Statistics 5401
30. Fisher Discrimination

Gary W. Oehlert

School of Statistics

313B Ford Hall

612-625-1557

gary@stat.umn.edu

Classification is a method that will predict the group membership of future data based on covariates x. Discrimination is more exploratory. It tries to transform the covariates x in such a way that the various groups will be well-separated on the transformed scale.

Fisher discrimination finds one or more linear combinations of x to best separate the groups.

For two groups, consider computing $y = a'x$. We want to separate the two groups, so why don't we try to find the a that makes the t-test between the two groups as big as possible?

$$t = \frac{\overline{y}_1 - \overline{y}_2}{s_y\sqrt{\frac{1}{n_1} + \frac{1}{n_2}}}$$

where s_y is the pooled estimate of variance. This t is maximized whenever

$$\delta^2 = \frac{(\overline{y}_1 - \overline{y}_2)^2}{s_y^2}$$

is maximized.

Let $d = \overline{x}_1 - \overline{x}_2$. It's not hard to see that

$$\delta^2 = \frac{(a'd)^2}{a'\mathbf{S}a}$$

where $\mathbf{S}$ is the pooled estimate of variance for the (multivariate) xs. To maximize δ^2, we choose

$$a = \mathbf{S}^{-1}d = \mathbf{S}^{-1}(\overline{x}_1 - \overline{x}_2)$$

(2-50 of the text). The maximum value is

$$D^2 = (\overline{x}_1 - \overline{x}_2)'\mathbf{S}^{-1}(\overline{x}_1 - \overline{x}_2)$$

You can use

$$y = a'x = (\overline{x}_1 - \overline{x}_2)'\mathbf{S}^{-1}x$$

to classify. Classify to π_1 if

$$y \geq (\overline{x}_1 - \overline{x}_2)'\mathbf{S}^{-1}(\overline{x}_1 + \overline{x}_2)/2$$

otherwise, classify into π_2.

This is just minimum cost classification for multivariate normals with equal variances when

$$\frac{p_2}{p_1}\frac{c(1|2)}{c(2|1)} = 1$$

This method can be extended to more than two populations, and more than two linear combinations used to discriminate.

We don't need to assume normality, but we do need to assume that all populations have the same variance.
Using the sample data, perform a MANOVA of x by group. Let $\mathbf{B}$ be the between groups sum of squares matrix, and let $\mathbf{W}$ be the within groups (error) sum of squares matrix.
$\mathbf{B}$ is more or less the variance among group means (needs some scaling and tweaking due to sample sizes), so $a'\mathbf{B}a$ is more or less the variance among the group means of the $a'x$'s.
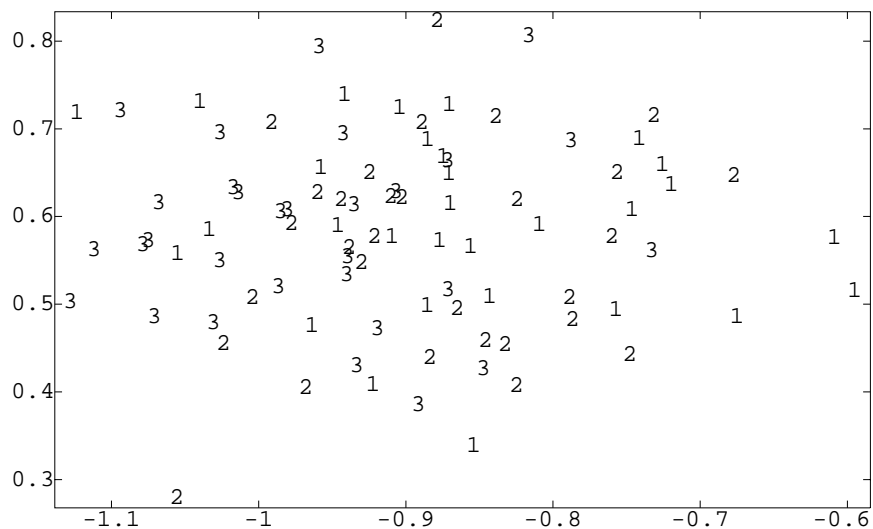Analogously, $a'\mathbf{W}a$ is proportional to the variance of $a'x$.
Thus it makes sense to try to find a to maximize

$$\frac{a'\mathbf{B}a}{a'\mathbf{W}a}$$

We've seen this problem before! The optimal a is the first eigenvector of $\mathbf{B}$ relative to $\mathbf{W}$.
The linear discriminants are the linear combinations of the data formed by the eigenvectors of $\mathbf{B}$ relative to $\mathbf{W}$.

```
Cmd> X <- matrix(vecread(""),5)'
Read from file "~/JW5data/T6-13.DAT"

Cmd> gps <- factor(X[,5])

Cmd> X <- X[,-5]

Cmd> manova("X=gps",silent:T)

Cmd> B <- matrix(SS[2,,])

Cmd> W <- matrix(SS[3,,])

Cmd> releigen(B,W)
component: values
(1)   0.18697   0.014913   0      0
component: vectors
           (1)         (2)            (3)              (4)
(1)   -0.01731   -0.01243   -0.00095319      0.011956
(2)-0.0031664   0.017222     -0.011639      0.0096574
(3)   0.015997 -0.007458     -0.0013435      0.012517
(4)   0.004921   0.013185      0.032188      0.0067102

Cmd> A <- releigen(B,W)$vectors[,run(2)]

Cmd> Y <- X %*% A

Cmd> chplot(Y[,1],Y[,2],gps)
```

```
0.8                          2        3
                      3                 
0.7  1   3       1        1    1 1    2        2
             3      2      3   2  1        3      1
                       1    2      1 3                2
              3     3  2 23  22   1    2         2  1  1   2
0.6       3    3 3       2   1         1       1
     3    3  1   3      1    2 1   1  1        2    3        1
                   2         2 21                            
0.5 3                 2  3       1 3  1     2   1              1
         3        3         1        1 2    2   2              
            2              3        2  2       1
                 3    1      2  3  3    2
0.4                2        1          2
                    3           
                          1
0.3
       2
    -1.1        -1        -0.9       -0.8       -0.7       -0.6
```

Cmd> readdata("",x1,x2,x3,x4,x5,site)
Read from file "~/JW5data/T11-7.DAT"

Cmd> X <- hconcat(x1,x2,x3,x4,x5)

Cmd> manova("X=site",silent:T)

Cmd> B <- matrix(SS[2,,])

Cmd> W <- matrix(SS[3,,])

Cmd> releigen(B,W)
component: values
(1)   4.18   0.666   0   0   0
component: vectors
          (1)   (2)   (3)   (4)   (5)
(1) -0.0452   -0.0171   0.0245   0.0175   0.0588
(2) 0.00836   0.00561   0.0129   0.00172 -0.00071
(3)   -0.335   0.27   0.0243   0.323   -0.206
(4)   0.0634   -0.13   -0.0368   0.0604   0.0629
(5)   0.0272   0.0554   -0.0158   0.00108   0.00876

Cmd> A <- releigen(B,W)$vectors[,run(2)]

Cmd> Y <- X %*% A

Cmd> chplot(Y[,1],Y[,2],site)

3