0.12 ジュニア版　　行列の転置と微分

最小２乗法などの計算を行列で行なうために使われる演算の決まりについて GAUSS の上で数値的に確かめておこう。そして、線形回帰の行列表現からその誤差ベクトルを出して、その成分の２乗和を最小にするパラメータを行列の微分計算によって求めてみよう。

行列の転置

行列の転置とは、これまでに扱ったように、行列の行と列を置き換えたものである。それ自身または任意の複数の行列の転置には次のような決まりがある。

$(A')' = A$

$(A+B)' = A'+B'$

$(AB)' = B'A'$

$(A'A)' = A'A$

プログラム

```
new;
cls;
A={1 2 3,
   4 5 6,
   8 8 9};
B={2 1 3,
   3 5 2,
   5 8 7};
print "A=" A;
print "B=" B;
print "(A')'=" (A')';
print;
print "(A+B)'=" (A+B)';
print "A'+B'=" A'+B';
```

画面表示

```
A=
        1.0000000       2.0000000       3.0000000
        4.0000000       5.0000000       6.0000000
        8.0000000       8.0000000       9.0000000
B=
        2.0000000       1.0000000       3.0000000
        3.0000000       5.0000000       2.0000000
```

```
          5.0000000       8.0000000       7.0000000
(A')'=
          1.0000000       2.0000000       3.0000000
          4.0000000       5.0000000       6.0000000
          8.0000000       8.0000000       9.0000000

(A+B)'=
          3.0000000       7.0000000       13.000000
          3.0000000       10.000000       16.000000
          6.0000000       8.0000000       16.000000
A'+B'=
          3.0000000       7.0000000       13.000000
          3.0000000       10.000000       16.000000
          6.0000000       8.0000000       16.000000
```

プログラム

```
new;
cls;
A={1 2 3,
   4 5 6,
   8 8 9};
B={2 1 3,
   3 5 2,
   5 8 7};
print "(A*B)'=" (A*B)';
print " B'A'=" B'A';
print;
print "(A'A)'=" (A'A)';
print " A'A   =" A'A;
```

画面表示

```
(A*B)'=
          23.000000       53.000000       85.000000
          35.000000       77.000000       120.00000
          28.000000       64.000000       103.00000
 B'A'=
          23.000000       53.000000       85.000000
          35.000000       77.000000       120.00000
```

28.000000	64.000000	103.00000

(A'A)'=

81.000000	86.000000	99.000000
86.000000	93.000000	108.00000
99.000000	108.00000	126.00000

A'A =

81.000000	86.000000	99.000000
86.000000	93.000000	108.00000
99.000000	108.00000	126.00000

行列の微分

複数の列ベクトルや行列を含む微分には次のような決まりがある。なお、βとxはともに列ベクトルであるものとする。Aは行列とする。

$$\frac{\partial \beta' x}{\partial x} = \frac{\partial x' \beta}{\partial x} = \beta$$

$$\frac{\partial x' A x}{\partial x} = (A + A')x$$

プログラム

```
new;
cls;
b= 3,2,1 ;
fn f(x)=b'x;
fn g(x)=x'b;

x0= 3,8,5 ;
print gradp(&f,x0)';
x1= 2,3,7 ;
print gradp(&g,x1)';
```

画面表示

```
3.0000000
2.0000000
1.0000000

2.9999999
1.9999999
```

0.99999999

上のプログラムでは、数値微分を求める組込み関数 gradp を用いた際に出てきた答えは転置されていることに気をつけよう。GAUSS では次の計算のことを考えて出てくる答えは転置されたものとなる。そのため、上の計算ではその答えをさらに転置させてもとに戻している。１番目の評価点ｘ0 での微分はβに等しいことがわかる。２番目の評価点ｘ1 での微分は完全にはβに等しくはならなくて、若干の誤差があることがわかる。これは GAUSS の微分が数値微分であって記号的な微分で求められた式に評価点の値を入れるという方法を取っていないためである。

プログラム

```
new;
cls;
A={1 2 3,
   4 5 6,
   7 8 9};
fn h(x)=x'*A*x;

x0={3,8,5};
print gradp(&h,x0)';
print "(A+A')x0=" (A+A')*x0;
```

画面表示

```
        104.00000
        168.00000
        232.00000
(A+A')x0=
        104.00000
        168.00000
        232.00000
```

線形回帰の行列表現と最小２乗法の導出

これら行列の転置と微分の決まりを用いて最小２乗法（誤差の２乗和を最小にするパラメータβを求める）計算をその基礎から理解してみよう。

線形回帰の行列表現

今、簡単化のため、５つのデータからなる説明変数Ｘと５つのデータからなる従属変数ｙを考えよう。定数項なしのモデルを考えることにすると、

$$y=\begin{pmatrix}2\\5\\6\\8\\9\end{pmatrix}, X=\begin{pmatrix}1\\2\\3\\4\\5\end{pmatrix}, e=\begin{pmatrix}e_1\\e_2\\e_3\\e_4\\e_5\end{pmatrix}\text{に対して、}y=X\beta+e$$

すなわち

$$\begin{pmatrix}2\\5\\6\\8\\9\end{pmatrix}=\begin{pmatrix}1\\2\\3\\4\\5\end{pmatrix}\beta+\begin{pmatrix}e_1\\e_2\\e_3\\e_4\\e_5\end{pmatrix}$$

というふうに表すことができる。同様にして、今度は２種類の５つのデータからなる説明変数Ｘに対して、定数項なしのモデルを考えることにすると、

$$y=\begin{pmatrix}2\\5\\6\\8\\9\end{pmatrix}, X=\begin{pmatrix}1&3\\2&6\\3&4\\4&3\\5&5\end{pmatrix}, e=\begin{pmatrix}e_1\\e_2\\e_3\\e_4\\e_5\end{pmatrix}\text{に対して、}y=X\beta+e$$

すなわち

$$\begin{pmatrix}2\\5\\6\\8\\9\end{pmatrix}=\begin{pmatrix}1&3\\2&6\\3&4\\4&3\\5&5\end{pmatrix}\begin{pmatrix}\beta_1\\\beta_2\end{pmatrix}+\begin{pmatrix}e_1\\e_2\\e_3\\e_4\\e_5\end{pmatrix}=\begin{pmatrix}\beta_1+3\beta_2+e_1\\2\beta_1+6\beta_2+e_2\\3\beta_1+4\beta_2+e_3\\4\beta_1+3\beta_2+e_4\\5\beta_1+5\beta_2+e_5\end{pmatrix}$$

というふうに表すことができる。３種類以上（３列以上）の説明変数Ｘについても同様に表すことができる。何種類の（何列の）説明変数になろうとも、ｙ＝Ｘβ＋e というふうに表すことができる。上の場合、計算した結果は両辺とも５行１列の列ベクトルになる。

誤差ベクトル表現

誤差からなる列ベクトルをあらためてｕとすればｙとＸそれに係数ベクトルβを用いて

$$u=y-X\beta$$

と表すことができる。ここで、Ｘは何種類の（何列の）データであっても構わない。直前の例の２種類のデータからなる説明変数Ｘのケースで考えてみると、

$$u=\begin{pmatrix}2\\5\\6\\8\\9\end{pmatrix}-\begin{pmatrix}1&3\\2&6\\3&4\\4&3\\5&5\end{pmatrix}\begin{pmatrix}\beta_1\\\beta_2\end{pmatrix}$$

というふうに表せる。当然のことながら、ｕは列ベクトル（縦ベクトル）である。

誤差の２乗和（u'u）から最小２乗パラメータの導出

最小２乗方というのは、例えば、定数項つきの１種類の説明変数のケースの話をすると、データを説明変数と従属変数との面にプロットしてみて、そのだいたい中心を通るように推定線を引いてみる。それぞれのデータプロットからその推定線に対して上下に垂直方向に下ろした線分の長さの２乗の全体の和を最小にすれば、その推定線はほぼデータの関係を表す線になるというアイディアである。ここで、それぞれのプロットから推定線に垂直方向におろした（のぼった）線分の和を最小にしてもよいのだが、その際には絶対値を考えなくてはならず困難をともなうので、線分の２乗を考えている。

これまで考えたことからわかるように、誤差の２乗和は

$$u'u=(y-X\beta)'(y-X\beta)$$

のことで、列の転置かける列で、結果は１×１の数値となる。これを最小にするパラメータβが計算したい係数である。それには、βに関する微分を行列に関する計算の決まりを用いて計算して、それを０と置いたものをβに関して解く。冒頭の行列の転置の計算の決まりを使うと次のようになる。

$$\min_{\beta}\,(y-X\beta)'(y-X\beta)=\min_{\beta}\,y'y-\beta'X'y-y'X\beta+\beta'X'X\beta$$

すなわち、βに関して上の式を微分すると、最初の項にはβはないから、行列の微分の計算より

$$-X'y-(y'X)'+2X'X\beta$$

$$=-2X'y+2X'X\beta$$

となる。さらにこれを０と置いて、この式が最小になるβを求める。すなわち、

$$-2X'y+2X'X\beta=0$$

$$X'X\beta=X'y$$

$$\therefore \quad \beta=(X'X)^{-1}X'y$$

上の式はＸが何種類の（何列の）データからなる説明変数であろうと成り立つ。

プログラム

```
new;
cls;
y={2,
   5,
   6,
   8,
   9};
X={1 3,
   2 6,
   3 4,
   4 3,
   5 5};

b=inv(X'X)*X'y;
print "     b=" b;
```

画面表示

```
   b=
         1.6908769
        0.21877768
```

定数項（１の列ベクトル）だけの線形回帰の意味

上では、いくつ説明変数があろうとも u＝y－Xβの２乗和、すなわち u'u を最小にするパラメータはβ＝(X'X)$^{-1}$X'y となることがわかった。Xにはデータ行列と定数項があれば１の列ベクトルが水平方向にマージされたものがくることが普通だが、定数項に対する１の列ベクトルだけの１列からなる列ベクトルを用いて、従属変数ｙに対する線形回帰をしてみよう。今、説明変数ｘが１の列ベクトルであるとすると、

$x = \begin{pmatrix} 1 \\ 1 \\ \vdots \\ \vdots \\ 1 \end{pmatrix}$であるとして、その行数をｎとすると、

$$x'x = \begin{pmatrix} 1 & 1 & \cdots & \cdots & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ \vdots \\ \vdots \\ 1 \end{pmatrix} = n \text{および} x'y = \begin{pmatrix} 1 & 1 & \cdots & \cdots & 1 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ \vdots \\ y_3 \end{pmatrix} = \sum_{i=1}^{n} y_i$$

１×１の数値に対するインバースは何分の１ということを意味するから、結局

$$\beta = (x'x)^{-1} x'y = \frac{1}{n} \sum_{i=1}^{n} y_i = \bar{y} \text{（ｙの平均）}$$

ということになる。１ばかりの列ベクトルを説明変数にして最小２乗法を計算すると従属変数の平均を求めたことになるのである。

この１ばかりの列ベクトルを「定数項」を求めるものとしてその他の何列かからなる説明変数データとマージして全体を説明変数Ｘとして、従属変数ｙに対して線形回帰すると、これまでは原点から回帰していたものが、その定数項分だけ平行移動した地点をあらためて新しい原点とした線形回帰がなされることになるのである。

以下では、１ばかりの列ベクトルだけを説明変数にした時の係数βが、従属変数の平均に等しくなることを示しておこう。

プログラム

```
new;
cls;
y={2,
   5,
   6,
   8,
```

```
    9};
X={1,
    1,
    1,
    1,
    1};

b=inv(X'X)*X'y;
ybar=meanc(y);
print "     b=" b;
print "ybar=" ybar;
```

画面表示

```
    b=          6.0000000
ybar=          6.0000000
```

定数項付き線形回帰のケース

上では、１ばかりの列ベクトルだけを説明変数に最小２乗法を行なうと、その係数βは従属変数ｙの平均に等しくなることがわかった。もう一度、一般的な複数の列のデータからなる説明変数を考えよう。今、説明変数の第１列目にこの１ばかりの列ベクトルを加えて線形回帰を行なう。

プログラム

```
new;
cls;
y={2,
    5,
    6,
    8,
    9};
X={1 3,
    2 6,
    3 4,
    4 3,
    5 5};
X=ones(5,1)~X;

b=inv(X'X)*X'y;
```

```
print "     b=" b;
```

画面表示

```
    b=
        0.14328358
          1.6805970
        0.19402985
```

上では、１ばかりの列を加えるのに ones という組込み関数を用いて、５行１列の１ばかりの行列を作っておいて、それをもとの説明変数Ｘに水平方向にマージしたものをあらためてＸとしている。すなわち、x_1 および x_2 を１ばかりの列を除く１、２番目の説明変数とすると、

$$y = 0.14328358 + 1.6805970x_1 + 0.19402985x_2$$

というような関係式で表せる。一方、定数項を含まない方法で計算した結果は、

$$y = 1.6908769x_1 + 0.21877768x_2$$

という関係式であった。定数項を含まない下の式は、x_1 および x_2 に０を入れると同時に従属変数ｙも０になる。このことは、この関係式が原点０を通るものであることを意味する。すなわち、原点０を通るように事実上「制約を加えて」線形回帰がなされていたのである。それに対して、１ばかりの列を加えた計算では、x_1 および x_2 に０を入れると従属変数ｙは１番目の係数（すなわち定数項の値）0.14328358 となる。この場合は、原点０は通らない。このことは、従属変数ｙからこの定数項の値を引いたそれぞれの値をあらためて従属変数として、定数項なしの線形回帰を行なうと、今度はぴったりと原点０を通ることを意味する。定数項が正の値であれば、線形回帰がｙの正方向にちょうど定数項の大きさだけシフトしたところを原点にして定数項なしの線形回帰をしたことに相当し、反対に、定数項が負の値であれば、線形回帰がｙの負方向にちょうど定数項の絶対値の値だけシフトしたところを原点にして定数項なしの線形回帰をしたことに相当するのである。定数項を除くその他の係数がすべて０に限りなく近ければ、その線形回帰の定数項は従属変数ｙのほぼ平均になるが、通常このようなことは生じない。なぜなら、（定数項を除いた数で）１説明変数のケースにしろ２説明変数以上のケースにしろ、従属変数ｙに対してその線形回帰がなにがしらの傾きβをもっているからである。