mathbf{u}^t\mathbf{x})\mathbf{u}$

したがって、

$$\begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{pmatrix} \mathbf{x} \quad \text{および} \quad \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{-1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \end{pmatrix} \mathbf{x}$$

は、ベクトル $\mathbf{x}$ のベクトル $\left(1/\sqrt{2} \quad 1/\sqrt{2}\right)$ および $\left(1/\sqrt{2} \quad -1/\sqrt{2}\right)$ への正射影を表わしている。つまり、（５）式は、ベクトル $\mathbf{x}$ に（１）式の行列を左から掛けたもの（式（５）の左辺）を、行列の２つの固有ベクトル $\left(1/\sqrt{2} \quad 1/\sqrt{2}\right)$ および $\left(1/\sqrt{2} \quad -1/\sqrt{2}\right)$ へのベクトル $\mathbf{x}$ の正射影にそれぞれの固有値８および２を掛けたものの和（式（５）の右辺）として表すものである。

一般に、(n,n)型の実対称行列（行列の成分が実数である対称行列）A は、n個の固有値 l_1、・・・、l_n をもつ。それらの固有値に対応する長さ１の固有ベクトルを $\mathbf{u}_1$、・・・、$\mathbf{u}_n$ とすると、

$$A = l_1 \mathbf{u}_1 \mathbf{u}_1^t + \cdots + l_n \mathbf{u}_n \mathbf{u}_n^t \qquad (6)$$

が成り立つ。（６）式を行列 A のスペクトル分解という。固有ベクトル $\mathbf{u}_1$、・・・、$\mathbf{u}_n$ はお互いに直交しているので、$\mathbf{u}_1$、・・・、$\mathbf{u}_n$ はn次元空間の正規直交基底になっている。

ベクトル $\mathbf{x}$ の左から行列 A を掛けると

$$A\mathbf{x} = l_1 \mathbf{u}_1 \mathbf{u}_1^t \mathbf{x} + \cdots + l_n \mathbf{u}_n \mathbf{u}_n^t \mathbf{x} \qquad (7)$$

となる。$\mathbf{u}_1^t \mathbf{x}$、・・・、$\mathbf{u}_n^t \mathbf{x}$ はベクトル $\mathbf{x}$ の $\mathbf{u}_1$、・・・、$\mathbf{u}_n$ への正射影の長さなので、$\mathbf{u}_1 \mathbf{u}_1^t \mathbf{x}$、・・・、$\mathbf{u}_n \mathbf{u}_n^t \mathbf{x}$ はベクトル $\mathbf{x}$ の $\mathbf{u}_1$、・・・、$\mathbf{u}_n$ への正射影を表わす。（７）式によって、ベクトル $\mathbf{x}$ の左から行列 A を掛けたものは、ベクトル $\mathbf{x}$ の $\mathbf{u}_1$、・・・、$\mathbf{u}_n$ への正射影をそれぞれの固有値 l_1、・・・、l_n 倍したものの和として表わされることがわかる。

（６）式を変形すると、

$$A = l_1 \mathbf{u}_1 \mathbf{u}_1^t + \cdots + l_n \mathbf{u}_n \mathbf{u}_n^t$$

$$= \left(\mathbf{u}_1 \cdots \mathbf{u}_n\right) \begin{pmatrix} l_1 & 0 & 0 & \cdots & 0 \\ 0 & l_2 & 0 & \cdots & 0 \\ & & \ddots & & \\ 0 & & \cdots & 0 & l_n \end{pmatrix} \begin{pmatrix} \mathbf{u}_1^t \\ \vdots \\ \mathbf{u}_n^t \end{pmatrix} \qquad (8)$$

となる。

いま、

$$P = (\mathbf{u}_1 \cdots \mathbf{u}_n)、\quad \Lambda = \begin{pmatrix} l_1 & 0 & 0 & \cdots & 0 \\ 0 & l_2 & 0 & \cdots & 0 \\ & & \ddots & & \\ 0 & & \cdots & 0 & l_n \end{pmatrix}$$

とおくと、（8）式は

$$A = P\Lambda P^t \qquad \text{（8 a）}$$

と書ける。ここで、$\mathbf{u}_1$、・・・、$\mathbf{u}_n$ は互いに直交する長さ１のベクトルなので、

$$P^t P = \begin{pmatrix} \mathbf{u}_1^t \\ \vdots \\ \mathbf{u}_n^t \end{pmatrix} (\mathbf{u}_1 \cdots \mathbf{u}_n)$$

$$= \begin{pmatrix} \mathbf{u}_1^t\mathbf{u}_1 & \mathbf{u}_1^t\mathbf{u}_2 & \mathbf{u}_1^t\mathbf{u}_3 & \cdots & \mathbf{u}_1^t\mathbf{u}_n \\ \mathbf{u}_2^t\mathbf{u}_1 & \mathbf{u}_2^t\mathbf{u}_2 & \mathbf{u}_2^t\mathbf{u}_3 & \cdots & \mathbf{u}_2^t\mathbf{u}_n \\ & & \ddots & & \\ \mathbf{u}_n^t\mathbf{u}_1 & & \cdots & \mathbf{u}_n^t\mathbf{u}_{n-1} & \mathbf{u}_n^t\mathbf{u}_n \end{pmatrix}$$

$$= \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ & & \ddots & & \\ 0 & & \cdots & 0 & 1 \end{pmatrix} = I$$

となる。すなわち、

$$P^{-1} = P^t \qquad \text{（9）}$$

が成り立つ。（9）式を満たす行列 P を直交行列という。

Power 法

ベクトル $\mathbf{x}$ に実対称行列 A を繰り返し掛けていくと、（7）式より固有値が最大である固

有ベクトルに近づくことがわかる。すなわち、

$$A^k\mathbf{x} = l_1^k\mathbf{u}_1^t\mathbf{u}_1\mathbf{x} + \cdots + l_n^k\mathbf{u}_n^t\mathbf{u}_n\mathbf{x}$$

なので、$l_1 > l_2 > \cdots > l_1$であるとき、$k \to \infty$とすると

$$\frac{\left\| l_i^k\mathbf{u}_i^t\mathbf{u}_i\mathbf{x} \right\|}{\left\| l_1^k\mathbf{u}_1^t\mathbf{u}_1\mathbf{x} \right\|} \to 0$$

となる。ここで、記号$\|\ \|$はベクトルの成分の２乗和の平方根（ノルム）を表わす。

Power 法では、上のことを用いて実対称行列Aの固有値と固有ベクトルを以下の手順で求める(2)。

（１）$i \leftarrow 1$とおく。
（２）ベクトル$\mathbf{x}_0$の初期値を適当に設定する。
（３）$\mathbf{y} \leftarrow A\mathbf{x}_0$、$l \leftarrow \|\mathbf{y}\|/\|\mathbf{x}_0\|$、$\mathbf{x}_1 \leftarrow \|\mathbf{y}\|^{-1}\mathbf{y}$とおく。
（４）$\mathbf{x}_1$が$\mathbf{x}_0$と同じとみなせる（固有値が正のとき）、あるいは$-\mathbf{x}_1$が$\mathbf{x}_0$と同じとみなせる（固有値が負のとき）とき、（５）に跳ぶ。
$\mathbf{x}_1$、$-\mathbf{x}_1$のいずれも$\mathbf{x}_0$と同じとはみなせないときは、$\mathbf{x}_0 \leftarrow \mathbf{x}_1$とおいて（３）に戻る。
（５）lをi番目の固有値l_i、$\mathbf{x}_1$をl_iに対応する固有ベクトル$\mathbf{u}_i$とする。
（６）Aから固有値l_iと固有ベクトル$\mathbf{u}_i$を除く。すなわち、$A \leftarrow A - l_i\mathbf{u}_i\mathbf{u}_i^t$とおく。
（７）Aが$\mathbf{0}$でないとき、$i \leftarrow i+1$とおいて（２）に戻る。Aが$\mathbf{0}$になったときはこの手続きを終了する。

上の手続きで固有値と固有ベクトルを求める手続き DecompPower をユニットファイル UMatCalc2.pas に宣言した。手続き DecompPower のヘッダーは次のようになっている。

```
procedure  DecompPower( var a         : TMatCalc2;
                        n, t_n_eigen : Longint;
                        var  n_eigen : Longint;
                        var lambda    : TVecCalc2;
                        var eigen_vctr : TMatCalc2 );
```

第１パラメータａに固有値を求める行列、第２パラメータｎに行列ａの大きさ（行列ａが(n,n)型であればｎ）を設定する。第３パラメータ t_n_eigen には求める固有値の数を設定する。実際に求められた固有値の数は第４パラメータ n_eigen に設定された変数に返される。行列の階数が t_n_eigen に設定された値以上であれば固有値は t_n_eigen に設定された数だけ求められるが、階数が t_n_eigen に設定された値より小さいときは階数の数の固有値が０でない固有値として算出される。第５パラメータ lambda と第６パラメータ eigen_vctr に設定された変数には、求まった固有値とそれに対応する固有ベクトルが返される。lambda[i]にｉ番目の固有値、eigen_vctr[j,i]にｉ番目の固有値に対応する固有ベクトルの第ｊ成分が返される。n_eigen の値より大きいｉに対する lambad[i]と eigen_vctr[j,i]の値は不定である。仮パラメータにおける型 TMatCalc2 と TVecCalc2 は、ユニットファイル UTypeDefMat2.pas に宣言されているもので、次のようになっている。

```
const
  NDimMat2 = 50;

type
  TMatCalc2 = array[1..NDimMat2,1..NDimMat2] of Extended;
  TVecCalc2 = array[1..NDimMat2] of Extended;
```

行列のサイズの最大値は NDimMat2 の値で与えられるので、大きな行列の固有値を計算するときは NDimMat2 の値を行列の大きさ以上の数にする。

手続き DecompPower を用いるときは、ユニット UMatCalc2 と UTypeDefMat2 の使用を uses 節に宣言しておく。

（１）式の行列の固有値と固有ベクトルは、手続き DecompPower を用いて次のように求めることができます。

```
a[1,1]:=5.0;  a[1,2]:=3.0;
a[2,1]:=3.0;  a[2,2]:=5.0;
DecompPower( a, 2, 2, n_eigen, lambda, eigen_vctr );
```

手続き DecompPower の実行後、n_eigen には２、lambda には２つの固有値８と２、eigen_vctr には２つの固有ベクトルが返される。

プロジェクト PCheckPower.dpr は上の計算を行うものである。このプロジェクトを実行して表示されるフォーム上の「GO」ボタンをクリックすると計算が始まる。計算の最初に、計算結果を書き出すテキストファイルの名前の設定を求めるダイアログボックスが表

示される。適当な名前を設定して「開く(O)」ボタンをクリックすると計算が始まり、計算結果が設定したファイル名のテキストファイルに書き出される。このファイルは、プロジェクトの実行終了後にエディタで開いて見ることができる（図２）。

図２　出力ファイルの表示

図２を見ると、固有値８に対応する固有ベクトルとして$\left(1/\sqrt{2} \quad 1/\sqrt{2}\right)^t$、固有値２に対応する固有ベクトルとして$-\left(1/\sqrt{2} \quad -1/\sqrt{2}\right)^t$が与えられている（$1/\sqrt{2} = 0.7071\cdots$）。固有値８に対応する固有ベクトルは式（４ａ）のものに一致しているが、固有値２に対応する方は符号が逆になっている。これは、固有値に対応する固有ベクトルを長さ１のものとして求めても、その向きは逆のものも許されるからである。すなわち、次式

$$A\mathbf{u} = l\mathbf{u}$$

の両辺に-1を掛けて

$$A(-\mathbf{u}) = l(-\mathbf{u})$$

を得るので、l と$\mathbf{u}$が固有値と固有ベクトルの組み合わせであるとき、l と$-\mathbf{u}$も固有値と固有ベクトルの組み合わせとなる。

ＱＬ法

Power 法は簡単であるが、計算時間が長くなるという欠点がある。Power 法より速い方法の１つにＱＬ法がある。ＱＬ法は３重対角行列という形の行列に適用されるものであるので、固有値を求める行列を同じ固有値をもつ３重対角行列に変換してからＱＬ法を適用する。まず、３重対角行列に変換する方法として、鏡像変換を用いるハウスホルダーの方法について説明する[(3)]。

ハウスホルダーの方法（３重対角行列化）

行列Aの固有値l とそれに対応する固有ベクトル$\mathbf{u}$は、（４）式、すなわち、

$$A\mathbf{u} = l\mathbf{u}$$

を満たす。これは、次式

$$A\mathbf{u} - l\mathbf{u} = (A - lI)\mathbf{u} = 0 \qquad （１０）$$

を $\mathbf{u}$ についての方程式とみたとき、0 以外の解 $\mathbf{u} \neq \mathbf{0}$ が存在することである。（１０）式が 0 以外の解をもつ条件は、$A - lI$ の行列式が 0 であること

$$\det(A - lI) = 0 \qquad （１１）$$

である。また、固有値 l は方程式（１１）の根になっている。

いま、正則行列 P に対して

$$B = P^{-1}AP \qquad （１２）$$

とおく。このとき、

$$\begin{aligned}\det(B - mI) &= \det(P^{-1}AP - mI) \\ &= \det(P^{-1}AP - mP^{-1}IP) \\ &= \det(P^{-1}(A - mI)P) \\ &= \det(P^{-1}) \cdot \det(A - mI) \cdot \det(P) \\ &= \det(A - mI) \cdot \det(P^{-1}) \cdot \det(P) \\ &= \det(A - mI)\end{aligned}$$

となる。したがって、（１１）式を満たす l は

$$\det(B - lI) = \det(P^{-1}AP - lI) = 0 \qquad （１３）$$

を満たす。

正則行列 P によって（１２）式の関係にある行列 A と B は相似であるといい、（１２）式の形による行列 A から B への変換を相似変換という。（１３）式は、相似変換によって固有値は変わらないことを示している。相似変換として鏡像変換という方法を用いて３重対角行列に変換する方法がハウスホルダーの方法である。

鏡像変換（elementary Hermitian transformation）[(3)] とは、原点を通る平面Mを鏡とする実像と虚像の関係のように、ベクトルとその変換後のベクトルがMに関して対称となる変換である。長さの等しい２つのベクトル $\mathbf{x}$ と $\mathbf{y}$ に対して、$\mathbf{x}$ を $\mathbf{y}$ に変換する鏡像変換は、行列

$$M = (I - 2\mathbf{u}\mathbf{u}^t)$$

によって

$$\mathbf{y} = M\mathbf{x} = (I - 2\mathbf{u}\mathbf{u}^t)\mathbf{x}$$

で与えられる（図３）。ここで、

$$\mathbf{u} = \frac{\mathbf{x} - \mathbf{y}}{\|\mathbf{x} - \mathbf{y}\|}$$

である。

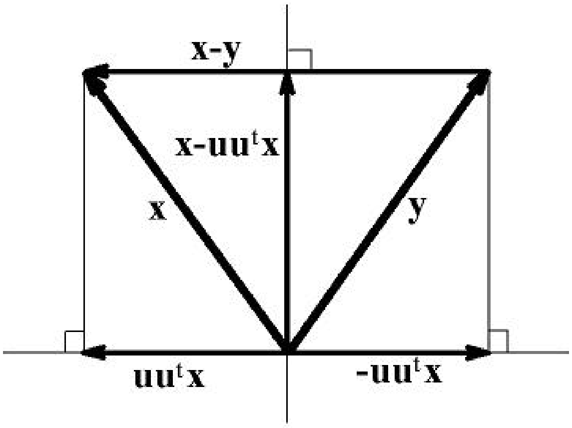


図３　$\mathbf{x}$ から $\mathbf{y}$ への鏡像変換 $\mathbf{y} = (\mathbf{x} - \mathbf{u}\mathbf{u}^t\mathbf{x}) - \mathbf{u}\mathbf{u}^t\mathbf{x} = (I - 2\mathbf{u}\mathbf{u}^t)\mathbf{x}$ 、$\mathbf{u} = \dfrac{\mathbf{x} - \mathbf{y}}{\|\mathbf{x} - \mathbf{y}\|}$

鏡像変換の行列 M は

$$M^t = (I - 2\mathbf{u}\mathbf{u}^t)^t = I^t - 2(\mathbf{u}\mathbf{u}^t)^t = I - 2\mathbf{u}\mathbf{u}^t = M$$

が成り立つので対称行列である。また、

$$MM = (I - 2\mathbf{u}\mathbf{u}^t)(I - 2\mathbf{u}\mathbf{u}^t) = I - 2\mathbf{u}\mathbf{u}^t - 2\mathbf{u}\mathbf{u}^t + 4\mathbf{u}\mathbf{u}^t\mathbf{u}\mathbf{u}^t = I$$

がなりたつので、

$$M^{-1} = M$$

となる。

鏡像変換を用いて対称行列を３重対角行列に変換する。３重対角行列とは、対角成分とその隣接成分以外は０であるものである。すなわち、次の形

$$\begin{pmatrix} * & * & 0 & & \cdots & & 0 \\ * & * & * & 0 & \cdots & & 0 \\ 0 & * & * & * & 0 & \cdots & 0 \\ & & & \ddots & & & \\ 0 & \cdots & 0 & * & * & * & 0 \\ 0 & & \cdots & 0 & * & * & * \\ 0 & & & \cdots & 0 & * & * \end{pmatrix}$$

の行列のことである。

(n,n)型の実対称行列Aを３重対角行列に変換するために、まず、第１行と第１列に注目する。Aを次のように分割する。

$$A = \left(\begin{array}{c|c} a_{11} & \mathbf{b}^t \\ \hline \mathbf{b} & \mathrm{C} \end{array}\right)$$

上のAの分割に対応して、次の(n,n)型の行列を考えPとおく。

$$P = \left(\begin{array}{c|c} 1 & \mathbf{0}^t \\ \hline \mathbf{0} & Q \end{array}\right)$$

このとき、

$$PAP = \left(\begin{array}{c|c} 1 & \mathbf{0}^t \\ \hline \mathbf{0} & Q \end{array}\right)\left(\begin{array}{c|c} a_{11} & \mathbf{b}^t \\ \hline \mathbf{b} & \mathrm{C} \end{array}\right)\left(\begin{array}{c|c} 1 & \mathbf{0}^t \\ \hline \mathbf{0} & Q \end{array}\right)$$

$$= \left(\begin{array}{c|c} 1 & \mathbf{0}^t \\ \hline \mathbf{0} & Q \end{array}\right)\left(\begin{array}{c|c} a_{11} & \mathbf{b}^t Q \\ \hline \mathbf{b} & CQ \end{array}\right)$$

$$= \left(\begin{array}{c|c} a_{11} & \mathbf{b}^t Q \\ \hline Q\mathbf{b} & QCQ \end{array}\right) \qquad （１４）$$

となる。

Qを鏡像変換として

$$\mathbf{b}^t Q = \begin{pmatrix} \|\mathbf{b}\| & 0 & \cdots & 0 \end{pmatrix}^t$$

となるように選ぶことができる。すなわち、

$$Q = I - 2\mathbf{u}\mathbf{u}'、\quad \mathbf{u} = \frac{\mathbf{b} - (\|\mathbf{b}\|\, 0 \cdots 0)}{\|\mathbf{b} - (\|\mathbf{b}\|\, 0 \cdots 0)\|}$$

とおく。このとき、

$$P^t = \left(\begin{array}{c|c} 1 & \mathbf{0}^t \\ \hline \mathbf{0} & Q^t \end{array}\right) = P$$

であり、また

$$PP = \left(\begin{array}{c|c} 1 & \mathbf{0}^t \\ \hline \mathbf{0} & Q \end{array}\right)\left(\begin{array}{c|c} 1 & \mathbf{0}^t \\ \hline \mathbf{0} & Q \end{array}\right)$$

$$= \left(\begin{array}{c|c} 1 & \mathbf{0}^t \\ \hline \mathbf{0} & QQ \end{array}\right)$$

$$= I$$

が成り立つ。すなわち、（１４）式の変換は

$$PAP = P^{-1}AP$$

となっていて相似変換である。したがって、固有値は変わらない。また、

$$(PAP)^t = PAP$$

が成り立つので、PAP は対称行列である。

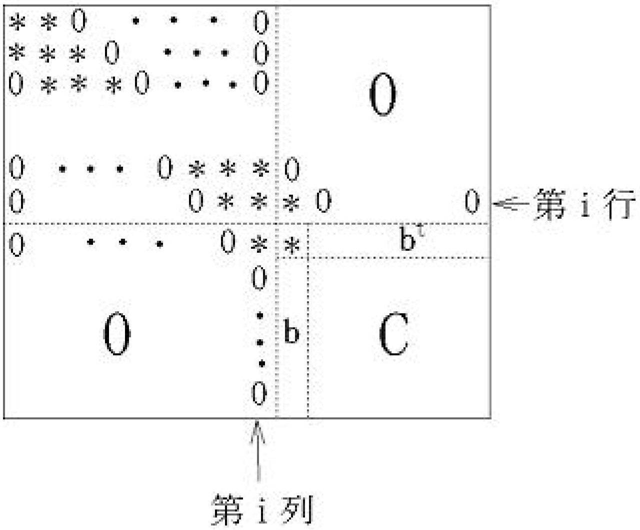


図４　第ｉ行、第ｉ列まで３重対角化された実対称行列

いま、実対称行列の３重対角化がｉ行ｉ列まで行われ、図４のような形になっているとする。続いて、第ｉ＋１行、第ｉ＋１列の３重対角化を次のように行う。

まず、図４の行列を図５のように分割して表わす。

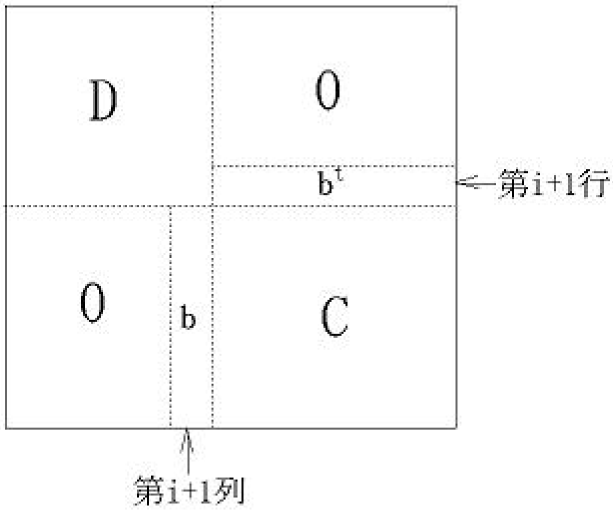


図５　図４の行列の分割

図５の分割に合わせて変換行列も図６のように分割して表わす。

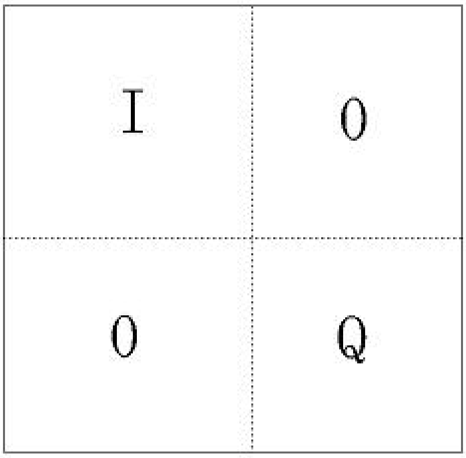


図６　図５の分割に合わせた(n,n)型の変換行列の分割。
I は(,i+1,i+1)型の単位行列、Q は(n-i-1,n-i-1)
型の鏡像変換行列。

図５の行列を A_i で、図６の行列を P_{i+1} で表わすと

$$P_{i+1}A_iP_{i+1} = \begin{pmatrix} I & 0 \\ 0 & Q \end{pmatrix} \left(\begin{array}{c|c} D & \begin{matrix} 0 \\ \mathbf{b}^t \end{matrix} \\ \hline \begin{matrix} 0 & \mathbf{b} \end{matrix} & C \end{array}\right) \begin{pmatrix} I & 0 \\ 0 & Q \end{pmatrix}$$

$$= \begin{pmatrix} I & 0 \\ 0 & Q \end{pmatrix} \left(\begin{array}{c|c} D & \begin{matrix} 0 \\ \mathbf{b}^t Q \end{matrix} \\ \hline \begin{matrix} 0 & \mathbf{b} \end{matrix} & CQ \end{array}\right)$$

$$= \left(\begin{array}{c|c} D & \begin{matrix} 0 \\ \mathbf{b}^t Q \end{matrix} \\ \hline \begin{matrix} 0 & Q\mathbf{b} \end{matrix} & QCQ \end{array}\right)$$

$\mathbf{b}^t Q$ を $\left(\|\mathbf{b}\|\ 0\ \cdots\ 0\right)$ となるように変換すると、第 i+1 列と第 i+1 行が３重対角化される。

このためにQを

$$Q = I - 2\mathbf{u}\mathbf{u}^t \text{、} \mathbf{u} = \frac{\mathbf{b} - (\|\mathbf{b}\| \; 0 \; \cdots \; 0)}{\|\mathbf{b} - (\|\mathbf{b}\| \; 0 \; \cdots \; 0)\|}$$

とおく。このとき、$A_{i+1} = P_{i+1} A_i P_{i+1}$は

$$A_{i+1}^t = (P_{i+1} A_i P_{i+1})^t = P_{i+1} A_i P_{i+1} = A_{i+1}$$

が成り立つので、A_{i+1}は対称行列である。また、

$$P_{i+1} P_{i+1} = \begin{pmatrix} I_{i+1} & 0 \\ 0 & Q \end{pmatrix} \begin{pmatrix} I_{i+1} & 0 \\ 0 & Q \end{pmatrix} = \begin{pmatrix} I_{i+1} & 0 \\ 0 & QQ \end{pmatrix} = \begin{pmatrix} I_{i+1} & 0 \\ 0 & I_{n-i-1} \end{pmatrix} = I_n$$

が成り立つ。ここで、I_{i+1}、I_{n-i-1}、I_nは、それぞれ(i+1,i+1)型、(n-i-1,n-i-1)型、(n,n)型の単位行列を表わす。したがって、

$$A_{i+1} = P_{i+1} A_i P_{i+1} = P_{i+1}^{-1} A_i P_{i+1}$$

は相似変換であり、固有値は変わらない。A_{i+1}はA_iと同じ、すなわち最初の行列Aと同じ

固有値をもつ。

上の方法で、第ｎ行、第ｎ列までの３重対角化を行うことによって、実対称行列Aを同じ固有値を持つ３重対角行列に変換することができる。

ＱＬ法

実対称行列が３重対角行列であるとき、ＱＬ法と呼ばれている方法によって固有値を求めることができる (4)。

ＱＬ法では、まず、実対称３重対角行列Aを直交行列Qと下三角行列Lの積に分解する。

$$A = QL \tag{15}$$

この分解における直交行列Qによって行列Aの相似変換$A^{(1)}$を次式で定める。

$$A^{(1)} = Q^{-1} A Q \tag{16}$$

相似変換で選られた行列$A^{(1)}$に、直交行列と下三角行列の積への分解とそれに基づく相似変換を行う。この直交行列と下三角行列の積への分解と相似変換を繰り返すとき、相似

変換で得られる行列は常に実対称３重対角行列であり、繰り返しが進むと対角行列に収束する。相似変換では固有値は変わらないので、この対角行列は最初の行列Aの固有値を対角成分とするものになっている。

（１５）式より、

$$L = Q^{-1}A$$

したがって、

$$A^{(1)} = Q^{-1}AQ = LQ \qquad （１６ａ）$$

となっている。（１６）式の相似変換は、（１５）式の直交行列Qと下三角行列Lの積による分解において、積の順序を逆にしたLQによって与えられることが式（１６ａ）よりわかる。

（１５）式の分解は、３重対角行列に対して隣り合う行間の回転を下の行から順番に繰り返すことによって下三角行列に変換することによって得られる。

３重対角行列

$$A = \begin{pmatrix} a_{11} & a_{12} & 0 & & \cdots & & 0 \\ a_{21} & a_{22} & a_{23} & 0 & \cdots & & 0 \\ \ddots & & & & & & \\ 0 & & \cdots & 0 & a_{n-1,n-2} & a_{n-1,n-1} & a_{n-1,n} \\ 0 & & \cdots & & 0 & a_{n,n-1} & a_{n,n} \end{pmatrix} \qquad （１７ａ）$$

に対して、まず(n-1,n)成分を０にすることを考える。第 n-1 行と第 n 行の回転を表わす行列

$$P_n = \begin{pmatrix} I & 0 & \\ 0 & \cos q_n & -\sin q_n \\ & \sin q_n & \cos q_n \end{pmatrix} \qquad （１７ｂ）$$

をAの左から掛けた行列$P_n A$の(n-1,n)成分は

$$a_{n-1,n}\cos q_n - a_{n,n}\sin q_n$$

となる。３重対角化はこの成分が０になるようにすることなので、これを０とおいて次式を得る。

$$\cos q_n = \frac{a_{n,n}}{a_{n-1,n}}\sin q_n$$

上式を

$$\cos^2 q_n + \sin^2 q_n = 1$$

に代入して

$$\frac{a_{n,n}^2}{a_{n-1,n}^2}\sin^2 q_n + \sin^2 q_n = 1$$

$$\sin^2 q_n = \frac{a_{n-1,n}^2}{a_{n-1,n}^2 + a_{n,n}^2}$$

となる。$\sin q_n$ の値として

$$\sin q_n = \frac{a_{n-1,n}}{\sqrt{a_{n-1,n}^2 + a_{n,n}^2}} \qquad \text{（１７ｃ）}$$

の方をとると

$$\cos q_n = \frac{a_{n,n}}{\sqrt{a_{n-1,n}^2 + a_{n,n}^2}} \qquad \text{（１８）}$$

となる。

（１７ｃ）、（１８）式で与えられる q_n による（１７ｂ）式の回転 P_n を（１７ａ）式の行列 A の左から掛けると次式のような形の行列が得られる。

$$A_n = \begin{pmatrix} a_{11} & a_{12} & 0 & & \cdots & & & 0 \\ & & & \ddots & & & & \\ & & & & & & & \\ 0 & \cdots & 0 & a_{n-2,n-3} & a_{n-2,n-2} & a_{n-2,n-1} & 0 \\ 0 & \cdots & & 0 & b_{n-1,n-2} & b_{n-1,n-1} & 0 \\ 0 & \cdots & & 0 & b_{n,n-2} & b_{n,n-1} & b_{n,n} \end{pmatrix}$$

いま、隣り合う行間の回転を順番に適用することによって、第ｎ行目から第ｉ行目までの対角成分の右側が次式のように０になったとする。

$$
A_{i+1}=\begin{pmatrix}
a_{11}a_{12}\,0\cdots 0 & & & & & & & & & \\
 & \ddots & & & & & & & & \\
0 & \cdots & 0 & a_{i-1,i-2} & a_{i-1,i-1} & a_{i-1,i} & 0 & \cdots & & 0 \\
0 & \cdots & 0 & & b_{i,i-1} & b_{i,i} & 0 & \cdots & & 0 \\
0 & \cdots & 0 & & b_{i+1,i-1} & b_{i+1,i} & b_{i+1,i+1} & 0 & \cdots & 0 \\
 & & & & \vdots & & & \ddots & & \\
0 & \cdots & 0 & & b_{n,i-1} & & \cdots & & b_{n,n-1} & b_{n,n}
\end{pmatrix}
$$

上の行列の(i-1,i)成分を0にするために次の回転を考える。

$$
P_i=\begin{pmatrix}
I & & & 0 & & \\
0\cdots 0 & \cos q_i & -\sin q_i & 0\cdots 0 \\
0\cdots 0 & \sin q_i & \cos q_i & 0\cdots 0 \\
 & 0 & & & I &
\end{pmatrix}
$$

上の行列P_i**において**cos**と**sin**の置かれている行は第 i-1 行と第 i 行である。すなわち、**P_i**は単位行列の第 i-1 行、第 i 行、第 i-1 列、第 i 列のところを次の２次元の回転**

$$
\begin{pmatrix}
\cos q_i & -\sin q_i \\
\sin q_i & \cos q_i
\end{pmatrix}
$$

で置き換えたものである。

P_i**を**A_{i+1}**の左から掛けた行列**P_iA_{i+1}**の(i-1,i)成分を0とおく。すなわち、**

$$
a_{i-1,i}\cos q-b_{i,i}\sin q_i=0
$$

とする。このとき、上式と

$$
\sin^2 q_i+\cos^2 q_i=1
$$

より

$$\sin^2 q_i + \frac{b_{i,i}}{a_{i-1,i}} \sin^2 q_i = 1$$

$$\sin^2 q_i = \frac{a_{i-1,i}^2}{a_{i-1,i} + b_{i,i}^2}$$

を得る。$\sin q_i$ の値として

$$\sin q_i = \frac{a_{i-1,i}}{\sqrt{a_{i-1,i}^2 + b_{i,i}^2}} \qquad （１９）$$

の方をとると、

$$\cos q_i = \frac{b_{i,i}}{\sqrt{a_{i-1,i}^2 + b_{i,i}^2}} \qquad （２０）$$

となる。

（１９)、(２０）式により決まる q_i で与えられる P_i を A_{i+1} の左から掛けると(i-1,i)成分が０になる。

P_i を A_{i+1} の左から掛けるという操作を、$A_{n+1} = A$ とおいて、$i = n$ から $i = 2$ まで順番に実行して得られる次の行列

$$P_2 P_3 \cdots P_n A = L$$

は下三角行列（対角成分の上側がすべて０の行列）になる。

P_i は回転を表わす行列で

$$P_i^{-1} = P_i^t$$

が成り立つので、

$$Q = (P_2 P_3 \cdots P_n)^{-1} = P_n^t \cdots P_3^t P_2^t$$

とおくと、

$$Q^{-1} A = L$$
$$A = QL$$

となる。すなわち、（１５）式の分解が得られる。上式に基づいて（１６ａ）式による A の相似変換

$$A^{(1)}=Q^{-1}AQ=LQ$$

を行う。

$A^{(1)}$が対称行列であることは、

$$\left(A^{(1)}\right)^t=\left(Q^{-1}AQ\right)^t=\left(Q^tAQ\right)^t=Q^tAQ=Q^{-1}AQ=A^{(1)}$$

よりわかる。

$A^{(1)}$が３重対角行列であることは、

$$A^{(1)}=LQ=LP_n^t\cdots P_2^t$$

において、P_n^t、・・・、P_2^tをLに順番に掛けていった効果を見ていくとわかる。すなわち、

Lは下三角行列であり、P_n^tをLの右から掛けることはLの第 n-1 列と第 n 列の間の回転を行うことになるので、(n-1,n-1)成分の右側が０でなくなる可能性があるが、それ以外の対角成分より上側の０はそのままである。したがって、

$$LP_n^t=\begin{pmatrix} b_{11} & 0 & & \cdots & & 0 \\ b_{21} & b_{22} & 0 & & \cdots & 0 \\ & & \ddots & & & \\ b_{n-2,1} & \cdots & b_{n-2,n-2} & 0 & & 0 \\ b_{n-1,1} & \cdots & \mathrm{b}_{\mathrm{n-1,n-2}} & b_{n-1,n-1} & 0 & \\ b_{n,1} & \cdots & \mathrm{b}_{\mathrm{n,n-2}} & \mathrm{b}_{\mathrm{n,n-1}} & b_{n,n} & \end{pmatrix}P_n^t$$

$$=\begin{pmatrix} b_{11} & 0 & & \cdots & 0 \\ b_{21} & b_{22} & 0 & \cdots & 0 \\ & & \ddots & & \\ b_{n-2,1} & \cdots & b_{n-2,n-2} & 0 & 0 \\ b_{n-1,1} & \cdots & \mathrm{b}_{\mathrm{n-1,n-2}} & \mathrm{c}_{n-1,n-1} & \mathrm{c}_{\mathrm{n-1,n}} \\ b_{n,1} & \cdots & \mathrm{b}_{\mathrm{n,n-2}} & \mathrm{c}_{\mathrm{n,n-1}} & \mathrm{c}_{n,n} \end{pmatrix}$$

という形になる。

いま、

$$
LP_n^t \cdots P_{i+1}^t = \begin{pmatrix}
b_{11} & 0 & & \cdots & & 0 & & & \\
b_{21} & b_{22} & 0 & \cdots & & 0 & & & \\
& & \ddots & & & & & & \\
b_{i-1,1} & \cdots & b_{i-1,i-1} & 0 & & \cdots & & & 0 \\
b_{i,1} & \cdots & b_{i,i-1} & c_{i,i} & c_{i,i+1} & 0 & \cdots & & 0 \\
& & & \vdots & & & & & \\
b_{n-2,1} & \cdots & \mathrm{b}_{\mathrm{n-2,i-1}} & \mathrm{c}_{\mathrm{n-2,i}} & \cdots & \mathrm{c}_{n-2,n-1} & & & 0 \\
b_{n-1,1} & \cdots & \mathrm{b}_{\mathrm{n-1,i-1}} & \mathrm{c}_{n-1,i} & & \cdots & & & \mathrm{c}_{\mathrm{n-1,n}} \\
b_{n,1} & \cdots & \mathrm{b}_{\mathrm{n,i-1}} & \mathrm{c}_{\mathrm{n,i}} & & \cdots & & & \mathrm{c}_{n,n}
\end{pmatrix}
$$

とおくと、$LP_n^t \cdots P_{i+1}^t$ の右から第 i-1 列と第 i 列の間の回転を表わす P_i^t を掛けたものは次のようになる。

$$
LP_n^t \cdots P_{i+1}^t P_i^t = \begin{pmatrix}
b_{11} & 0 & & \cdots & & 0 & & & & \\
b_{21} & b_{22} & 0 & \cdots & & 0 & & & & \\
& & \ddots & & & & & & & \\
b_{i-2,1} & \cdots & b_{i-2,i-2} & 0 & & & \cdots & & & 0 \\
b_{i-1,1} & \cdots & \mathrm{b}_{\mathrm{i-1,i-2}} & b_{i-1,i-1} & 0 & & \cdots & & & 0 \\
b_{i,1} & \cdots & \mathrm{b}_{\mathrm{i,i-2}} & b_{i,i-1} & c_{i,i} & c_{i,i+1} & 0 & \cdots & & 0 \\
& & & \vdots & & & & & & \\
b_{n-2,1} & \cdots & \mathrm{b}_{\mathrm{n-2,i-2}} & \mathrm{b}_{\mathrm{n-2,i-1}} & \mathrm{c}_{\mathrm{n-2,i}} & \cdots & \mathrm{c}_{n-2,n-1} & 0 & & \\
b_{n-1,1} & \cdots & \mathrm{b}_{\mathrm{n-1,i-2}} & \mathrm{b}_{\mathrm{n-1,i-1}} & \mathrm{c}_{n-1,i} & & \cdots & & & \mathrm{c}_{\mathrm{n-1,n}} \\
b_{n,1} & \cdots & \mathrm{b}_{\mathrm{n,i-2}} & \mathrm{b}_{\mathrm{n,i-1}} & \mathrm{c}_{\mathrm{n,i}} & & \cdots & & & \mathrm{c}_{n,n}
\end{pmatrix} P_i^t
$$

$$
=\begin{pmatrix}
b_{11} & 0 & & \cdots & 0 & & & & \\
b_{21} & b_{22} & 0 & \cdots & 0 & & & & \\
 & & \ddots & & & & & & \\
b_{i-2,1} & \cdots & b_{i-2,i-2} & 0 & & & \cdots & & 0 \\
b_{i-1,1} & \cdots & b_{i-1,i-2} & c_{i-1,i-1} & c_{i-1,i} & 0 & \cdots & & 0 \\
b_{i,1} & \cdots & b_{i,i-2} & c_{i,i-1} & c'_{i,i} & c_{i,i+1} & 0 & \cdots & 0 \\
 & & \vdots & & & & & & \\
b_{n-2,1} & \cdots & b_{n-2,i-2} & c_{n-2,i-1} & c'_{n-2,i} & \cdots & c_{n-2,n-1} & & 0 \\
b_{n-1,1} & \cdots & b_{n-1,i-2} & c_{n-1,i-1} & c'_{n-1,i} & & \cdots & & c_{n-1,n} \\
b_{n,1} & \cdots & b_{n,i-2} & c_{n,i-1} & c'_{n,i} & & \cdots & & c_{n,n}
\end{pmatrix}
$$

したがって、$i=2$ まで上の操作を繰り返すと

$$
A^{(1)} = LQ = LP_n^t \cdots P_2^t = \begin{pmatrix}
c_{11} & c_{12} & 0 & \cdots & & 0 \\
c_{21} & c_{22} & c_{23} & 0 & \cdots & 0 \\
 & & & \ddots & & \\
c_{n-2,1} & \cdots & & c_{n-2,n-1} & 0 & \\
c_{n-1,1} & \cdots & & c_{n-1,n-1} & c_{n-1,n} & \\
c_{n,1} & \cdots & & c_{n,n-1} & c_{n,n} &
\end{pmatrix}
$$

という形に書けることがわかる。

上式と $A^{(1)} = Q^t AQ$ が対称行列であることから、$A^{(1)}$ は次の形の対称３重対角行列

$$
A^{(1)} = \begin{pmatrix}
a_1^{(1)} & a_1^{(2)} & 0 & & & \cdots & 0 \\
a_1^{(2)} & a_2^{(1)} & a_2^{(2)} & 0 & & \cdots & 0 \\
0 & a_2^{(2)} & & & & & \\
0 & & & & & & \\
 & & & \ddots & & & \\
0 & \cdots & 0 & a_{n-3}^{(2)} & a_{n-2}^{(1)} & a_{n-2}^{(2)} & 0 \\
0 & \cdots & & 0 & a_{n-2}^{(2)} & a_{n-1}^{(1)} & a_{n-1}^{(2)} \\
0 & & \cdots & & 0 & a_{n-1}^{(2)} & a_n^{(1)}
\end{pmatrix}
$$

であることがわかる。したがって、上の対称３重対角行列をあらためて A として、分解

$A = QL$ と相似変換 $A^{(1)} = Q^{-1}AQ = LQ$ を繰り返すことができる。この分解と相似変換を繰り返していくと $A^{(1)}$ は対角行列に収束することが知られている。最初の行列 A と相似変換で得られる $A^{(1)}$ は同じ固有値をもつので、収束した対角行列の対角成分には A の固有値が並んでいることになる。

分解と相似変換の繰り返しの途中で対角成分の隣の成分の一部が０になったときには小さい行列に分けることができる。いま、

$$A^{(1)} = \begin{pmatrix} A_{11} & 0 \\ 0 & A_{22} \end{pmatrix} \qquad \text{（２１）}$$

というように、$A^{(1)}$ 行列が２つの対称３重対角行列 A_{11} と A_{22} に分割される形になったとする。このとき、（１１）式の形は

$$\det(A^{(1)} - lI) = \det(A_{11} - lI_{11}) \cdot \det(A_{22} - lI_{22})$$

と分解できる。ここで、I_{11} および I_{22} は、それぞれ A_{11} および A_{22} と同じ型の単位行列である。上式より、（２１）式の形の行列 $A^{(1)}$ の固有値は、A_{11} と A_{22} の２つの行列の固有値を別個に求めたものを合わせたものとなっていることがわかる。$A^{(1)}$ が（２１）式の形になったときは、$A^{(1)}$ より大きさの小さい A_{11} と A_{22} に対して個別にＱＬ法を適用することにより計算の効率を上げる。

(n,n)型の対称行列 A のハウスホルダー法による３重対角化と、３重対角化された行列に対するＱＬ法による固有値の算出によって固有値 l_1、・・・、l_n が求まると、対応する固有ベクトル $\mathbf{u}_1$、・・・、$\mathbf{u}_n$ は連立一次方程式

$$A\mathbf{u}_i = l_i\mathbf{u}_i$$

の解として求めることができる。

以上の手順をまとめると次のようになる。

（１）実対称行列 A をハウスホルダー法によって３重対角行列 B に変換する。

（２）B の直交行列 Q と下三角行列 L の積 $B = QL$ への分解を、B の隣合う行間の回転 P_i の繰り返しによって得る。

（３）B の相似変換 $B^{(1)} = Q^{-1}BQ = LQ$ を積 LQ として求める。

（３ａ）$B^{(1)}$ が対角行列であれば、その対角成分を固有値とする。

（３ｂ）$B^{(1)}$ が対角行列でないとき、式（２１）のように２つの行列に分解できるときは、それぞれの（部分）行列 B_{11} および B_{22} を B として（２）に戻る。

（３ｃ）$B^{(1)}$ が対角行列でなく、かつ、式（２１）のように２つの行列にも分解できないときは、$B^{(1)}$ を B として（２）に戻る。

（４）固有値 l が求まれば $A\mathbf{u} = l\mathbf{u}$ を解いて固有ベクトル $\mathbf{u}$ を求める。

以上の方法によって固有値と固有ベクトルを求める手続き QL_decomp をユニットファイル UMatCalc2.pas に用意した。QL_decomp のヘッダーは次のようになっている。

```
procedure  QL_decomp( var a                //  入力行列
                          : TMatCalc2;
                      n,                   //  入力行列の行数・列数
                      t_n_eigen            //  求める固有値の数
                          : Longint;
                      var  n_eigen         //  求まった固有値の数
                          : Longint;
                      var lambda           //  lambda[i]：ｉ番目の固有値
                          : TVecCalc2;
                       var eigen_vctr      //  eigen_vctr[*,i]：ｉ番目の
                          : TMatCalc2 );   //               固有ベクトル
```

第１パラメータａに固有値と固有ベクトルを求める実対称行列を表す配列を設定する。第２パラメータｎには、ａに設定した行列のサイズ、(n,n)型であればｎ、を設定する。第３パラメータ t_n_eigen には求める固有値の個数を設定するが、実際に求まった固有値の総数は第４番目のパラメータ n_eigen に返される。求まった固有値と固有ベクトルは、第５番目のパラメータ Lambda と第６番目のパラメータ eigen_vctr に返される。lambda[i]にｉ番目の固有値、eigen_vctr[j,i]にｉ番目の固有値に対応する固有ベクトルの第ｊ成分が格納される。

手続き QL_decomp の使用例をプロジェクト PCheckQL.dpr として用意した。このプロジェクトでは、QL_decomp が次のように呼出されている。

```
a[1,1]:=5.0;  a[1,2]:=3.0;
a[2,1]:=3.0;  a[2,2]:=5.0;
QL_decomp( a, 2, 2, n_eigen, lambda, eigen_vctr );
```

手続き QL_decomp による計算は、プロジェクトの実行時フォームの「ＧＯ」ボタンのクリックで始まります。lambda と eigen_vctr に返された固有値と固有ベクトルはテキストファイルに書き出されますが、このファイルの名前は「ＧＯ」ボタンのクリックで表示されるダイアログボックスにおいて設定する。

特異値分解

対称行列Aは（6）式のようにスペクトル分解ができる。

$$A = l_1\mathbf{u}_1\mathbf{u}_1^t + l_2\mathbf{u}_2\mathbf{u}_2^t + \cdots + l_n\mathbf{u}_n\mathbf{u}_n^t \qquad (6)$$

一般の(m,n)型の行列Aのときは、次のように分解できる[5]。

$$A = m_1\mathbf{u}_1\mathbf{v}_1^t + m_2\mathbf{u}_2\mathbf{v}_2^t + \cdots + m_r\mathbf{u}_r\mathbf{v}_r^t \qquad (22)$$

ここで、rは行列Aの階数、$m_i > 0$、$\mathbf{u}_1$、・・・、$\mathbf{u}_r$は互いに直交する長さ１の(m,1)型の列ベクトル、$\mathbf{v}_1$、・・・、$\mathbf{v}_r$は互いに直交する長さ１の(n,1)型の列ベクトルである。したがって、

$$A\mathbf{v}_i = m_i\mathbf{u}_i、\quad \mathbf{u}_i^t A = m_i\mathbf{v}_i \qquad (22\mathrm{a})$$

が成り立つ。

（２２）式の形の分解を特異値分解（singular value decomposition, SVD）といい、m_iを特異値（singular value)、$\mathbf{u}_i$を左特異ベクトル、$\mathbf{v}_i$を右特異ベクトルという[2]。

(n,1)型のベクトル$\mathbf{x}$に式（２２）の行列Aを左から掛けると

$$\begin{aligned} A\mathbf{x} &= m_1\mathbf{u}_1\mathbf{v}_1^t\mathbf{x} + m_2\mathbf{u}_2\mathbf{v}_2^t\mathbf{x} + \cdots + m_r\mathbf{u}_r\mathbf{v}_r^t\mathbf{x} \\ &= m_1\mathbf{u}_1(\mathbf{v}_1^t\mathbf{x}) + m_2\mathbf{u}_2(\mathbf{v}_2^t\mathbf{x}) + \cdots + m_r\mathbf{u}_r(\mathbf{v}_r^t\mathbf{x}) \end{aligned} \qquad (23)$$

となる。ここで、$\mathbf{v}_1$、・・・、$\mathbf{v}_r$は互いに直交する長さ１の列ベクトル、すなわち、r次元空間の正規直交基底（座標軸）になっていて、$\mathbf{v}_i^t\mathbf{x}$は$\mathbf{x}$の$\mathbf{v}_i$への正射影の長さ、すなわち$\mathbf{x}$の$\mathbf{v}_i$軸での座標値である。式（２３）は、ベクトル$\mathbf{x}$の行列Aによる変換$A\mathbf{x}$は、$\mathbf{x}$の$\mathbf{v}_i$への正射影$\mathbf{v}_i^t\mathbf{x}$に重みm_iを付けた値を$\mathbf{u}_i$軸上の座標$m_i(\mathbf{v}_i^t\mathbf{x})\mathbf{u}_i = m_i\mathbf{u}_i(\mathbf{v}_i^t\mathbf{x})$とするものであることを表わしている。

いま、

$$U = (\mathbf{u}_1 \cdots \mathbf{u}_r)、\ \Delta = \begin{pmatrix} m_1 & 0 & \cdots & & 0 \\ 0 & m_2 & 0 & \cdots & 0 \\ & & \ddots & & \\ 0 & & \cdots & 0 & m_r \end{pmatrix}、\ V = (\mathbf{v}_1 \cdots \mathbf{v}_r)$$

とおくと、（２２）式は

$$A=(\mathbf{u}_1\cdots\mathbf{u}_r)\begin{pmatrix} m_1 & 0 & \cdots & 0 \\ 0 & m_2 & 0\ \cdots & 0 \\ & & \ddots & \\ 0 & \cdots & 0 & m_r \end{pmatrix}\begin{pmatrix}\mathbf{v}_1^t \\ \vdots \\ \mathbf{v}_r^t\end{pmatrix}=U\Delta V^t$$

となる。ここで、

$$U^tU=\begin{pmatrix}\mathbf{u}_1^t \\ \vdots \\ \mathbf{u}_r^t\end{pmatrix}(\mathbf{u}_1\cdots\mathbf{u}_r)$$

$$=\begin{pmatrix} \mathbf{u}_1^t\mathbf{u}_1 & \mathbf{u}_1^t\mathbf{u}_2 & \cdots & & \mathbf{u}_1^t\mathbf{u}_r \\ \mathbf{u}_2^t\mathbf{u}_1 & \mathbf{u}_2^t\mathbf{u}_2 & \mathbf{u}_2^t\mathbf{u}_3 & \cdots & \mathbf{u}_2^t\mathbf{u}_r \\ & & \ddots & & \\ \mathbf{u}_r^t\mathbf{u}_1 & \cdots & & \mathbf{u}_r^t\mathbf{u}_{r-1} & \mathbf{u}_r^t\mathbf{u}_r \end{pmatrix}$$

$$=\begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & 0\ \cdots & 0 \\ & & \ddots & \\ 0 & \cdots & 0 & 1 \end{pmatrix}$$

$$=I$$

である。

同様に、

$$V^tV=I$$

が成り立つ。

したがって、

$$\begin{aligned} AA^t &= (U\Delta V^t)(U\Delta V^t)^t \\ &= U\Delta V^tV\Delta U^t \\ &= U\Delta I\Delta U^t \\ &= U\Delta^2U^t \end{aligned}$$

$$\begin{aligned} A^tA &= (U\Delta V^t)^t(U\Delta V^t) \\ &= V\Delta U^tU\Delta V^t \\ &= V\Delta^2V^t \end{aligned}$$

が成り立つ。

上式を（８ａ）式と比較することにより、m_i^2 と $\mathbf{u}_i$ は AA^t の固有値と固有ベクトル、m_i^2 と $\mathbf{v}_i$ は A^tA の固有値と固有ベクトルになっていることがわかる。また、$\mathbf{u}_i$ と $\mathbf{v}_i$ の間には（２２ａ）式の関係が成り立っている。したがって、行列 A の特異値分解は、行列 AA^t あるいは A^tA のスペクトル分解によって求めることができる。この方法で特異値分解を求める手続き SVD をユニットファイル UMatCalc2.pas に用意した。手続き SVD のヘッダーは次のようになっている。

```
procedure  SVD( var a              //   入力行列
                   : TMatCalc2;
                m,                 //   入力行列の行数
                n                  //   入力行列の列数
                   : Longint;
                var u              //   u[*,i]:ｉ番目の左特異ベクトル
                   : TMatCalc2;
                var lambda         //   lambda[i]:ｉ番目の特異値
                   : TVecCalc2;
                var  rnk           //   ０でない特異値の数
                   : Longint;
                var v              //   v[*,i]：ｉ番目の右特異ベクトル
                   : TMatCalc2     );
```

(m,n)型の行列の特異値分解を求めるときは、第１パラメータａにその行列を表す配列、第２パラメータｍ、第３パラメータｎにその行数と列数を設定する。特異値分解の結果は、第４パラメータｕに左特異ベクトル、第５パラメータ lambda に特異値、第６パラメータ rnk に求まった特異値の数、第７パラメータに右特異ベクトルが返される。lambda[i]にｉ番目の特異値、u[j,i]に対応する左特異ベクトルの第ｊ成分、v[j,i]に右特異ベクトルの第ｊ成分が格納される。例えば、次のように呼出す。

```
a[1,1]:=3.0;  a[1,2]:=3.0;  a[1,3]:=1.0;  a[1,4]:=1.0;
a[2,1]:=1.0;  a[2,2]:=1.0;  a[2,3]:=3.0;  a[2,4]:=3.0;

SVD( a, 2, 4, u, lambda, n_eigen, v );
```

プロジェクト PCheckSVD.dpr は上の例を実行するものである。実行開始時に表示される

フォーム上の「GO」ボタンのクリックで計算が始まる。計算結果は、「GO」ボタンのクックで表示されるダイアログボックスにおいて設定した名前のテキストファイルに書き出される。このテキストファイルに書き出された結果は、プロジェクトの実行終了後、エディタで開いて見ることができる。

主成分分析

　スペクトル分解とか特異値分解はデータ分析で広く用いられている。ここではスペクトル分解を用いる分析法の例として、主成分分析のプログラムを作成した。

　ある対象を多数の側面から測定しても、本質的には少数のものでそれらの測定値が表わされるということがある。例えば、男子の体格を１０個の側面、身長、座高、胸幅、足長、足幅、手幅、頸囲、胸囲、体重、頭囲、で測ったとき、これらの値は、体格のよさを表わす値（成分）で全体の５８%が説明され、体形を表わす値（成分）で全体の１３%が説明されるというデータがある[5]。この場合、１０個の変数よりなるデータの変動量の７０%以上が２個の成分で表わされる。この多数の変数で表わされているデータを幾つかの少数の成分で表わすための分析法の一つとして主成分分析法[5][6]がある。

　いま、１つ１つのデータがｐ個の変数、z_1、・・・、z_p、で表わされているとする。このとき、これらの表わすデータを合成変数 f で出来るだけよく表わすことを考える。

　各変数 z_i は平均が０、分散が１であるように基準化されているものとする。すなわち、データの素点からその平均点を引いたものを標準偏差で割っておく。この基準化によってｐ個の変数の変動量（分散）はお互いに等しくなる。平均値を０（原点）に置いているので、原点（平均値）を中心とした変動を扱うのが容易になっている。

　合成変数 f として、次のように z_1、・・・、z_p の重み付き和を考える。

$$f = w_1 z_1 + \cdots + w_p z_p$$

この合成変数は、上式より平均が０になっている。さらに、分散が１であるという条件のもとで、合成変数 f が z_1、・・・、z_p の変動（情報）をよく表わすように重み、w_1、・・・、w_p、を決める。

　f と z_i との相関係数を a_i とおき、

$$\mathbf{a} = (a_1 \cdots a_p)^t$$

とおく。z_1、・・・、z_p の変動（情報）をよく表わす合成変数 f を、a_i の２乗和 $\mathbf{a}^t\mathbf{a}$ を最大にするいう基準を満たすものとして求める。この $\mathbf{a}$ は、z_1、・・・、z_p の間の相関を成分とする行列（相関行列）を $\mathbf{R}$ とおくとき、ある数 l に対して次式を満たすものとして求めることができる[(5)]。

$$\mathbf{R}\mathbf{a} = l\mathbf{a}、\quad \mathbf{a}^t\mathbf{a} = l$$

$\mathbf{a}^t\mathbf{a}$ を最大にするということから、l は $\mathbf{R}$ の最大固有値であることがわかる。

このa に対して、重み w_1、・・・、w_p を成分とするベクトル

$$\mathbf{w} = (w_1 \cdots w_p)^t$$

は次式で与えられる。

$$\mathbf{w} = \frac{1}{l}\mathbf{a}$$

合成変数を複数個、f_1、・・・、f_r、とるときは、

$$f_j = w_{j1}z_1 + \cdots + w_{jp}z_p、\quad j = 1, \cdots, r$$

とおき、各 f_j の分散が１でかつ互いに直交するという条件の下で、f_j と z_1、・・・、z_p の相関係数の２乗和が最大になるようにする。f_j と z_1、・・・、z_p の相関係数を成分とするベクトルを $\mathbf{a}_j$ とおくとき、$\mathbf{a}_j$ は次式を満たすものとして求められる。

$$\mathbf{R}\mathbf{a}_j = l_j\mathbf{a}_j、\quad \mathbf{a}_j^t\mathbf{a}_j = l_j$$

f_1、・・・、f_r の順で z_1、・・・、z_p の変動量をよく表わすようにするとき、l_j は $\mathbf{R}$ の固有値を大きいものから順にとる。このとき、重みベクトル

$$\mathbf{w}_j = (w_{j1} \cdots w_{jp})^t$$

は次式で与えられる。

$$\mathbf{w}_j = \frac{1}{l_j}\mathbf{a}_j$$

　上のようにして求められた合成変量を主成分という。

　f_j と z_1、・・・、z_p の相関係数の２乗和が

$$\mathbf{a}_j^t\mathbf{a}_j = l_j$$

で与えられ、p 個の変数 z_1、・・・、z_p それぞれの分散が１に標準化されていて、R の p 個の固有値の和が次式

$$\sum_{j=1}^{p} l_j = tr(\mathbf{R}) = p$$

を満たすことから、次式

$$\frac{l_j}{\sum_{k=1}^{p} l_k} = \frac{\mathbf{a}_j^t\mathbf{a}}{p} \qquad (２４)$$

は、合成変量 f_j によって p 個の変量 z_1、・・・、z_p の変動量のどれぐらいの割合が表わされているのかを示していると考えられる。式（２４）の値を合成変量 f_j の寄与率という。

　上の合成変量（主成分）を求める方法は主成分分析と呼ばれている。主成分分析を行うプログラム例 PPCA.dpr を作成した。PPCA.dpr の実行開始時のフォームは図 C1 のようになっている。

図 C1　プロジェクト PPCA.dpr の実行開始時のフォーム

グリッドの列や行は、「追加（変数）」ボタンおよび「追加（データ）」ボタンのクリックで増やすことができる。追加される行および列は、グリッド内のアクティブなセルの後に挿入される。セルは、そのセル内をクリックするとアクティブになる。「削除（データ）」あるいは「削除（変数）」ボタンをクリックすると、アクティブになっているセルの行あるいは列が削除される。これらの操作によってグリッド内の列数および行数を適当に設定しておいて、変数のラベルやデータ値を設定する。

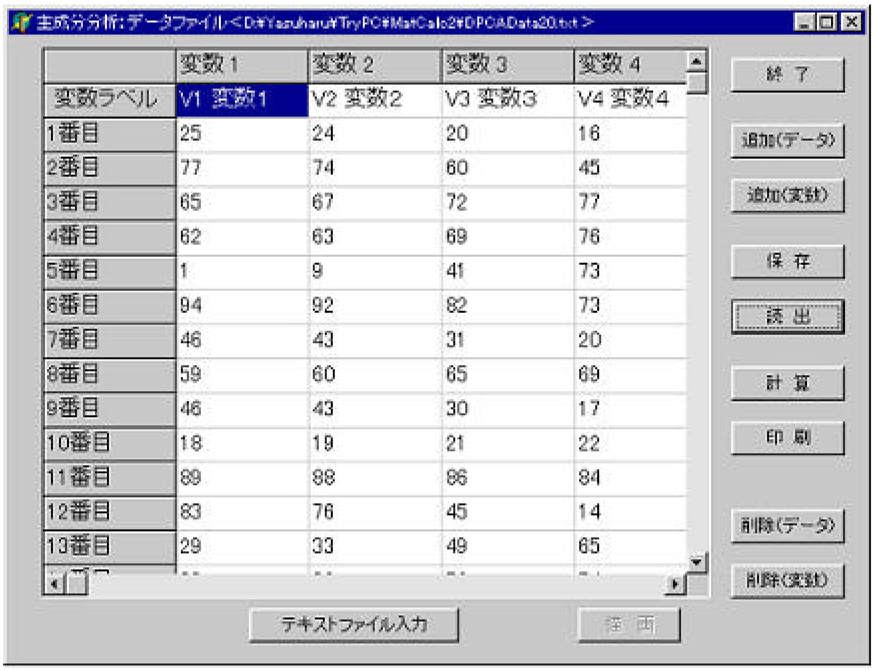


図 C2　データの設定されたグリッド

図 C2 はデータを設定した状態である。データの設定後、「計算」ボタンのクリックで主成分分析の計算が始まる。このとき、グリッド内のセルは、変数ラベル欄を除いて、すなわちデータ値設定用のセルはすべて数値が設定されている必要がある。数値の設定されていないセルがあるとエラーとなる。このエラーは Delphi の try 文で処理されているので、「OK」ボタンをクリックしたり、Delphi の統合環境で実行しているときは「実行｜実行」メニューの選択で図 C3 のようにエラーの原因になったセルを表示させることができる。

図 C3　空白セルのため、「計算」ボタンをクリックするとエラーが表示される。

問題のセルを訂正した後、再度「計算」ボタンをクリックして主成分分析の計算を始める。

グリッドに設定されたデータは「保存」ボタンのクリックでファイルに保存することができる。保存用のファイル名は、「保存」ボタンのクリックで表示されるダイアログボックスにおいて設定する。保存したファイルのデータは、「読出」ボタンのクリックでグリッドに読み込むことができる。

データのグリッド内への読み込みは、リスト C1のような形式のテキストファイルも読み込むことができる。リスト C1の形式のファイルから読み込むときは、「テキストファイル入力」ボタンをクリックする。

リスト C1　「テキストファイル入力」ボタンのクリックで読み込み可能なファイル例

```
          テキストファイル入力ボタンのクリックで
                読み込み可能なファイル

*/
5
V1 変数１
V2 変数２
V3 変数３
V4 変数４
V5 変数５
  1     25    24    20    16    15
  2     77    74    60    45    42
  3     65    67    72    77    78
  4     62    63    69    76    77
  5      1     9    41    73    81
  6     94    92    82    73    70
  7     46    43    31    20    17
  8     59    60    65    69    70
  9     46    43    30    17    14
 10     18    19    21    22    23
 11     89    88    86    84    83
 12     83    76    45    14     7
 13     29    33    49    65    69
 14     33    38    56    74    79
 15     57    55    43    32    29
 16     12    15    27    39    42
 17     79    73    51    28    22
 18     79    77    72    67    66
 19     68    62    40    19    13
 20      5     7    17    26    28
-10
```

リスト C1 の形式は、次のようになっている。

行の先頭が「*/」で始まる行のところまではコメント行として読み込みのときは読み飛ばされて無視される。

「*/」で始まる行の次の行に変数の数を書く。

その次の行から１行に１変数ずつ変数用のラベルを書く。ラベルが無い場合でも空白行として１変数に１行分をとる。

変数の数だけのラベル（空白のラベルも数える）を書いた後、データを書く。各ケース（組）分のデータを１行に書くが、先頭にケース番号を表わす数値を付けておく。

データの読み込みは、負のケース番号を読み込んだところで終了する。リスト C1 では、負のケース番号として「－１０」が書かれている。

グリッドにデータが設定されている状態で「計算」ボタンをクリックすると、計算結果を書き出すためのテキストファイルの名前を設定するダイアログボックスが表示される。この計算結果が書き出されるファイルは、プログラムの実行終了後、エディタなどで開いて見ることができる。図 C2 のデータ（DPCAData20.txt）の場合の計算結果の出力の一部を示すと、リスト C2 のようになっている。

リスト C2　主成分分析のテキストファイルへの出力例の一部

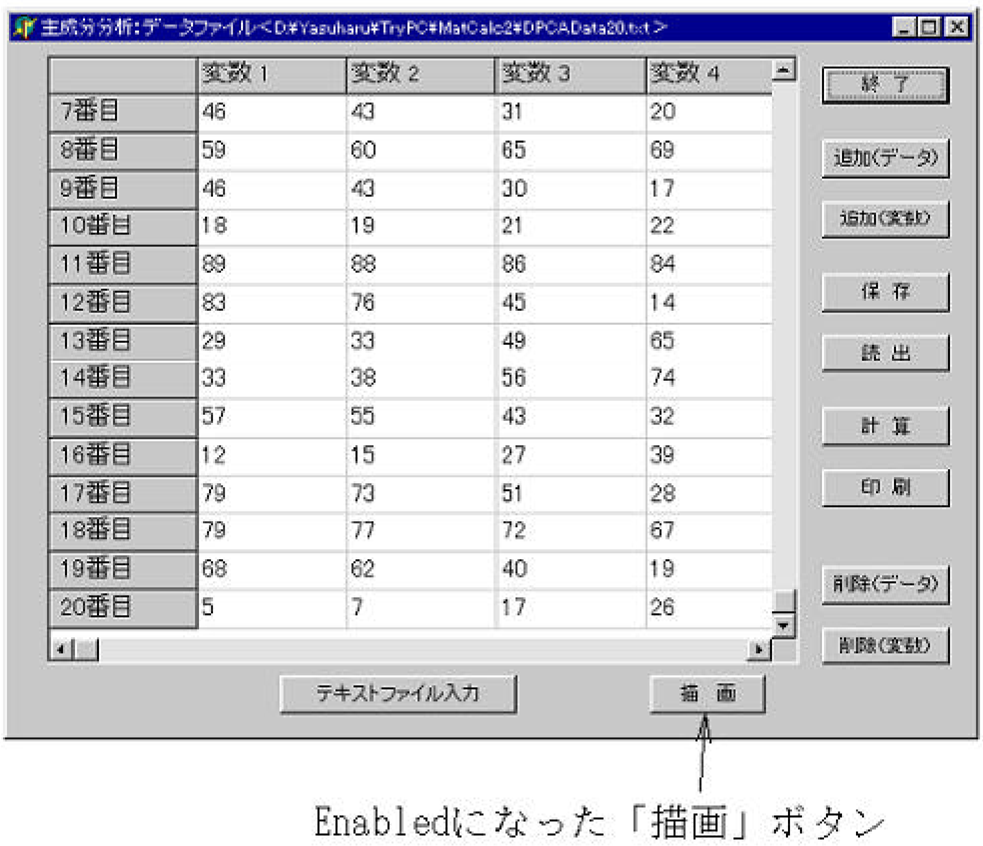
```
相関行列に対する主成分分析

         固有値      寄与率(%) 累積寄与率(%)
      3.3813142        67.63        67.63
      1.6182664        32.37        99.99
      0.0002371         0.00       100.00
      0.0001244         0.00       100.00
      0.0000579         0.00       100.00


主成分構造ベクトル

                   第1主成分      第2主成分      第3主成分     ・・・
      V1 変数１     0.7429406     -0.6692886     -0.0084671    ・・・
      V2 変数２     0.8082172     -0.5888226      0.0030043    ・・・
      V3 変数３     0.9999262     -0.0043984      0.0093447    ・・・
      V4 変数４     0.8029060      0.5960345     -0.0081753    ・・・
      V5 変数５     0.7291281      0.6843469      0.0014846    ・・・


重みベクトル

                   第1主成分      第2主成分      第3主成分     ・・・
      V1 変数１     0.2197195     -0.4135837    -35.7139520    ・・・
      V2 変数２     0.2390246     -0.3638601     12.6720059    ・・・
      V3 変数３     0.2957212     -0.0027180     39.4157506    ・・・
      V4 変数４     0.2374538      0.3683167    -34.4833960    ・・・
      V5 変数５     0.2156345      0.4228889      6.2618563    ・・・


主成分得点 ＝

        ID   第1主成分   第2主成分   第3主成分   第4主成分   第5主成分
         1   -1.40887    -0.16826    -0.37954    -0.07041     0.07830
         2    0.52349    -0.79377     1.96249    -0.11127    -0.70399
         3    1.12620     0.50560     1.05738     0.48869     1.74309
         4    1.00516     0.57600    -1.75702     0.07776    -0.11202
         5   -0.36972     2.24931    -0.21448     0.34422    -0.19492
```

寄与率の第１主成分と第２主成分に対するものの和（累積寄与率）が 99.99%となっていて、この２つの主成分でデータの変動量のほとんどが説明できていることがわかる。この場合のデータ DPCAData20.txt は、次の式によってシミュレーションにより作成されたものである。

```
v1:=random; v2:=random;
x1:=round(100*v1);
x2:=round(90*v1+10*v2);
x3:=round(50*v1+50*v2);
x4:=round(10*v1+90*v2);
x5:=round(100*v2);
```

すなわち、関数 Round による丸めに伴う変動分を除いて、２つの変数 v1 と v2 により５つの変数 x1、・・・、x5 の値が与えられている。このことが、２つの主成分で 99.99%の変動分が説明されるという分析結果になっている。

「計算」ボタンのクリックによる計算が終了すると、「描画」ボタンが Enabled になる（図 C4）。



	変数 1	変数 2	変数 3	変数 4
7番目	46	43	31	20
8番目	59	60	65	69
9番目	46	43	30	17
10番目	18	19	21	22
11番目	89	88	86	84
12番目	83	76	45	14
13番目	29	33	49	65
14番目	33	38	56	74
15番目	57	55	43	32
16番目	12	15	27	39
17番目	79	73	51	28
18番目	79	77	72	67
19番目	68	62	40	19
20番目	5	7	17	26

図 C4　計算終了後、「描画」ボタンが Enabled になる。

「描画」ボタンをクリックすると、図 C5 のようなフォームが表示される。

図 C5　「描画」ボタンのクリックで表示される主成分の選択用フォーム

横軸と縦軸にそれぞれどの主成分をとるのかを指定する。第ｉ主成分を指定するときは数字のｉを設定する。縦軸と横軸の主成分の指定後、「ＯＫ」ボタンをクリックすると図Ｃ６のような画面になり、指定した主成分の構造ベクトルを座標値とする変数の散布図が描かれる。

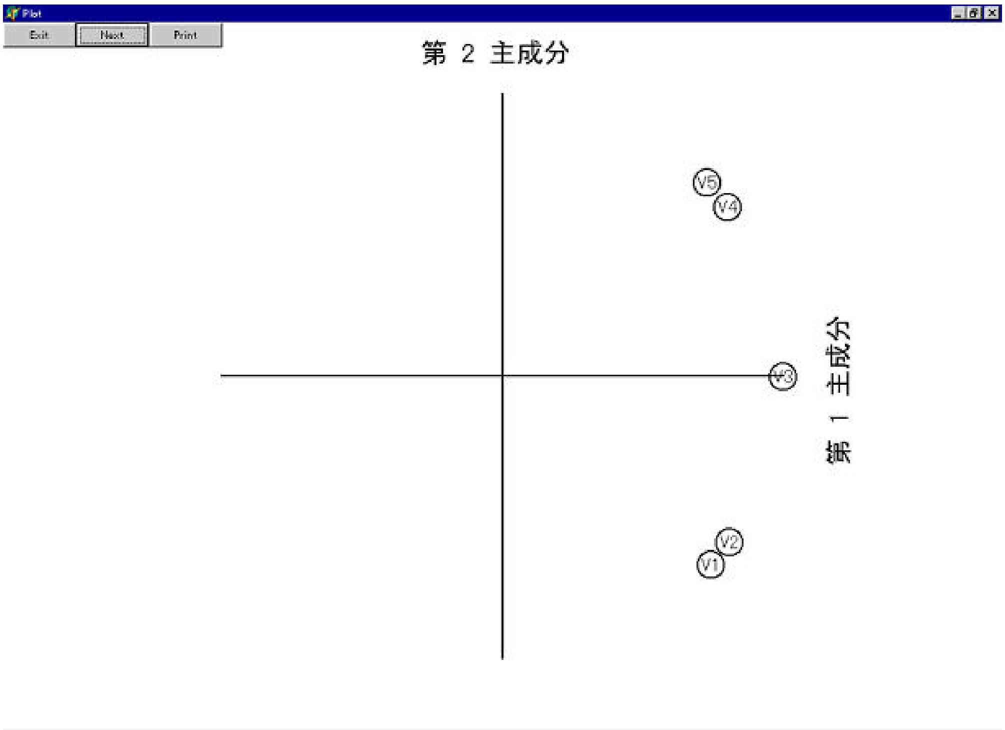


図 C6　主成分構造の図示

図において、各変数はそのラベルを表わす文字列の先頭２バイトの文字列（シフト JIS コード）で表示される。第１変数のラベルは「V1 変数１」なので、先頭２バイトの文字列

「v1」を小円で囲んだもので表示されている。ラベルが１バイトのときは、その１バイト分の文字で表示される。ラベルがないときは小円の中は空白になる。構造ベクトルのマップにおいて小円がどの変数を表わしているのかがわかるように、変数には先頭２バイトの文字（文字列）で識別可能であるようにラベルを付けておく必要がある。

図Ｃ６より、第１主成分は全ての変数に対して構造ベクトルの成分の値（変数との相関係数）が０より大きい値になっていることがわかる。すなわち、第１主成分は全ての変数の変動の全体としての傾向を表わしている。変数 x3 が第１主成分上にプロットされているのは、この変数が５つの変数 x1、・・・、x5 の変動の全体の傾向の中心にあるからである。第１主成分に対して第２主成分は、５つの変数の間の違いを表わしている。x1、・・・、x5 を生成する 2 つの変数 v1 と v2 のうち、v2 の重みの大きいもの、x5 と x4、は第２主成分との相関が正の値で大きく、v1 の重みの方が大きいもの、x1 と x2、は第２主成分との相関が負の値で絶対値が大きくなっている。すなわち、第２主成分は５つの変数を生成した２つの乱数 v1 と v2 の重みの違いを表わしている。

一般逆行列

行列での割り算をおこなうものとして逆行列があるが、逆行列は０でない行列であっても存在するとは限らない。逆行列の存在する行列は正則であるというが、正則でない行列の場合も含めて、逆行列の概念をより一般化した一般逆行列（generalized inverse, g-inverse）[(5)(7)] というものが定義されている。

(n,n)型の行列 A の逆行列 A^{-1} は、次式を満たすものとして定義されている。

$$AA^{-1} = A^{-1}A = I$$

ここで、I は(n,n)型の単位行列（対角成分がすべて１で、他の成分はすべて０である行列）である。(n,n)型の行列 A に対して、逆行列 A^{-1} は常に存在するとは限らない。

逆行列は行数と列数が同じである正方行列に対して定義されているものであるが、一般逆行列は行数と列数が等しくない場合にも定義されているものである。(m,n)型の行列 A の一般逆行列は次式を満たすものとして定義され、A^{-} で表わされる。

$$AA^{-}A = A$$

上式を満たす一般逆行列は、特異値分解から簡単に得ることができる。

A の特異値分解を次のようにおく。

$$A = U\Delta V^{t}$$

上の特異値分解に対して行列 B を次式で与える。

$$B = V\Delta^{-1}U^t$$

ここで、Δ^{-1} は次式で与えられるものである。

$$\Delta^{-1} = \begin{pmatrix} m_1 & 0 & \cdots & 0 \\ 0 & m_2 & 0 \cdots & 0 \\ & & \ddots & \\ 0 & \cdots & 0 & m_r \end{pmatrix}^{-1} = \begin{pmatrix} m_1^{-1} & 0 & \cdots & 0 \\ 0 & m_2^{-1} & 0 \cdots & 0 \\ & & \ddots & \\ 0 & \cdots & 0 & m_r^{-1} \end{pmatrix}$$

このとき、次式が成り立つ。

$$\begin{aligned} ABA &= U\Delta V^t V\Delta^{-1}U^t U\Delta V^t \\ &= U\Delta I\Delta^{-1} I\Delta V^t \\ &= U\Delta V^t \\ &= A \end{aligned}$$

すなわち、行列 B は一般逆行列の条件を満たしているので

$$A^- = V\Delta^{-1}U^t \qquad \textbf{(D 1)}$$

となる。

$$r = rank(A) < k = \min(m,n)$$

のときは、特異値分解における行列 U と V を拡張して、

$$A = U_0\Delta_0V_0^t$$

とおくことができる。ただし、Δ_0 は(k,k)型の対角行列で、

$$\Delta_0 = \begin{pmatrix} \Delta & 0 \\ 0 & 0 \end{pmatrix}$$

$$U_0 = \left(U\mathbf{u}_{r+1}\cdots\mathbf{u}_k\right)、V_0 = \left(V\mathbf{v}_{r+1}\cdots\mathbf{v}_k\right)$$

$$U_0^tU_0 = I 、\qquad V_0^tV_0 = I$$

である。

いま、

$$C = V_0 \begin{pmatrix} \Delta^{-1} & 0 \\ 0 & K \end{pmatrix} U_0^t$$

とおく。ここで、K は(k-r,k-r)型の任意の行列である。
　このとき、

$$\begin{aligned} ACA &= U_0 \begin{pmatrix} \Delta & 0 \\ 0 & 0 \end{pmatrix} V_0^t V_0 \begin{pmatrix} \Delta^{-1} & 0 \\ 0 & K \end{pmatrix} U_0^t U_0 \begin{pmatrix} \Delta & 0 \\ 0 & 0 \end{pmatrix} V_0^t \\ &= U_0 \begin{pmatrix} I & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \Delta & 0 \\ 0 & 0 \end{pmatrix} V_0^t \\ &= U_0 \begin{pmatrix} \Delta & 0 \\ 0 & 0 \end{pmatrix} V_0^t \\ &= A \end{aligned}$$

すなわち、C も一般逆行列である。C は K に対応して無数に決まるので、一般逆行列も無数に存在することになる。
　式（Ｄ１）で与えられる一般逆行列を求める手続き GInvBySVD をユニットファイル UMatCalc2.pas に宣言した。GInvBySVD のヘッダーは次のようになっている。

```
procedure GInvBySVD( a     : TMatCalc2;   //   (m,n) matrix
                     var b : TMatCalc2;   //   (n,m) matrix = g-inv(a)
                     m, n  : Longint    );
```

第１パラメータに、一般逆行列を求める(m,n)型の行列ａを表す配列を設定する。求められた一般逆行列は第２パラメータｂに返される。第３，４パラメータには行列の行数ｍと列数ｎを設定する。一般逆行列の行数はｎ、列数はｍになる。
　手続き GInvBySVD の使用例をプログラム PCalcGInv.dpr として用意した。PCalcGInv.dpr では、行列ａを

```
a[1,1]:=1.0; a[1,2]:=2.0; a[1,3]:=3.0;
a[2,1]:=5.0; a[2,2]:=7.0; a[2,3]:=9.0;
```

と設定した後、GInvBySVD を次のように呼出して配列ｂに一般逆行列を求めている。

```
GInvBySVD( a, b, 2, 3 );
```

プロジェクト PCalcGInv.dpr の実行で表示されるフォーム上の「GO」をクリックすると計算が始まる。計算が始まると、まず計算結果を書き出すためのテキストファイルの名前の設定を求めるダイアログボックスが表示される。計算結果はこのダイアログボックスにおいて設定した名前のファイルに書き出されるので、プログラムの実行終了後にエディタで開いて見ることができる。

掃出し法による一般逆行列の計算

上の特異値分解による方法では、求めた一般逆行列の精度は特異値分解の精度に依存している。ここでは、特異値分解によらない方法、行と列についての基本変形によって得られる標準形 (1) を用いる方法について説明する。

(m,n)型の行列 A が基本変形によって次の標準形になったとする。

$$RAC = \begin{pmatrix} I_r & 0 \\ 0 & 0 \end{pmatrix} = J_{m,n,r}$$

ここで、R および C は、A の行についての基本変形（左基本変形）および列についての基本変形(右基本変形)を表わす行列である。I_r は(r,r)型の単位行列で、(m,n)型の行列 $J_{m,n,r}$ は(1,1)成分から(r,r)成分までの対角成分が１で他の成分はすべて０である行列である。

このとき、

$$B = CJ^t_{m,n,r}R \qquad \text{（D２）}$$

とおくと、

$$\begin{aligned} ABA &= R^{-1}J_{m,n,r}C^{-1}CJ^t_{mn,r}RR^{-1}J_{m,n,r}C^{-1} \\ &= R^{-1}J_{m,n,r}I_nJ^t_{m,n,r}I_mJ_{m,n,r}C^{-1} \\ &= R^{-1}J_{m,n,r}C^{-1} \\ &= A \end{aligned}$$

が成り立つ。すなわち、（D２）で与えられる行列は A の一般逆行列になっている。

$$A^- = CJ^t_{m,n,r}R \qquad \text{（D３）}$$

上式によって一般逆行列を求める手続き GInv をユニットファイル UMatCalc2.pas に用意した。ヘッダーは次のようになっている。

```
procedure GInv( a : TMatCalc2;       //   (m,n) matrix
                var b : TMatCalc2;   //   (n,m) matrix
                m, n  : Longint;
                var rank : Longint );
```

第１パラメータから第４パラメータまでは GInvBySVD と同じであるが、第５パラメータが加えられている。第５パラメータには、第１パラメータに設定された行列ａの階数が返される。したがって、GInv の使用法も GInvBySVD と同様で、

```
GInv( a, b, m, n, rank );
```

というように GInv を呼出すと、ｂにａの一般逆行列が返される。

ムーア・ペンローズ逆行列

式（Ｄ１）で与えられる一般逆行列は、次の４つの式を満たす。

$$AA^{-}A = A、\quad A^{-}AA^{-} = A^{-}、\quad (AA^{-})^{t} = AA^{-}、\quad (A^{-}A)^{t} = A^{-}A$$

上の４つの式を満たす一般逆行列はムーア・ペンローズ逆行列と呼ばれ、A^{+} で表わされる。したがって、次式が成り立つ。

（ＭＰ１）$AA^{+}A = A$
（ＭＰ２）$A^{+}AA^{+} = A^{+}$
（ＭＰ３）$(AA^{+})^{t}AA^{+}$
（ＭＰ４）$(A^{+}A)^{t} = A^{+}A$

式（Ｄ３）によって与えられる一般逆行列は、例えば

$$AA^{-} = R^{-1}J_{m,n,r}C^{-1}CJ_{m,n,r}^{t}R$$

$$= R^{-1}J_{m,n,r}R$$

となり、一般には $R^{t} = R^{-1}$ が成り立たないので（ＭＰ３）も成り立たず、ムーア・ペンローズ逆行列ではない。

特異値分解を用いないでムーア・ペンローズ逆行列を求める方法として、ペンローズによる次の方法がある[7]。

（1）$B \leftarrow A^t A$ とおく。
（2）$C_1 \leftarrow I$、$k \leftarrow rank(A)$、$i \leftarrow 1$ とおく。
（3）$i \ge k$ なら（6）に跳ぶ。
（4）$C_{i+1} \leftarrow \dfrac{tr(C_i B)}{i} I - C_i B$ とおく。
（5）$i \leftarrow i+1$ とおき、（3）に戻る。
（6）$A^+ \leftarrow \dfrac{k}{tr(C_k B)} C_k A^t$ とおく。

上のペンローズの方法でムーア・ペンローズ逆行列 A^+ を求める手続き GInvByPenrose をユニットファイル UMatCalc2.pas に用意した。ヘッダーは

```
procedure GInvByPenrose( a : TMatCalc2;        //   (m,n) matrix
                         var aplus : TMatCalc2;   //   (n,m) matrix
                         m, n  : Longint;
                         var rank : Longint );
```

となっていて手続き GInv と同じである。使い方も GInv と同じである。

参 考 文 献

（1）斎藤正彦「線型代数入門」、Pp.278、東京大学出版会、1966.
（2）I.Borg and P.Groenen. Modern multidimensional scaling: Theory and Applications. Pp.471, Springer-Verlag New York,Inc., 1997.
（3）戸川隼人「マトリクスの数値計算」、Pp.323、オーム社、1971.
（4）R.L.Burden and J.D.Faires. *Numerical Analysis, 3rd ed.*, Pp.676, PWS Publishers, 1985
（5）柳井晴夫「多変量データ解析法—理論と応用—」、Pp.198、朝倉書店、1994.
（6）W.W.Cooley and P.R.Lohnes. Multivariate data analysis. Pp.364,John Wiley & Sons,Inc., 1971.
（7）J.D.Carroll, P.E.Green (with contributions by A.Chaturvedi). Mathematical tools for applied multivariate analysis, revised edition. Pp.376, Academic Press, 1997.