データ解析

Rによる多変量解析入門

正準相関分析と判別分析

lec20040115 下平英寿 shimo@is.titech.ac.jp

概要

- 正準相関分析

  二つのデータ行列 X と Y の関係

  少数個の合成変量を用いてわかりやすく表現

- 判別分析

  データ行列 X と分類群

  少数個の合成変量を用いて入力ベクタの分類

正準判別分析は正準相関分析の分類問題への応用である．

二つのデータ行列

データ行列 X

$$\boldsymbol{X} = \underbrace{\begin{bmatrix} x_{11} & \dots & x_{1p} \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \cdot & & \cdot \\ x_{n1} & \dots & x_{np} \end{bmatrix}}_{p} \Bigg\} n = \begin{bmatrix} \boldsymbol{x}^{(1)} \\ \cdot \\ \cdot \\ \cdot \\ \boldsymbol{x}^{(n)} \end{bmatrix} = [\boldsymbol{x}_1, \dots, \boldsymbol{x}_p]$$

$\boldsymbol{x}^{(i)}$ は行ベクトル，$\boldsymbol{x}_j$ は列ベクトル

各列の平均を引き去って「中心化」してあるものと仮定して議論を進める

$$\boldsymbol{X} \leftarrow \boldsymbol{X} - \tfrac{1}{n}\boldsymbol{1}_n\boldsymbol{1}_n'\boldsymbol{X}$$

データ行列 Y

$$\boldsymbol{Y} = \underbrace{\begin{bmatrix} y_{11} & \dots & y_{1q} \\ \cdot & & \cdot \\ \cdot & & \cdot \\ \cdot & & \cdot \\ y_{n1} & \dots & y_{nq} \end{bmatrix}}_{q} \Bigg\} n = \begin{bmatrix} \boldsymbol{y}^{(1)} \\ \cdot \\ \cdot \\ \cdot \\ \boldsymbol{y}^{(n)} \end{bmatrix} = [\boldsymbol{y}_1, \dots, \boldsymbol{y}_q]$$

$\boldsymbol{y}^{(i)}$ は行ベクトル，$\boldsymbol{y}_j$ は列ベクトル

各列の平均を引き去って「中心化」してあるものと仮定して議論を進める

$$\boldsymbol{Y} \leftarrow \boldsymbol{Y} - \tfrac{1}{n}\boldsymbol{1}_n\boldsymbol{1}_n'\boldsymbol{Y}$$

データ行列

```
> ### データ行列
> a1 <- grep("^A04",X2000$code,value=T)
> a2 <- c("A02102","A02103","A02104")
> jna1 <- paste(seq(along=a1),a1,X2000$jitem[a1])
> jna2 <- paste(seq(along=a2),a2,X2000$jitem[a2])
> jna1
 [1] "1 A0410301 未 婚 者 割 合［20～24歳・男］ "
 [2] "2 A0410302 未 婚 者 割 合［20～24歳・女］ "
 [3] "3 A0410401 未 婚 者 割 合［25～29歳・男］ "
 [4] "4 A0410402 未 婚 者 割 合［25～29歳・女］ "
 [5] "5 A0410501 未 婚 者 割 合［30～34歳・男］ "
 [6] "6 A0410502 未 婚 者 割 合［30～34歳・女］ "
 [7] "7 A0410601 未 婚 者 割 合［35～39歳・男］ "
 [8] "8 A0410602 未 婚 者 割 合［35～39歳・女］ "
 [9] "9 A0410701 未 婚 者 割 合［40～44歳・男］ "
[10] "10 A0410702 未 婚 者 割 合［40～44歳・女］ "
[11] "11 A0410801 未 婚 者 割 合［45～49歳・男］ "
[12] "12 A0410802 未 婚 者 割 合［45～49歳・女］ "
```



```
[13] "13 A0430701 死 別 者 割 合［60歳以上・男］ "
[14] "14 A0430702 死 別 者 割 合［60歳以上・女］ "
[15] "15 A0440501 離 別 者 割 合［40～49歳・男］ "
[16] "16 A0440502 離 別 者 割 合［40～49歳・女］ "
[17] "17 A0440601 離 別 者 割 合［50～59歳・男］ "
[18] "18 A0440602 離 別 者 割 合［50～59歳・女］ "
> jna2
[1] "1 A02102 人 口 性 比［15歳未満人口］ "
[2] "2 A02103 人 口 性 比［15～64歳人口］ "
[3] "3 A02104 人 口 性 比［65歳以上人口］ "
> xx1 <- X2000$x[,a1]; xx2 <- X2000$x[,a2]
> cbind(xx1,xx2)[1:5,]
         A0410301 A0410302 A0410401 A0410402 A0410501 A0410502 A0410601
Hokkaido     91.2     85.7     64.8     52.6     39.0     28.3     23.2
Aomori       90.0     83.3     64.0     48.9     40.3     24.2     26.0
Iwate        88.7     82.2     63.7     48.2     42.1     24.2     29.3
Miyagi       91.3     86.4     66.7     52.6     41.6     26.1     25.8
Akita        90.6     84.5     64.8     48.7     40.6     22.7     27.1
         A0410602 A0410701 A0410702 A0410801 A0410802 A0430701 A0430702
Hokkaido     16.5     16.3     11.0     12.2      7.8      9.0     38.4
Aomori       12.5     19.5      8.0     14.7      5.6      8.9     42.9
Iwate        12.6     22.4      7.7     16.8      5.6      9.5     41.5
Miyagi       12.9     18.5      7.8     13.9      5.4      9.1     38.6
Akita        11.5     19.1      6.5     14.4      4.6      9.1     42.9
         A0440501 A0440502 A0440601 A0440602 A02102 A02103 A02104
Hokkaido      4.7      8.7      5.5      8.4  104.6   94.0   74.7
Aomori        4.7      8.3      5.8      7.9  103.8   96.1   67.6
Iwate         3.9      6.5      4.6      6.1  105.0   99.1   69.4
Miyagi        3.6      6.0      4.4      6.4  105.4  101.1   72.0
Akita         4.3      6.7      4.6      6.2  104.7   97.1   68.1
```

正準相関分析

相関を最大にする射影

$$\boldsymbol{X} = n \times p\text{行列}, \quad \boldsymbol{Y} = n \times q\text{行列}, \quad n \geq p \geq q \geq 1$$

$$\boldsymbol{f} = \boldsymbol{X}\boldsymbol{a}, \quad \boldsymbol{g} = \boldsymbol{Y}\boldsymbol{b}$$

$$\sigma_f^2 = \tfrac{1}{n-1}\boldsymbol{f}'\boldsymbol{f}, \quad \sigma_{fg} = \tfrac{1}{n-1}\boldsymbol{f}'\boldsymbol{g}, \quad \sigma_g^2 = \tfrac{1}{n-1}\boldsymbol{g}'\boldsymbol{g}$$

相関係数 $r_{fg} = \dfrac{\sigma_{fg}}{\sigma_f \sigma_g}$ を最大にする係数ベクトル $\boldsymbol{a}, \boldsymbol{b}$ の組を探す

$\sigma_f^2 = \sigma_g^2 = 1$ **の条件付で** $r_{fg} = \sigma_{fg}$ **を最大にする**

$\boldsymbol{f}$ と $\boldsymbol{g}$ を正準変量，r_{fg} を正準相関と呼ぶ

```
> ## 正準変量，正準相関
> cc <- mycancor(xx1,xx2)
> cc$xcoef[,1]
   A0410301     A0410302     A0410401     A0410402     A0410501     A0410502
```



```
-0.250675683  0.123458405 -0.020660917  0.064348915 -0.152135140  0.082958152
    A0410601     A0410602     A0410701     A0410702     A0410801     A0410802
 0.178551547 -0.053490586 -0.209977531 -0.116131537  0.007432648  0.27230795
    A0430701     A0430702     A0440501     A0440502     A0440601     A0440602
-0.021652769  0.206633834 -0.112270088  0.362064102 -0.181190209 -0.15447428
> cc$ycoef[,1]
    A02102      A02103      A02104
 0.1398500 -0.1776664 -0.1084593
> xx1c <- scale(xx1,sc=F); xx2c <- scale(xx2,sc=F) # 中心化
> f <- xx1c %*% cc$xcoef[,1]; g <- xx2c %*% cc$ycoef[,1] # 正準変量
> cbind(f,g)
                 [,1]         [,2]
Hokkaido   0.61910223  0.427650145
Aomori     0.48409580  0.712732086
Iwate      0.01816858  0.152326180
Miyagi    -0.31794928 -0.429060828
Akita      0.45722151  0.606701048
Yamagata  -0.48541606 -0.402789963
Fukushima -0.41541009 -0.523879355
```

```
...省略...
Saga       1.59699053  1.500038656
Nagasaki   1.78313041  1.572458539
Kumamoto   1.12368939  1.204420358
Ooita      1.35554904  1.275481306
Miyazaki   1.24781941  1.297035177
Kagoshima  1.32116396  1.352056609
Okinawa    0.05234008 -0.033253243
> var(cbind(f,g))
          [,1]      [,2]
[1,] 1.0000000 0.9889504
[2,] 0.9889504 1.0000000
```

正準相関の導出(ステップ1)

$$\Sigma_{XX} = \tfrac{1}{n-1}\boldsymbol{X}'\boldsymbol{X}, \quad \Sigma_{XY} = \tfrac{1}{n-1}\boldsymbol{X}'\boldsymbol{Y}, \quad \Sigma_{YY} = \tfrac{1}{n-1}\boldsymbol{Y}'\boldsymbol{Y}$$

$$\sigma_f^2 = \boldsymbol{a}'\Sigma_{XX}\boldsymbol{a}, \quad \sigma_{fg} = \boldsymbol{a}'\Sigma_{XY}\boldsymbol{b}, \quad \sigma_g^2 = \boldsymbol{b}'\Sigma_{YY}\boldsymbol{b},$$

$\boldsymbol{a}'\Sigma_{XX}\boldsymbol{a} = \boldsymbol{b}'\Sigma_{YY}\boldsymbol{b} = 1$ の条件付で $\boldsymbol{a}'\Sigma_{XY}\boldsymbol{b}$ を最大化するには，ラグランジュの未定乗数を λ_a，λ_b として次の関数の極値を考える．

$$\psi(\boldsymbol{a}, \boldsymbol{b}, \lambda_a, \lambda_b) = 2\boldsymbol{a}'\Sigma_{XY}\boldsymbol{b} - \lambda_a(\boldsymbol{a}'\Sigma_{XX}\boldsymbol{a} - 1) - \lambda_b(\boldsymbol{b}'\Sigma_{YY}\boldsymbol{b} - 1)$$

$$\frac{1}{2}\frac{\partial\psi}{\partial\boldsymbol{a}} = \Sigma_{XY}\boldsymbol{b} - \lambda_a\Sigma_{XX}\boldsymbol{a} = 0, \quad \frac{1}{2}\frac{\partial\psi}{\partial\boldsymbol{b}} = \Sigma_{YX}\boldsymbol{a} - \lambda_b\Sigma_{YY}\boldsymbol{b} = 0$$

上記の2式にそれぞれ $\boldsymbol{a}', \boldsymbol{b}'$ を左からかけると

$$\boldsymbol{a}'\Sigma_{XX}\boldsymbol{b} = \lambda_a\boldsymbol{a}'\Sigma_{XX}\boldsymbol{a} = \lambda_b\boldsymbol{b}'\Sigma_{YY}\boldsymbol{b}$$

したがって $\lambda_a = \lambda_b = \boldsymbol{a}'\Sigma_{XX}\boldsymbol{b}$ となる．これを λ とかく．

$$r_{fg} = \lambda$$

正準相関の導出(ステップ2)

$$\Sigma_{XY}\boldsymbol{b} = \lambda\Sigma_{XX}\boldsymbol{a}, \quad \Sigma_{YX}\boldsymbol{a} = \lambda\Sigma_{YY}\boldsymbol{b}$$

$$\Sigma_{XX}^{-1}\Sigma_{XY}\Sigma_{YY}^{-1}\Sigma_{YX}\boldsymbol{a} = \lambda^2\boldsymbol{a}$$
$$\Sigma_{YY}^{-1}\Sigma_{YX}\Sigma_{XX}^{-1}\Sigma_{XY}\boldsymbol{b} = \lambda^2\boldsymbol{b}$$

ここで $\boldsymbol{X}$ と $\boldsymbol{Y}$ の特異値分解を用いて式を整理する．

$$\boldsymbol{X} = \boldsymbol{U}_X\boldsymbol{D}_X\boldsymbol{V}_X', \quad \boldsymbol{Y} = \boldsymbol{U}_Y\boldsymbol{D}_Y\boldsymbol{V}_Y', \quad \boldsymbol{U}_X'\boldsymbol{U}_Y = \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}'$$

$$\tilde{a} = \tfrac{1}{\sqrt{n-1}}\boldsymbol{D}_X\boldsymbol{V}_X'\boldsymbol{a}, \quad \tilde{b} = \tfrac{1}{\sqrt{n-1}}\boldsymbol{D}_Y\boldsymbol{V}_Y'\boldsymbol{b}$$

によって $\boldsymbol{U}_X, \boldsymbol{D}_X, \boldsymbol{V}_X, \boldsymbol{U}_Y, \boldsymbol{D}_Y, \boldsymbol{V}_Y, \boldsymbol{U}, \boldsymbol{D}, \boldsymbol{V}, \tilde{a}, \tilde{b}$ を定義する．

$$\Sigma_{XX} = \tfrac{1}{n-1}\boldsymbol{V}_X\boldsymbol{D}_X^2\boldsymbol{V}_X', \quad \Sigma_{YY} = \tfrac{1}{n-1}\boldsymbol{V}_Y\boldsymbol{D}_Y^2\boldsymbol{V}_Y'$$

$$\Sigma_{XY} = \tfrac{1}{n-1}\boldsymbol{V}_X\boldsymbol{D}_X\boldsymbol{U}\boldsymbol{D}\boldsymbol{V}'\boldsymbol{D}_Y\boldsymbol{V}_Y'$$

正準相関の導出(ステップ3)

$$\boldsymbol{U}\boldsymbol{D}^2\boldsymbol{U}'\tilde{a} = \lambda^2\tilde{a}, \quad \boldsymbol{V}\boldsymbol{D}^2\boldsymbol{V}'\tilde{b} = \lambda^2\tilde{b}, \quad \|\tilde{a}\| = \|\tilde{b}\| = 1$$

特異値分解 $\boldsymbol{U}_X'\boldsymbol{U}_Y = \boldsymbol{U}\boldsymbol{D}\boldsymbol{V}'$ だったので $\boldsymbol{U} = [\boldsymbol{u}_1, \ldots, \boldsymbol{u}_q] = p \times q$ 行列，$\boldsymbol{V} = [\boldsymbol{v}_1, \ldots, \boldsymbol{v}_q] = q \times q$ 行列，$\boldsymbol{D} = \mathrm{diag}(d_1, \ldots, d_q)$．$d_1 \geq d_2 \geq \cdots \geq d_q \geq 0$ としておく．従って r_{fg} の極値問題の解は

$$\lambda = d_1, \ldots, d_q, \quad \tilde{a} = \boldsymbol{u}_1, \ldots, \boldsymbol{u}_q, \quad \tilde{b} = \boldsymbol{v}_1, \ldots, \boldsymbol{v}_q$$

$r_{fg} = \lambda$ の最大値を得るには $\tilde{a} = \boldsymbol{u}_1$，$\tilde{b} = \boldsymbol{v}_1$，$\lambda = d_1$ とすればよい．

$$\boldsymbol{a} = \sqrt{n-1}\boldsymbol{V}_X\boldsymbol{D}_X^{-1}\tilde{a}, \quad \boldsymbol{b} = \sqrt{n-1}\boldsymbol{V}_Y\boldsymbol{D}_Y^{-1}\tilde{b}$$

$$\boldsymbol{f} = \boldsymbol{X}\boldsymbol{a} = \sqrt{n-1}\boldsymbol{U}_X\tilde{a}, \quad \boldsymbol{g} = \boldsymbol{Y}\boldsymbol{b} = \sqrt{n-1}\boldsymbol{U}_Y\tilde{b}$$

q 組の正準変量：$i = 1, \ldots, q$ に対して

$$\boldsymbol{f}_i = \sqrt{n-1}\boldsymbol{U}_X\boldsymbol{u}_i, \quad \boldsymbol{g}_i = \sqrt{n-1}\boldsymbol{U}_Y\boldsymbol{v}_i, \quad r_{f_i g_i} = d_i$$

正準変量の性質

$$\boldsymbol{f}_i = \boldsymbol{X}\boldsymbol{a}_i, \quad \boldsymbol{g}_i = \boldsymbol{Y}\boldsymbol{b}_i, \quad i = 1, \ldots, q$$

をまとめて行列表現する．

$$\boldsymbol{F} = [\boldsymbol{f}_1, \ldots, \boldsymbol{f}_q], \quad \boldsymbol{A} = [\boldsymbol{a}_1, \ldots, \boldsymbol{a}_q], \quad \boldsymbol{F} = \boldsymbol{X}\boldsymbol{A}$$

$$\boldsymbol{G} = [\boldsymbol{g}_1, \ldots, \boldsymbol{g}_q], \quad \boldsymbol{B} = [\boldsymbol{b}_1, \ldots, \boldsymbol{b}_q], \quad \boldsymbol{G} = \boldsymbol{Y}\boldsymbol{B}$$

すると以下の関係がある．

$$\boldsymbol{A} = \sqrt{n-1}\boldsymbol{V}_X\boldsymbol{D}_X^{-1}\boldsymbol{U}, \quad \boldsymbol{B} = \sqrt{n-1}\boldsymbol{V}_Y\boldsymbol{D}_Y^{-1}\boldsymbol{V}$$

$$\boldsymbol{F} = \sqrt{n-1}\boldsymbol{U}_X\boldsymbol{U}, \quad \boldsymbol{G} = \sqrt{n-1}\boldsymbol{U}_Y\boldsymbol{V}$$

$$\tfrac{1}{n-1}\boldsymbol{F}'\boldsymbol{F} = \boldsymbol{I}_q, \quad \tfrac{1}{n-1}\boldsymbol{G}'\boldsymbol{G} = \boldsymbol{I}_q, \quad \tfrac{1}{n-1}\boldsymbol{F}'\boldsymbol{G} = \boldsymbol{D}$$

$$\tfrac{1}{n-1}\boldsymbol{X}'\boldsymbol{F} = \tfrac{1}{\sqrt{n-1}}\boldsymbol{V}_X\boldsymbol{D}_X\boldsymbol{U}, \quad \tfrac{1}{n-1}\boldsymbol{Y}'\boldsymbol{G} = \tfrac{1}{\sqrt{n-1}}\boldsymbol{V}_Y\boldsymbol{D}_Y\boldsymbol{V}$$

R関数の定義

```
> source("~shimo/class/gakubu200209/myfunc20030111.R") # 関数のロード
## 正準相関分析
mycancor <- function(x1,x2,scale.arg=F) {
  x1 <- scale(as.matrix(x1),scale=scale.arg) # 中心化
  x2 <- scale(as.matrix(x2),scale=scale.arg)
  n <- nrow(x1)
  s1 <- mysvd(x1); s2 <- mysvd(x2) # x1とx2の特異値分解
  ss <- mysvd(t(s2$u) %*% s1$u) # 「Uy'Ux」の特異値分解
#  c1 <- sqrt(n-1) * s1$v %*% diag(1/s1$d) %*% ss$v # 係数(A)
#  c2 <- sqrt(n-1) * s2$v %*% diag(1/s2$d) %*% ss$u # 係数(B)
  c1 <- sqrt(n-1) * s1$v %*% (1/s1$d * ss$v) # 係数(A)
  c2 <- sqrt(n-1) * s2$v %*% (1/s2$d * ss$u) # 係数(B)
  y1 <- x1 %*% c1; y2 <- x2 %*% c2 # 正準変量(F,G)
  v1 <- t(x1) %*% y1 / (n-1) # XとFの共分散
  v2 <- t(x2) %*% y2 / (n-1) # YとGの共分散
  list(cor=ss$d,xcoef=c1,ycoef=c2,xcan=y1,ycan=y2,xcov=v1,ycov=v2)
}
```


正準相関分析の数値例

```
> ### 正準相関分析 (x1=X, x2=Y)
> cc <- mycancor(xx1,xx2)
> cc$cor # 正準相関係数
     PC 1      PC 2      PC 3
0.9889504 0.8932883 0.6563581
> cc$xcoef # 係数(A)


                 PC 1         PC 2        PC 3
  A0410301 -0.250675683  0.243851306  0.47965575
  A0410302  0.123458405 -0.262873112  0.18931438
  A0410401 -0.020660917 -0.130416652  1.23343459
  A0410402  0.064348915  0.147467038 -1.70965113
  A0410501 -0.152135140  0.195889079 -1.13579916
  A0410502  0.082958152  0.218756559  1.82723729
  A0410601  0.178551547 -0.122816255  0.50312081
  A0410602 -0.053490586 -0.621380257 -0.42315662
  A0410701 -0.209977531  0.266951147  0.62742272
```

```
A0410702 -0.116131537 -0.701704340 -0.35730623
A0410801  0.007432648  0.019265615 -0.56016177
A0410802  0.272307951  1.049486992  0.05099804
A0430701 -0.021652769  0.005950758  0.49034290
A0430702  0.206633834  0.293091432  0.17565885
A0440501 -0.112270088  0.354370141  1.93844089
A0440502  0.362064102 -1.068049033 -0.26170475
A0440601 -0.181190209  0.204609862 -2.16572770
A0440602 -0.154474286  0.718542080  0.92737532
> cc$ycoef # 係数(B)

              PC 1       PC 2         PC 3
A02102  0.1398500  0.1813134 -2.088476340
A02103 -0.1776664  0.2831442 -0.007814717
A02104 -0.1084593 -0.3645199  0.021435348
> cc$xcan[1:5,] # 正準変量(F)

                PC 1       PC 2     PC 3
Hokkaido  0.61910223 -2.9668213 1.203897
```

```
Aomori    0.48409580  0.5758445 0.945848
Iwate     0.01816858  1.0875194 1.162894
Miyagi   -0.31794928 -0.1908324 0.221460
Akita     0.45722151  0.4815717 1.850286
> cc$ycan[1:5,] # 正準変量(G)

               PC 1        PC 2        PC 3
Hokkaido  0.4276501 -3.07126315  0.89909607
Aomori    0.7127321 -0.03361941  2.40127526
Iwate     0.1523262  0.37725346 -0.08975687
Miyagi   -0.4290608  0.06831543 -0.88504493
Akita     0.6067010  0.23044690  0.52454951
> var(cc$xcan)
              PC 1          PC 2          PC 3
PC 1  1.000000e+00 -3.349550e-16  3.347493e-17
PC 2 -3.349550e-16  1.000000e-00 -9.834259e-17
PC 3  3.347493e-17 -9.834259e-17  1.000000e-00
> var(cc$ycan)
              PC 1          PC 2          PC 3
```

```
PC 1  1.000000e-00 -1.220574e-16 1.189173e-15
PC 2 -1.220574e-16  1.000000e-00 2.954333e-17
PC 3  1.189173e-15  2.954333e-17 1.000000e-00
> var(cc$xcan,cc$ycan)
             PC 1          PC 2          PC 3
PC 1 9.889504e-01  5.113693e-17  1.291549e-15
PC 2 1.198471e-17  8.932883e-01 -9.493840e-18
PC 3 2.731777e-17 -2.326424e-16  6.563581e-01
> cc$xcov # XとFの共分散

                PC 1        PC 2        PC 3
 A0410301 -1.0640513 -0.85841043  0.10349740
 A0410302 -0.7993054 -1.31091015 -0.02788536
 A0410401 -2.4032480 -0.86420087  0.09272874
 A0410402 -0.4489871 -1.39280454  0.09792115
 A0410501 -2.0300205  0.30715940  0.55008214
 A0410502  0.3704776 -1.03560256  0.89001963
 A0410601 -1.2263788  1.09137382  0.89607693
 A0410602  0.3944501 -0.72621448  0.70053889
```

```
A0410701 -0.8683686  1.18446375  0.79183671
A0410702  0.2259314 -0.55220860  0.49648955
A0410801 -0.8228701  0.93505175  0.67196567
A0410802  0.3146324 -0.29968390  0.30306244
A0430701  0.1753849  0.31530186 -0.20126856
A0430702  1.8266965  1.04420935  0.17093487
A0440501  0.4611338  0.06024694  0.16304250
A0440502  0.9187505  0.02540602  0.29708747
A0440601  0.4068799  0.10110710  0.23261537
A0440602  0.6783375  0.10654667  0.24635532
> cc$ycov # YとGの共分散

               PC 1        PC 2       PC 3
A02102 -0.01625802 -0.02350316 -0.4819471
A02103 -3.82339174  1.12831766 -0.1580684
A02104 -2.97794164 -1.87859354 -0.3625032
> var(xx2,cc$ycan) # 上に一致
             PC 1        PC 2       PC 3
A02102 -0.01625802 -0.02350316 -0.4819471
A02103 -3.82339174  1.12831766 -0.1580684
A02104 -2.97794164 -1.87859354 -0.3625032
```

データ行列の表示

バイプロット

$F = n \times q$行列に直交する適当な$\bar{F} = n \times (p-q)$を用いて

$$X = FK' + \bar{F}\bar{K}', \quad Y = GL'$$

ここでK, LはX, YをF, Gに回帰させた係数とみなせる．すると

$$\frac{1}{n-1}X'F = K, \quad \frac{1}{n-1}Y'G = L$$

２組のバイプロット(q次元まで)が行なえる

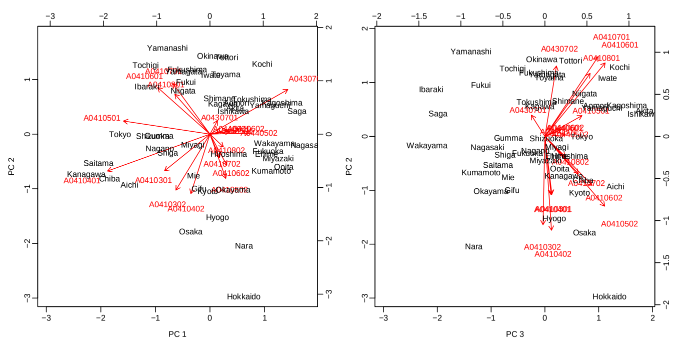
$$(F, K), \quad (G, L)$$

バイプロットの数値例

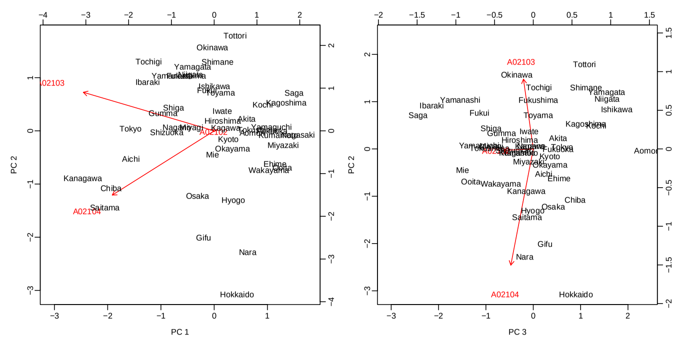
```
> ## バイプロット
> sum( (cc$ycan %*% t(cc$ycov) - scale(xx2,sc=F))^2 ) # これはOK
[1] 1.783360e-27
> t(cc$ycoef) %*% cc$ycov  # 逆行列

            PC 1          PC 2         PC 3
PC 1  1.000000e-00 -1.450663e-16 1.185927e-15
PC 2 -2.339708e-16  1.000000e-00 2.526191e-17
PC 3  1.191430e-15  2.677641e-17 1.000000e+00
> sum( (cc$xcan %*% t(cc$xcov) - scale(xx1,sc=F))^2 ) # これだけではダメ
[1] 2801.381
> t(cc$xcoef) %*% cc$xcov  # 一般逆行列

            PC 1          PC 2          PC 3
PC 1  1.000000e+00 -3.604972e-16 -1.756746e-17
PC 2 -3.981190e-16  1.000000e-00 -1.553797e-16
PC 3 -4.439266e-16 -3.642784e-16  1.000000e-00
```



```
> mybiplot(cc$xcan,cc$xcov)
> mybiplot(cc$xcan,cc$xcov,choi=3:2)
```




```
> mybiplot(cc$ycan,cc$ycov)
> mybiplot(cc$ycan,cc$ycov,choi=3:2)
```



そのほか

データ行列を標準化してから正準相関分析

```
> ### 標準化してから正準相関分析 (x1=X, x2=Y)
> round(sqrt(apply(xx1,2,var)),2)
A0410301 A0410302 A0410401 A0410402 A0410501 A0410502 A0410601 A0410602
    1.93     2.48     3.39     3.42     3.13     3.41     2.69     2.52
A0410701 A0410702 A0410801 A0410802 A0430701 A0430702 A0440501 A0440502
    2.52     1.98     2.21     1.64     0.67     2.40     0.66     1.49
A0440601 A0440602
    0.79     1.35
> round(sqrt(apply(xx2,2,var)),2)
A02102 A02103 A02104
  0.48   3.99   3.54
> cc <- mycancor(xx1,xx2,scale=T) # 標準化も行う
> cc$cor # 正準相関係数は標準化しても変わらない
     PC 1      PC 2      PC 3
0.9889504 0.8932883 0.6563581
> cc$xcoef # 係数(A) は標準化で変化する
```



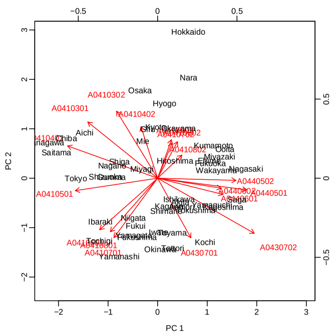
```
                PC 1         PC 2        PC 3
A0410301 -0.48478370 -0.471585983 -0.92761008
A0410302  0.30577496  0.651069602 -0.46888340
A0410401 -0.07003521  0.442078993 -4.18102684
A0410402  0.22038908 -0.505060952  5.85539680
A0410501 -0.47633654 -0.613330530  3.55619772
A0410502  0.28260307 -0.745210379 -6.22461883
A0410601  0.47977242  0.330010311 -1.35189804
A0410602 -0.13457605  1.563319990  1.06461252
A0410701 -0.53002705 -0.673840331 -1.58374571
A0410702 -0.22998568  1.389647920  0.70760550
A0410801  0.01642728 -0.042579943  1.23804286
A0410802  0.44732113 -1.723995587 -0.08377464
A0430701 -0.01453437 -0.003994433 -0.32914157
A0430702  0.49585549 -0.703326231 -0.42152537
A0440501 -0.07367504 -0.232548448 -1.27206378
A0440502  0.53913619  1.590392092  0.38969480
A0440601 -0.14336159 -0.161891720  1.71357031
A0440602 -0.20920859 -0.973140460 -1.25596882
```

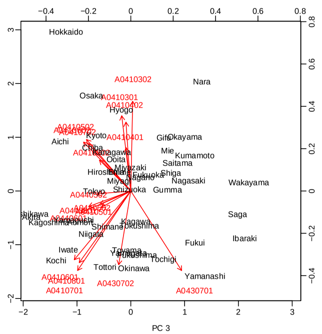
```
> cc$ycoef # 係数(B)

              PC 1        PC 2        PC 3
A02102  0.06751869 -0.08753696  1.00830317
A02103 -0.70880665 -1.12961457  0.03117711
A02104 -0.38390099  1.29024914 -0.07587223
> cc$xcan[1:5,] # 正準変量(F) は標準化で変化しない.

               PC 1       PC 2      PC 3
Hokkaido  0.61910223  2.9668213 -1.203897
Aomori    0.48409580 -0.5758445 -0.945848
Iwate     0.01816858 -1.0875194 -1.162894
Miyagi   -0.31794928  0.1908324 -0.221460
Akita     0.45722151 -0.4815717 -1.850286
> cc$ycan[1:5,] # 正準変量(G)

              PC 1        PC 2        PC 3
Hokkaido 0.4276501  3.07126315 -0.89909607
Aomori   0.7127321  0.03361941 -2.40127526
```

```
Iwate     0.1523262 -0.37725346  0.08975687
Miyagi   -0.4290608 -0.06831543  0.88504493
Akita     0.6067010 -0.23044690 -0.52454951
> cc$xcov # XとFの共分散

               PC 1        PC 2        PC 3
A0410301 -0.5502078  0.44387346 -0.05351723
A0410302 -0.3227242  0.52928755  0.01125887
A0410401 -0.7089764  0.25494580 -0.02735568
A0410402 -0.1310947  0.40666925 -0.02859089
A0410501 -0.6483598 -0.09810236 -0.17568844
A0410502  0.1087537  0.30400120 -0.26126533
A0410601 -0.4564077 -0.40616441 -0.33348295
A0410602  0.1567840  0.28865193 -0.27844654
A0410701 -0.3440162 -0.46924166 -0.31369704
A0410702  0.1140843  0.27883838 -0.25070298
A0410801 -0.3723138 -0.42307118 -0.30403591
A0410802  0.1915333  0.18243339 -0.18449008
A0430701  0.2612819 -0.46972501  0.29984243
```

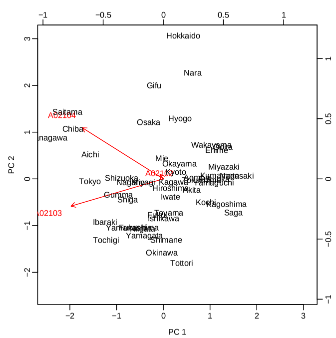
```
A0430702  0.7612244 -0.43514489 -0.07123230
A0440501  0.7027011 -0.09180760 -0.24845315
A0440502  0.6169991 -0.01706175 -0.19951306
A0440601  0.5142427 -0.12778609 -0.29399526
A0440602  0.5008671 -0.07867134 -0.18190248
> cc$ycov # YとGの共分散

             PC 1        PC 2       PC 3
A02102 -0.03367489  0.04868158 0.99824652
A02103 -0.95835459 -0.28281915 0.03962074
A02104 -0.84132519  0.53073843 0.10241404
```

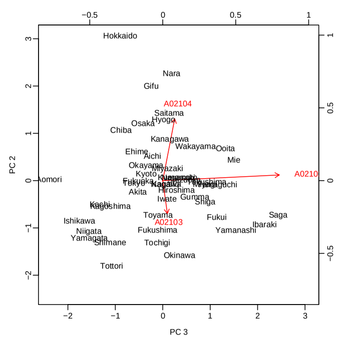
```
> mybiplot(cc$xcan,cc$xcov)
> mybiplot(cc$xcan,cc$xcov,choi=3:2)
```





```
> mybiplot(cc$ycan,cc$ycov)
> mybiplot(cc$ycan,cc$ycov,choi=3:2)
```





正準相関係数の検定

```
mycancor.diag <- function(cc) {
  r <- cc$cor
  lam <- -log(1-r^2)
  cum <- rev(cumsum(rev(lam)))
  n <- nrow(cc$xcan); p1 <- nrow(cc$xcoef); p2 <- nrow(cc$ycoef)
  j <- seq(along=r)-1; deg <- (p1-j)*(p2-j) # 自由度
  xsq <- (n-(p1+p2+1)/2)*cum # カイ二乗統計量
  pv <- pchisq(xsq,deg,lower=F) # 確率値
  list(xsq=xsq,deg=deg,pv=pv)
}
> mycancor.diag(cc)
$xsq
[1] 215.30159  77.86235  20.28722
$deg
[1] 54 34 16
$pv
[1] 3.918338e-21 2.722149e-05 2.075619e-01
```

回帰分析との関係

```
> ### 正準相関分析と回帰分析の関係
> a1 <- c("A02102","A02103","A02104")
> a2 <- "A0410501"
> jna1 <- paste(seq(along=a1),a1,X2000$jitem[a1])
> jna2 <- paste(seq(along=a2),a2,X2000$jitem[a2])
> jna1
[1] "1 A02102 人　口　性　比［15歳未満人口］ "
[2] "2 A02103 人　口　性　比［15〜64歳人口］ "
[3] "3 A02104 人　口　性　比［65歳以上人口］ "
> jna2
[1] "1 A0410501 未　婚　者　割　合［30〜34歳・男］ "
> xx1 <- X2000$x[,a1,drop=F]; xx2 <- X2000$x[,a2,drop=F]
> cc <- mycancor(xx1,xx2)
> cc$cor # 正準相関係数
   PC 1
0.65735
> cc$xcoef # 係数(A)
```



```
                PC 1
  A02102 -0.47860917
  A02103  0.20967632
  A02104  0.06095876
> cc$ycoef # 係数(B)

                PC 1
  A0410501 0.3193858
> ## 回帰分析との比較
> x1 <- scale(xx1,sc=F); x2 <- scale(xx2,sc=F) # 中心化
> f <- lsfit(x1,x2)
> f$coef # 回帰係数
    Intercept        A02102        A02103        A02104
 8.187637e-15 -9.850585e-01  4.315493e-01  1.254634e-01
> cc$xcoef * (cc$cor/cc$ycoef)[[1]] # 回帰係数に一致

               PC 1
  A02102 -0.9850585
```

```
  A02103  0.4315493
  A02104  0.1254634
> cbind(x1 %*% f$coef[-1],
+       cc$xcan * (cc$cor/cc$ycoef)[[1]])[1:10,] # 予測値
                           PC 1
Hokkaido  -1.3766256 -1.3766256
Aomori    -0.5731156 -0.5731156
Iwate     -0.2347037 -0.2347037
Miyagi     0.5605763  0.5605763
Akita     -0.9653872 -0.9653872
Yamagata   1.6989494  1.6989494
Fukushima  1.3757392  1.3757392
Ibaraki    1.9883057  1.9883057
Tochigi    2.8711286  2.8711286
Gumma      1.7771090  1.7771090
```

正準判別分析

要素の分類

要素$\{1,\ldots,n\}$が$q+1$個の群$J_0,J_1,\ldots,J_q$に分けられている．

$$J_0 \cup J_1 \cup \cdots \cup J_q = \{1,\ldots,n\}$$

各群の個数を$|J_i| = n_i$とすると$n_0 + \cdots + n_q = n$

```
> ## 日本の地方の分類
> jap1 <- c("Ibaraki","Okinawa","Miyazaki","Iwate","Tokyo","Aichi",
+    "Wakayama","Kochi","Shizuoka","Kanagawa","Kagoshima","Chiba",
+    "Miyagi","Mie","Fukushima","Tokushima") # 太平洋側
> jap2 <- c("Fukui","Akita","Tottori","Toyama","Shimane","Fukuoka",
+    "Niigata","Ishikawa","Yamagata","Kyoto","Saga") # 日本海側
> jap3 <- c("Saitama","Gifu","Nagano","Shiga","Gumma","Tochigi",
+    "Nara","Yamanashi") # 内陸
> jap4 <- c("Osaka","Okayama","Hiroshima","Kagawa",
+    "Ehime","Ooita") # 瀬戸内
> jap5 <- c("Hokkaido","Aomori","Hyogo","Yamaguchi",
+    "Nagasaki","Kumamoto") # その他
> length(c(jap1,jap2,jap3,jap4,jap5))
```



```
[1] 47
> ib <- c(jap1,jap2,jap3,jap4)
> gb <- sapply(list(jap1,jap2,jap3,jap4),length)
> names(gb) <- c("taihei","nihon","nairiku","setonai")
> gb
 taihei   nihon nairiku setonai
     16      11       8       6
> sum(gb)
[1] 41
> kb <- mydiagk(gb)
> kb
      taihei nihon nairiku setonai
 [1,]      1     0       0       0
 [2,]      1     0       0       0
 [3,]      1     0       0       0
 [4,]      1     0       0       0
 [5,]      1     0       0       0
 [6,]      1     0       0       0
 [7,]      1     0       0       0
```

```
 [8,]      1     0       0       0
 [9,]      1     0       0       0
[10,]      1     0       0       0
[11,]      1     0       0       0
[12,]      1     0       0       0
[13,]      1     0       0       0
[14,]      1     0       0       0
[15,]      1     0       0       0
[16,]      1     0       0       0
[17,]      0     1       0       0
[18,]      0     1       0       0
[19,]      0     1       0       0
[20,]      0     1       0       0
[21,]      0     1       0       0
[22,]      0     1       0       0
[23,]      0     1       0       0
[24,]      0     1       0       0
[25,]      0     1       0       0
[26,]      0     1       0       0
```

```
[27,]      0     1       0       0
[28,]      0     0       1       0
[29,]      0     0       1       0
[30,]      0     0       1       0
[31,]      0     0       1       0
[32,]      0     0       1       0
[33,]      0     0       1       0
[34,]      0     0       1       0
[35,]      0     0       1       0
[36,]      0     0       0       1
[37,]      0     0       0       1
[38,]      0     0       0       1
[39,]      0     0       0       1
[40,]      0     0       0       1
[41,]      0     0       0       1
```

各群への所属を表すベクトル

各要素の群への所属を表す$q+1$個の$n\times 1$ベクトル$h_0,\ldots,h_q$を定義する．

$$\boldsymbol{h}_j\text{の第}i\text{要素は}\quad h_{ij} = \begin{cases} 1 & i \in J_i \\ 0 & i \notin J_i \end{cases}$$

$$\boldsymbol{h}_0 = (\underbrace{1,\ldots,1}_{n_0},\ \underbrace{0,\ldots,0}_{n_1+\ldots+n_q})'$$

$$\boldsymbol{h}_1 = (\underbrace{0,\ldots,0}_{n_0},\underbrace{1,\ldots,1}_{n_1},\ \underbrace{0,\ldots,0}_{n_2+\cdots+n_q})'$$

$$\vdots$$

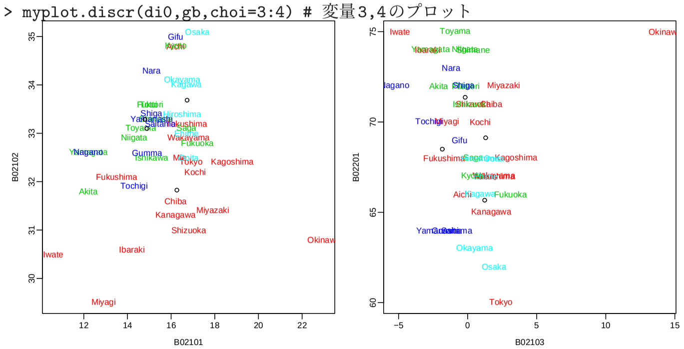
$$\boldsymbol{h}_q = (\ \underbrace{0,\ldots,0}_{n_0+\ldots,n_{q-1}}\ ,\underbrace{1,\ldots,1}_{n_q})'$$

データ行列

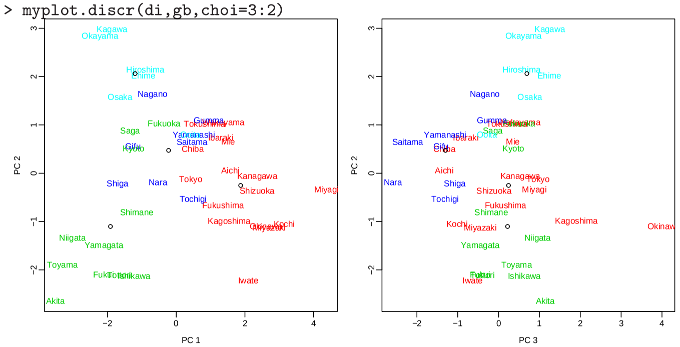
```
> ### 気象データ
> ia <- grep("^B02",X2000$code,value=T)
> ia
[1] "B02101" "B02102" "B02103" "B02201" "B02401" "B02402" "B02301" "B02303"
[9] "B02304"
> jna <- paste(seq(along=ia),ia,X2000$jitem[ia])
> jna
[1] "1 B02101 年 平 均 気 温 "
[2] "2 B02102 最 高 気 温（日最高気温の月平均の最高値） "
[3] "3 B02103 最 低 気 温（日最低気温の月平均の最低値） "
[4] "4 B02201 年 平 均 相 対 湿 度 "
[5] "5 B02401 日 照 時 間（年 間） "
[6] "6 B02402 降 水 量（年 間） "
[7] "7 B02301 快 晴 日 数（年 間） "
[8] "8 B02303 降 水 日 数（年 間） "
[9] "9 B02304 雪 日 数（年 間） "
> xx <- X2000$x[ib,ia]
> xx   # jap1,...,jap4の４群の順に県名が並ぶ．(jap5は含まれない)
```


	B02101	B02102	B02103	B02201	B02401	B02402	B02301	B02303	B02304
Ibaraki	14.2	30.6	-2.9	74	2057	1400	41	103	10
Okinawa	23.0	30.8	14.3	75	1605	2613	12	129	0
Miyazaki	17.9	31.4	2.6	72	2105	2594	49	113	1
Iwate	10.6	30.5	-4.9	75	1702	1418	4	134	110
... 省略 ...									
Nara	15.1	34.3	-1.2	73	1809	1320	27	94	25
Yamanashi	15.1	33.3	-2.2	64	2249	1479	46	91	13
Osaka	17.2	35.1	1.9	62	2009	1164	21	93	14
Okayama	16.5	34.1	0.5	63	2007	813	36	78	17
Hiroshima	16.5	33.4	0.9	68	2065	1139	32	97	21
Kagawa	16.7	34.0	0.9	66	2077	857	25	95	9
Ehime	16.7	33.0	1.3	67	2035	1150	28	101	10
Ooita	16.8	32.5	1.9	68	2050	1458	40	99	6

```
> ## データ行列のプロット（各群を色分け）
> di0 <- list(x=xx,m=t(kb)%*%xx/gb)   # データ行列と各群の平均
> myplot.discr(di0,gb,choi=1:2) # 変量1,2のプロット
> myplot.discr(di0,gb,choi=3:4) # 変量3,4のプロット
```



```
> ## 正準判別
> di <- mydiscr(xx,gb)
> myplot.discr(di,gb)
> myplot.discr(di,gb,choi=3:2)
```



正準判別分析

$X=$ データ行列を中心化，　$Y=[h_1,\dots,h_q]$ を中心化

$\boxed{X \text{ と } Y \text{ の正準相関分析を行う}}$

$\Longrightarrow$ 正準変量 $f_1,\dots,f_q$，$g_1,\dots,g_q$，正準相関 $d_1,\dots,d_q$

1. 正準変量 f_1, f_2 等を用い要素をプロット
2. 各群における正準変量の平均をプロット
   正準変量 f_j の第 i 群における平均値は

$$m_{ij} = \frac{1}{n_i} h_i' f_j$$

注意：実際には各正準変量を「群内分散」を用いて標準化することが多い．


```
## 正準判別分析
mydiscr <- function(x,grp,std=T) {
  g <- mydiagk(grp)
  cc <- mycancor(x,g[,-1,drop=F])
  xm <- (t(g) %*% cc$xcan)/grp # 各群の平均
  ## 各成分の群内分散が１になるように標準化する
#  gv <- sqrt(apply((cc$xcan - g %*% xm)^2,2,sum)/(sum(grp)-length(grp)))
  if(std)
    gv <- sqrt( (sum(grp)-1)/(sum(grp)-length(grp)) * (1-cc$cor^2) )
  else gv <- 1
  ax <-  sweep(cc$xcoef,2,gv,"/") # 係数
  xx <- sweep(cc$xcan,2,gv,"/") # 正準変量
  centers <- sweep(xm,2,gv,"/") # 各群の中心
  list(cor=cc$cor, vars=ax, x=xx, m=centers, cc=cc, mx=apply(x,2,mean))
}
```

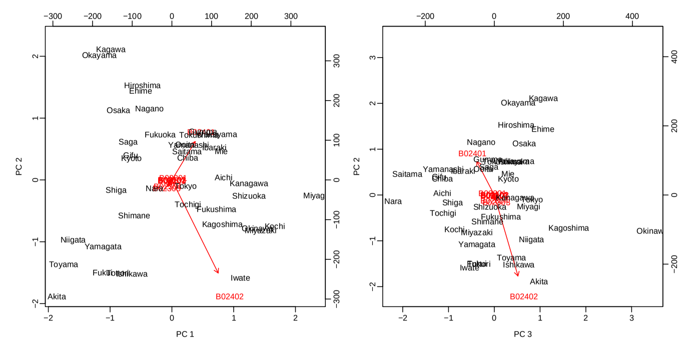


```
> di <- mydiscr(xx,gb)
> di$cor # 正準相関係数
     PC 1      PC 2      PC 3
0.8605819 0.7277284 0.5710784
> mydiscr.diag(di)$pv # 検定
[1] 4.214857e-08 8.797471e-04 5.841204e-02
> di$var # 係数

                PC 1         PC 2         PC 3
  B02101 -1.781407938  1.731185921  2.339958359
  B02102 -0.303578081 -0.363482122 -1.012162818
  B02103  0.913776451 -0.689519639 -0.889118070
  B02201  0.110171766  0.016857873 -0.165666007
  B02401  0.006224009  0.002281211 -0.000363109
  B02402  0.003562181 -0.003440037 -0.002150354
  B02301 -0.056129189 -0.003617921 -0.056589948
  B02303 -0.078787074  0.007033102  0.041001512
  B02304 -0.014169310  0.023639623  0.034229824
```



```
> di$x # 正準変量

                PC 1        PC 2       PC 3
  Ibaraki    1.2956145  0.74041083 -0.8033852
  Okinawa    2.6142846 -1.09321791  4.0479144
  Miyazaki   2.7095853 -1.14033615 -0.4498118
  Iwate      2.0937056 -2.21551210 -0.6353942
  Tokyo      0.4210998 -0.13408794  0.9602375
... 省略 ...
  Ehime     -0.9600538  2.02482041  1.2419293
  Ooita      0.4032450  0.80547306 -0.2828877
> di$m

              PC 1       PC 2       PC 3
  taihei   1.8763795 -0.2520081  0.2414654
  nihon   -1.9145290 -1.1018804  0.2187769
  nairiku -0.2230385  0.4731919 -1.3026539
  setonai -1.1963242  2.0612132  0.6918731
```

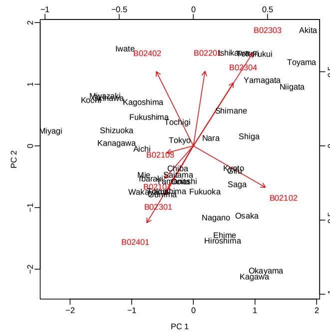
```
> mybiplot(di$cc$xcan,di$cc$xcov) # バイプロットはこのままだと見づらい
> mybiplot(di$cc$xcan,di$cc$xcov,choi=3:2)
```

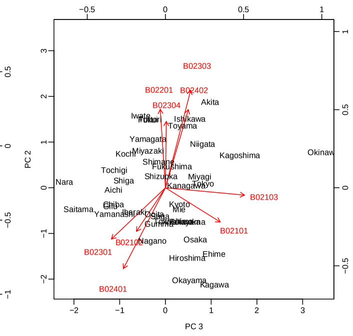


```
> di2 <- mydiscr(scale(xx),gb) # 標準化しても結果は本質的に同じ
> di2$cor
     PC 1      PC 2      PC 3
0.8605819 0.7277284 0.5710784
> di2$var

              PC 1        PC 2        PC 3
  B02101  3.6979798 -3.59372519  4.85746054
  B02102  0.3828444  0.45838976 -1.27644537
  B02103 -2.7327036  2.06205009 -2.65896125
  B02201 -0.4272654 -0.06537779 -0.64248182
  B02401 -1.1650511 -0.42701214 -0.06796914
  B02402 -1.6682836  1.61107993 -1.00707985
  B02301  0.7739804  0.04988849 -0.78033394
  B02303  2.0926470 -0.18680474  1.08903259
  B02304  0.4277421 -0.71363124  1.03332744
> di2$x
```

```
                 PC 1        PC 2       PC 3
  Ibaraki   -1.2956145 -0.74041083 -0.8033852
  Okinawa   -2.6142846  1.09321791  4.0479144
  Miyazaki  -2.7095853  1.14033615 -0.4498118
  Iwate     -2.0937056  2.21551210 -0.6353942
  Tokyo     -0.4210998  0.13408794  0.9602375
  Aichi     -1.5763250 -0.06036161 -1.3298168
  Wakayama  -1.3695475 -1.04209407  0.5031283
... 省略 ...
  Yamanashi -0.5093253 -0.80049554 -1.3078582
  Osaka      1.6362947 -1.58524689  0.7655985
  Okayama    2.2202595 -2.83312844  0.6132078
  Hiroshima  0.8982369 -2.14956461  0.5579766
  Kagawa     1.8663455 -2.96904560  1.2554139
  Ehime      0.9600538 -2.02482041  1.2419293
  Ooita     -0.4032450 -0.80547306 -0.2828877
```

```
> mybiplot(di2$cc$xcan,di2$cc$xcov) # バイプロットは標準化したほうが良い
> mybiplot(di2$cc$xcan,di2$cc$xcov,choi=3:2)
```





分類未知データへの適用

$n = n_0 + \cdots + n_q$ 個の要素を $q+1$ 群に分類したデータ行列 X に加えて分類未知の n_{q+1} 個の要素を並べたデータ行列 $\tilde{X}$ を考える．

$$\begin{bmatrix} X \\ \tilde{X} \end{bmatrix} = \begin{bmatrix} x^{(1)} \\ \vdots \\ x^{(n+n_{q+1})} \end{bmatrix}$$

X だけをつかって平均ベクトルを求め，それを $X, \tilde{X}$ から引く．

$$X \leftarrow X - \tfrac{1}{n} 1_n 1_n' X, \quad \tilde{X} \leftarrow \tilde{X} - \tfrac{1}{n} 1_{n_{q+1}} 1_n' X,$$

X の正準判別分析から求めた係数行列 A を使い，$\tilde{X}$ の正準変量を求める．

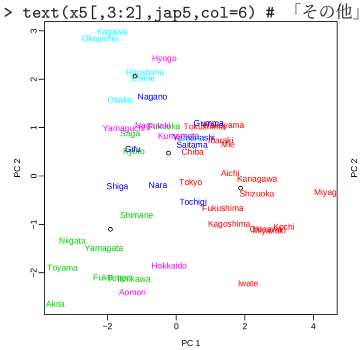
$$F = XA, \quad \tilde{F} = \tilde{X}A$$

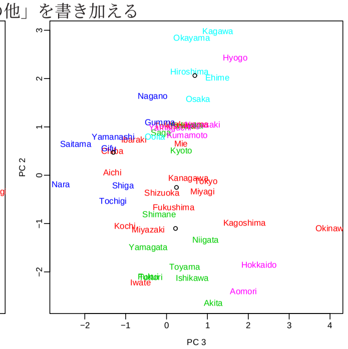
```
> ## 未知のデータへの適用
> xx5 <- X2000$x[jap5,ia]
> xx5
```



```
          B02101 B02102 B02103 B02201 B02401 B02402 B02301 B02303 B02304
Hokkaido     9.0   28.3   -7.4     74   1582   1445      8    160    145
Aomori      10.7   29.3   -3.6     75   1503   1406      3    177    119
Hyogo       17.0   33.5    1.8     64   1980   1027     21     79     22
Yamaguchi   15.7   32.8   -0.3     71   1981   1388     35    118     22
Nagasaki    17.3   32.1    2.6     69   1990   1561     41    111     10
Kumamoto    17.1   33.5    0.6     69   2076   1826     30    104     10
> xx5c <- sweep(xx5,2,apply(xx,2,mean)) # xx の平均を引く
> x5 <- xx5c %*% di$vars # 正準変量への変換
> x5

                PC 1       PC 2       PC 3
  Hokkaido  -0.20239151 -1.8496582 2.25912867
  Aomori    -1.27279482 -2.3964731 1.88312984
  Hyogo     -0.34417279  2.4190324 1.67959321
  Yamaguchi -1.52989928  0.9729923 0.08380598
  Nagasaki  -0.68099971  1.0347943 0.87655554
  Kumamoto   0.07089291  0.8338611 0.50417971
```

```
> myplot.discr(di,gb) # 4群
> text(x5[,1:2],jap5,col=6) # 「その他」(Hokkaido, Aomori など) を書き加える
> myplot.discr(di,gb,choi=3:2) # 4群
> text(x5[,3:2],jap5,col=6) # 「その他」を書き加える
```





どの群に近いかの判別

正準変量 f_j の第 i 群における平均値は

$$m_{ij} = \frac{1}{n_i} h_i' f_j$$

分類未知群の正準変量は

$$\tilde{f}_j = (\tilde{f}_{1j}, \ldots, \tilde{f}_{n_{q+1}j})'$$

第 k 要素から第 i 群の中心までの２乗距離（最初の r 個の正準変量を使う）

$$\mathrm{dis}(k,i) = \sum_{j=1}^{r} (\tilde{f}_{kj} - m_{ij})^2$$

各要素 $k = 1, \ldots, n_{q+1}$ について $\mathrm{dis}(k,i)$ を最小にする群 i を選択する．

ただし $r = q$ とするか，または $1 \le r \le q$ の範囲で選ぶ．


```
## 正準判別による分類
mydiscr.diag <- function(di,xx=NULL,nc=0,mx=T) {
  ans <- mycancor.diag(di$cc)
  if(is.null(xx)) return(ans)
  if(mx) xx <- sweep(xx,2,di$mx)
  xcan <- xx %*% di$vars
  if(nc<1) nc <- ncol(di$m)
  m0 <- di$m[,1:nc,drop=F]
  x0 <- xcan[,1:nc,drop=F]
  m2 <- apply(m0,1,function(x)sum(x*x)) # ||m||^2 のベクトル
  x2 <- apply(x0,1,function(x)sum(x*x)) # ||x||^2 のベクトル
  vv <- - t(m0 %*% t(x0)*2-m2) + x2
  dc <- t(apply(vv,1,order))
  ans$vv <- vv
  ans$dc <- dc
  ans
}
```



```
> ## どの群に近いかの判別
> dd5 <- mydiscr.diag(di,xx5) # 「その他」の群の判別
> dd5
$xsq（検定のカイ二乗統計量）
[1] 86.18981 39.63589 13.61780

$deg（自由度）
[1] 27 16  7

$pv（確率値: p<0.05 となるのは最初の２個）
[1] 4.214857e-08 8.797471e-04 5.841204e-02

$vv（各群平均への距離の二乗）

              taihei     nihon   nairiku    setonai
  Hokkaido 10.944740  7.653622 18.082354 18.739108
  Aomori   17.211091  4.857864 19.486184 21.295908
  Hyogo    14.133521 16.996830 12.694766  1.829788
```

```
  Yamaguchi 13.128218  4.471254  3.879957  1.665243
  Nagasaki   8.599388  6.519646  5.274079  1.353203
  Kumamoto   4.507913  7.770450  3.481125  3.147461

$dc (１列めが最も近い群の番号:
               1=taihei     2=nihon   3=nairiku   4=setonai)

            [,1] [,2] [,3] [,4]
  Hokkaido     2    1    3    4
  Aomori       2    1    3    4
  Hyogo        4    3    1    2
  Yamaguchi    4    3    2    1
  Nagasaki     4    3    2    1
  Kumamoto     4    3    1    2
北海道，青森が日本海に分類
兵庫，山口，長崎，熊本が瀬戸内に分類
> names(gb)[dd5$dc[,1]]
[1] "nihon"   "nihon"   "setonai" "setonai" "setonai" "setonai"
> dd <- mydiscr.diag(di,xx) # もとの４群もやってみる
```

```
> dd
$xsq
[1] 86.18981 39.63589 13.61780

$deg
[1] 27 16  7

$pv
[1] 4.214857e-08 8.797471e-04 5.841204e-02

$vv

                taihei      nihon    nairiku    setonai
  Ibaraki    2.4138963 14.7438733  2.6269819 10.1900746
  Okinawa   15.7411911 35.1725217 39.1326221 35.7341883
  Miyazaki   1.9612227 21.8309229 11.9310946 26.8094917
  Iwate      4.6714614 18.0357285 13.0416676 30.8763138
  Tokyo      2.6483777  6.9415479  5.9043804  7.5074268
... 省略 ...
```

```
  Osaka     15.9891019  7.5970812  7.5116274  0.4255534
  Okayama   26.4387111 15.7333413 13.2287183  1.6504848
  Hiroshima 13.5662270 11.7198007  6.7280641  0.1145903
  Kagawa    25.4112691 17.6493763 15.4734542  1.5906664
  Ehime     14.2302302 11.7341217  9.4256463  0.3597100
  Ooita      3.5633380  9.2617408  1.5425648  5.0856634

$dc

            [,1] [,2] [,3] [,4]
  Ibaraki      1    3    4    2
  Okinawa      1    2    4    3
  Miyazaki     1    3    2    4
  Iwate        1    3    2    4
  Tokyo        1    3    2    4
  Aichi        1    3    4    2
  Wakayama     1    3    4    2
  Kochi        1    3    2    4
  Shizuoka     1    3    2    4
```

```
  Kanagawa     1    3    4    2
  Kagoshima    1    2    3    4
  Chiba        3    1    4    2
  Miyagi       1    3    4    2
  Mie          1    3    4    2
  Fukushima    1    3    2    4
  Tokushima    1    3    4    2
  Fukui        2    3    4    1
  Akita        2    3    4    1
  Tottori      2    3    1    4
  Toyama       2    3    4    1
  Shimane      2    3    4    1
  Fukuoka      4    3    1    2
  Niigata      2    4    3    1
  Ishikawa     2    3    1    4
  Yamagata     2    3    4    1
  Kyoto        4    2    3    1
  Saga         4    3    2    1
  Saitama      3    1    4    2
```

```
  Gifu         3    2    4    1
  Nagano       4    3    2    1
  Shiga        2    3    4    1
  Gumma        1    3    4    2
  Tochigi      3    1    2    4
  Nara         3    2    1    4
  Yamanashi    3    1    4    2
  Osaka        4    3    2    1
  Okayama      4    3    2    1
  Hiroshima    4    3    2    1
  Kagawa       4    3    2    1
  Ehime        4    3    2    1
  Ooita        3    1    4    2

> sapply(1:4,function(i) sum(dd$dc[as.logical(kb[,i]),1] != i) ) # 各群
 での誤判別の個数
[1] 1 3 3 1
```

正準変量の数を変えてみる

```
> mydiscr.diag(di,xx5,3)$dc # ３個使って「その他」の群の判別

            [,1] [,2] [,3] [,4]
  Hokkaido     2    1    3    4
  Aomori       2    1    3    4
  Hyogo        4    3    1    2
  Yamaguchi    4    3    2    1
  Nagasaki     4    3    2    1
  Kumamoto     4    3    1    2

太平洋：なし
日本海：北海道，青森
内陸：なし
瀬戸内：兵庫，山口，長崎，熊本

> sapply(1:4,function(i) sum(
   mydiscr.diag(di,xx,3)$dc[as.logical(kb[,i]),1] != i)) # ３個使って誤判別
[1] 1 3 3 1
```

```
> mydiscr.diag(di,xx5,2)$dc # ２個使って「その他」の群の判別

            [,1] [,2] [,3] [,4]
  Hokkaido     2    3    1    4
  Aomori       2    3    1    4
  Hyogo        4    3    1    2
  Yamaguchi    4    3    2    1
  Nagasaki     3    4    2    1
  Kumamoto     3    4    1    2

太平洋：なし
日本海：北海道，青森
内陸：長崎，熊本
瀬戸内：兵庫，山口

> sapply(1:4,function(i) sum(
  mydiscr.diag(di,xx,2)$dc[as.logical(kb[,i]),1] != i) ) # ２個使って誤判別
[1] 3 3 2 1
```

```
> mydiscr.diag(di,xx5,1)$dc # １個使って「その他」の群の判別

            [,1] [,2] [,3] [,4]
  Hokkaido     3    4    2    1
  Aomori       4    2    3    1
  Hyogo        3    4    2    1
  Yamaguchi    4    2    3    1
  Nagasaki     3    4    2    1
  Kumamoto     3    4    1    2

太平洋：なし
日本海：なし
内陸：北海道，兵庫，長崎，熊本
瀬戸内：青森，山口

> sapply(1:4,function(i) sum(
  mydiscr.diag(di,xx,1)$dc[as.logical(kb[,i]),1] != i) ) # １個使って誤判別
[1] 3 5 3 4
```

正準相関分析と正準判別分析の関係

全変動＝群内変動＋群間変動

任意の合成変量$\boldsymbol{f} = X\boldsymbol{a}$を考える．$\boldsymbol{f} = [f_1, \ldots, f_n]', \boldsymbol{a} = [a_1, \ldots, a_p]'$．
中心化済み$1_n'X = 0$．群iにおける変量$\boldsymbol{f}$の平均は$(i = 0, \ldots, q)$

$$m_i = \frac{1}{n_i}\sum_{j \in J_i} f_j = \frac{1}{n_i}\boldsymbol{h}_i'\boldsymbol{f}$$

全変動 $$S_T = \sum_{j=1}^{n} f_j^2$$

群内変動 $$S_W = \sum_{i=0}^{q} n_i \left(\frac{1}{n_i}\sum_{j \in J_i}(f_j - m_i)^2\right)$$

群間変動 $$S_B = \sum_{i=0}^{q} n_i m_i^2$$

$$\boxed{S_T = S_W + S_B}$$

正準判別分析：全変動S_T一定の条件下でS_Bの最大化

相関の最大化

$$\boldsymbol{g} = \boldsymbol{H}\boldsymbol{c}, \quad \boldsymbol{H} = [\boldsymbol{h}_0, \ldots, \boldsymbol{h}_q], \quad \boldsymbol{c} = [c_0, \ldots, c_q]'$$

$\sigma_{fg} = \frac{1}{n-1}\boldsymbol{f}'\boldsymbol{g}$を最大にする$\boldsymbol{c}$を求める．ただし$1_n'\boldsymbol{g} = 0$，$\frac{1}{n-1}\boldsymbol{g}'\boldsymbol{g} = 1$．

$$1_n'\boldsymbol{g} = 1_n'\boldsymbol{H}\boldsymbol{c} = n_0c_0 + \cdots + n_qc_q$$
$$\boldsymbol{g}'\boldsymbol{g} = \boldsymbol{c}'\boldsymbol{H}'\boldsymbol{H}\boldsymbol{c} = n_0c_0^2 + \cdots + n_qc_q^2$$
$$\boldsymbol{f}'\boldsymbol{g} = \boldsymbol{f}'\boldsymbol{H}\boldsymbol{c} = n_0c_0m_0 + \cdots + n_qc_qm_q$$

ラグランジュの未定乗数法

$$\psi(\boldsymbol{c}, \lambda_1, \lambda_2) = \boldsymbol{f}'\boldsymbol{g} - \lambda_1 1_n'\boldsymbol{g} - \tfrac{1}{2}\lambda_2(\boldsymbol{g}'\boldsymbol{g} - (n-1))$$

$$\frac{\partial\psi}{\partial c_i} = n_i(m_i - \lambda_1 - \lambda_2 c_i) \text{ より } c_i = \frac{m_i - \lambda_1}{\lambda_2}$$

$1_n'\boldsymbol{g} = -\lambda_1/\lambda_2$より$\lambda_1 = 0$．$\boldsymbol{g}'\boldsymbol{g} = S_B/\lambda_2^2$より$\lambda_2 = \sqrt{S_B/(n-1)}$．
従って$c_i = m_i\sqrt{(n-1)/S_B}$のとき$\sigma_{fg}$は最大値$\sqrt{S_B/(n-1)}$を取る．

群間変動の最大化

条件$1_n'\boldsymbol{g} = 0$，$\frac{1}{n-1}\boldsymbol{g}'\boldsymbol{g} = 1$下で$\boldsymbol{g} = \boldsymbol{H}\boldsymbol{c}$を動かすと，任意の$\boldsymbol{f}$に対して$\sigma_{fg}$の最大値は

$$\sigma_{fg} = \sqrt{\tfrac{1}{n-1}S_B}$$

今度は条件$\frac{1}{n-1}\boldsymbol{f}'\boldsymbol{f} = 1$下で上記の$\sigma_{fg}$を最大化する．すなわち全変動$S_T$一定の条件下で$S_B$の最大化を考える．正準判別分析は本来このように定義される．

これは与えた条件下で$\boldsymbol{f} = X\boldsymbol{a}$と$\boldsymbol{g} = Y\boldsymbol{b}$を同時に動かして相関$r_{fg}$を最大化に等しく，正準相関分析になる．

群内分散による正準変量の標準化

正準相関分析の結果得られる正準変量$\boldsymbol{f}_1, \ldots, \boldsymbol{f}_q$は全分散が１に標準化されている．正準判別分析では通常群内分散が１に標準化する．

$$\boldsymbol{f}_j \leftarrow \boldsymbol{f}_j\sqrt{\frac{n-(q+1)}{(n-1)(1-d_j^2)}}$$

正準変量の一つを$\boldsymbol{f}$とかくと，

$$\text{全分散} = \tfrac{1}{n-1}S_T = \tfrac{1}{n-1}\boldsymbol{f}'\boldsymbol{f} = 1$$

$$\text{群内分散} = \tfrac{1}{n-(q+1)}S_W = \frac{n-1}{n-(q+1)}\left(\frac{S_T}{n-1} - \frac{S_B}{n-1}\right)$$

ただし正準相関をdと書くと$S_B/(n-1) = d^2$である．従って

$$\boldsymbol{f}_j\text{の群内分散} = \frac{n-1}{n-(q+1)}(1-d_j^2)$$

演習問題

6-1．正準相関分析または正準判別分析の数値例を以下の手順に従って示せ．
(i) データ行列Xに用いる変量を数個〜十数個の範囲で選ぶ．
(ii-a) 正準相関分析の場合はデータ行列Yに用いる変量を同様に選ぶ．
(ii-b) 正準判別分析の場合は判別群（県名の分類）を決める．
(iii) `mycancor`または`mydiscr`を用いて分析を実行する．
(iv) 結果のバイプロットなどを図示する．係数を眺める．
(v) 面白い解釈が可能ならばそれを述べる．