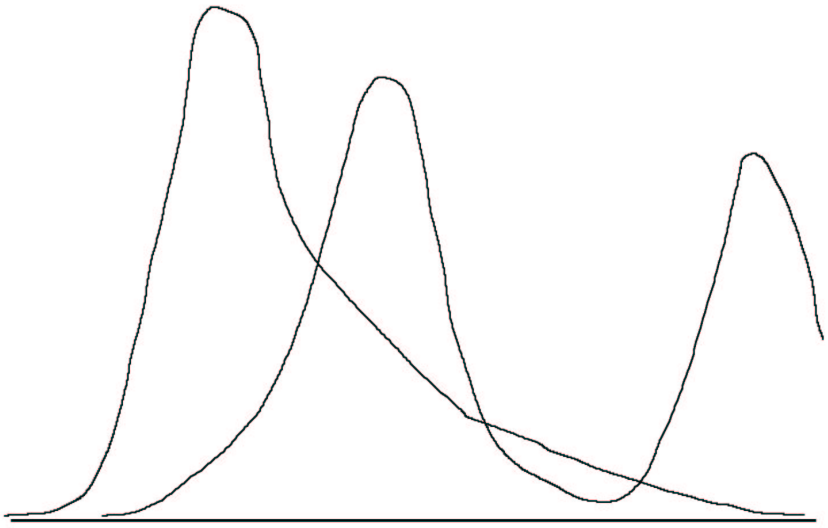
Statistics 5401
27. Density-Based Classification

Gary W. Oehlert
School of Statistics
313B Ford Hall
612-625-1557
gary@stat.umn.edu

Classification and *discrimination* are two closely related topics in statistics.
Classification is used for procedures that allocate objects into two or more well defined groups.
Discrimination is used in a more exploratory way for features or aspects of data that separate groups, and thus may be used as a basis for classification.
I tend to use classification and discrimination more or less interchangeably.
These kinds of ideas and methods have arisen in many contexts and in many areas of research; consequently, they also go by other names. For example, in computer science, classification methods are called *supervised learning*.
Example. Spino-cerebullar ataxias are genetic diseases. There are at least eight kinds of SCAs, distinguished by eight different mutations. There are over 30 different symptoms that have been associated with the different SCAs.
Suppose you have a patient and wish to classify him or her into SCA type. You can do an expensive DNA analysis, or you can try to classify based on observed symptoms.
Using a set of over 100 patients for whom we have symptoms AND genetic information, develop a procedure to classify new patients into type based on symptoms.
Down to specifics. Let x be a p-vector of variables or attributes. Each object (unit, case, subject) has its p attributes, and it also has a group or type.
For the SCA data, the set of symptoms for a subject is the x vector, and the type of mutation is the group.
There are two populations: π_1 and π_2. Let $f_1()$ be the density of x when we sample from π_1. Let $f_2()$ be the density when we sample from π_2.
Here are two densities:



The basic idea is that if we see an x in a region where $f_1()$ is high, then we might want to classify this observation into π_1.
If we see an observation in a region where $f_2()$ is high, then we might want to classify the observation into π_2.
However, this is not quite enough, because two more factors may influence us.
The first factor is prevalence. Prevalence is the prior probability of each population, or alternatively, the fraction of the total population belonging to population 1 or population 2.
For two groups, we have prevalences p_1 and p_2, with $p_1 + p_2 = 1$.
The issue with prevalence is that if we know, for example, that there are 10 times as many units in population 1 as population 2, then we should be more likely to classify a unit as population 1, even if the density $f_1()$ is really high.
The second factor is misclassification cost.
Let $c(1|2)$ be the cost of classifying a population 2 object as a population 1 object, and let $c(2|1)$ be the cost of classifying a population 1 object as a population 2 object.
If $c(1|2) > c(2|1)$, then we'll tend to classify more objects into population 2, because we'll tend to reduce our misclassification cost.
For example, there could be two diseases that present similar symptoms. One disease is not serious, but the second disease can have life threatening complications.
We will tend to treat this group of symptoms as the second (serious) disease, because the cost of not supplying the appropriate therapy is so high.
So, lets put the pieces together. Classify x into population 1 if

$$\frac{f_1(x)}{f_2(x)} > \frac{c(1|2)}{c(2|1)} \frac{p_2}{p_1}$$

otherwise classify into population 2. The region where we classify into π_1 is R_1, whereas the region where we classify into π_2 is R_2.
This is the minimum expected cost classification rule. It minimizes the expected misclassification cost.
We assign to the population with the higher density, but we also weight by prevalence and misclassification cost.
If the misclassification costs are equal, we assign to π_1 if

$$\frac{f_1(x)}{f_2(x)} > \frac{p_2}{p_1}$$

If the prevalences are equal, we assign to π_1 if

$$\frac{f_1(x)}{f_2(x)} > \frac{c(1|2)}{c(2|1)}$$

If the both are equal, we assign to π_1 if

$$\frac{f_1(x)}{f_2(x)} > 1$$

When costs or prevalences are unknown, they are often taken to be equal. This is not perfect, but you have to do something.
With very large data sets, we can estimate $f_1()$ and $f_2()$ without making distributional assumptions.
For most data sets, we need to make distributional assumptions. For example, we might assume that both populations follow multivariate normal distributions.

What's really happening? If p_1 and p_2 are the prior probabilities of the populations, and $f_1()$ and $f_2()$ describe the density of the data for a given population, then the probability that x comes from π_1 given is

$$\begin{aligned} P(\pi_1|x) &= \frac{P(x \text{ and } \pi_1)}{P(x)} \\ &= \frac{P(x|\pi_1)P(\pi_1)}{P(x|\pi_1)P(\pi_1) + P(x|\pi_2)P(\pi_2)} \\ &= \frac{f_1(x)p_1}{f_1(x)p_1 + f_2(x)p_2} \end{aligned}$$

Similarly,

$$P(\pi_2|x) = \frac{f_2(x)p_2}{f_1(x)p_1 + f_2(x)p_2}$$

$P(\pi_1|x)$ is the posterior probability of π_1; that is, the probability after seeing the data: among all the potential data from all the populations that have covariates x, what fraction is from π_1.
If we classify this x into π_2, we'll incur the cost $c(2|1)$ for all the π_1 data with that x, but we'll have no cost for the data from π_2 with that x.
Thus the expected cost for classifying into π_2 is

$$EC_2 = c(2|1)P(\pi_1|x) = c(2|1)\frac{f_1(x)p_1}{f_1(x)p_1 + f_2(x)p_2}$$

Go through the same exercise to get the expected cost for classifying into π_1 is

$$EC_1 = c(1|2)P(\pi_2|x) = c(1|2)\frac{f_2(x)p_2}{f_1(x)p_1 + f_2(x)p_2}$$

The ratio of these costs is

$$\frac{EC_2}{EC_1} = \frac{c(2|1)f_1(x)p_1}{c(1|2)f_2(x)p_2}$$

We classify into π_1 when this ratio is greater than 1, that is, when the cost of classifying into π_2 is greater. This is the least cost rule from above.
Classification with normal distributions.
It turns out that you get very different classification rules when $\Sigma_1 = \Sigma_2$ and when $\Sigma_1 \neq \Sigma_2$.
Let's begin with the equal variance case. Then

$$\begin{aligned} &\log\left(\frac{f_1(x)}{f_2(x)}\right) \\ &= \frac{1}{2}\left((x-\mu_2)'\Sigma^{-1}(x-\mu_2) - (x-\mu_1)'\Sigma^{-1}(x-\mu_1)\right) \\ &= (\mu_1-\mu_2)'\Sigma^{-1}x - (\mu_1-\mu_2)'\Sigma^{-1}(\mu_1-\mu_2)/2 \end{aligned}$$

So we classify into π_1 if

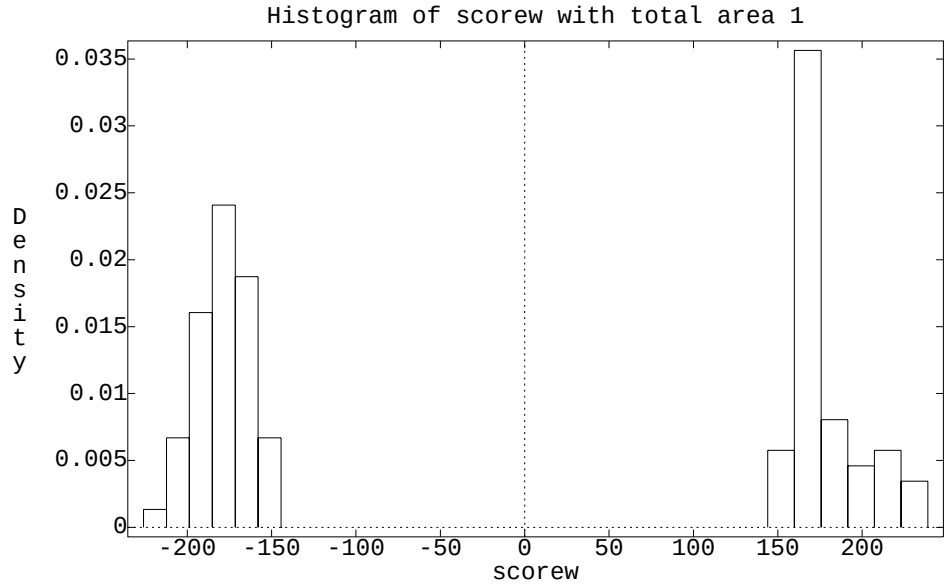
$$\begin{aligned} &(\mu_1-\mu_2)'\Sigma^{-1}x - (\mu_1-\mu_2)'\Sigma^{-1}(\mu_1-\mu_2)/2 \\ &> \ln\left[\frac{c(1|2)}{c(2|1)}\frac{p_2}{p_1}\right] \end{aligned}$$

Otherwise, we classify into π_2.
We almost never know μ_1, μ_2, or Σ, so we must use the estimated values $\overline{\mathbf{x}}_1$, $\overline{\mathbf{x}}_2$, and $\mathbf{S}$ (the pooled variance estimate). Thus, in practice, we classify into π_1 if

$$\begin{aligned}(\overline{\mathbf{x}}_1 - \overline{\mathbf{x}}_2)'\Sigma^{-1}x - (\overline{\mathbf{x}}_1 - \overline{\mathbf{x}}_2)'\Sigma^{-1}(\overline{\mathbf{x}}_1 - \overline{\mathbf{x}}_2)/2 \\ > \ \ln\left[\frac{c(1|2)}{c(2|1)}\frac{p_2}{p_1}\right]\end{aligned}$$

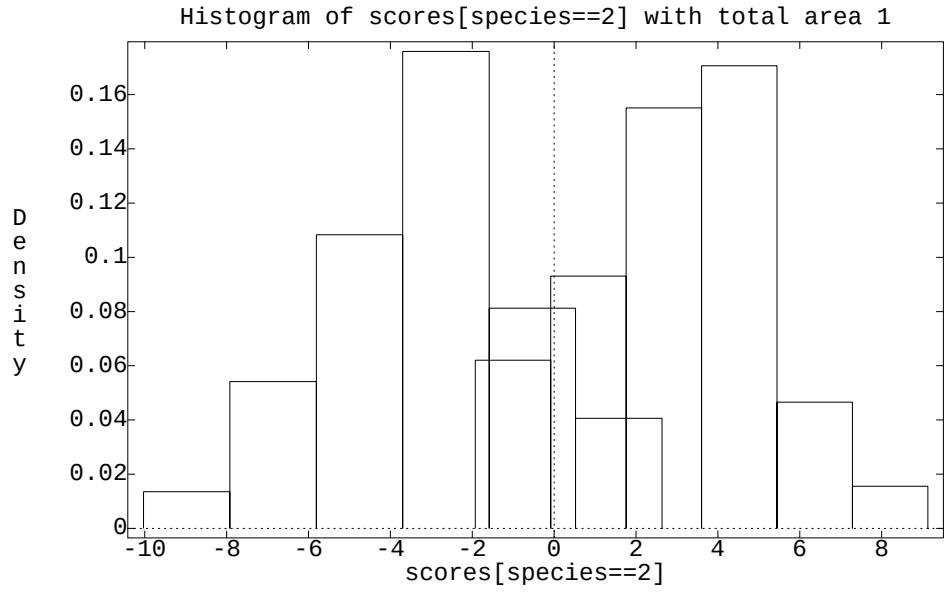
Otherwise, we classify into π_2.

```
Cmd> men <- matrix(vecread(""),8)'
Read from file "~/JW5data/T8-6.DAT"

Cmd> women <- matrix(vecread(""),7)'
Read from file "~/JW5data/T1-9.DAT"

Cmd> men <- men[,-7]

Cmd> xbar1 <- tabs(men,mean:T)

Cmd> xbar2 <- tabs(women,mean:T)

Cmd> dim(men)
(1)          55           7

Cmd> dim(women)
(1)          55           7

Cmd> S <- (tabs(men,covar:T)+\
tabs(women,covar:T))/2

Cmd> cfs <- (xbar1-xbar2)'%*%solve(S);cfs
(1,1)   -14.234   0.43618   0.69155   -113.59
(1,5)   -109.25    54.667  -0.13036

Cmd> const <- (xbar1-xbar2)'%*%solve(S)%*%\
(xbar1+xbar2)/2;  const
(1,1)      -154.55

Cmd> scorem <- men%*%cfs' - const

Cmd> scorew <- women%*%cfs' - const

Cmd> hist(scorem);hist(scorew,add:T);\
 showplot(xmin:?,xmax:?)
```




```
Cmd> X <- matrix(vecread(""),8)'
Read from file "~/JW5data/T6-15.DAT"

Cmd> species <- X[,8]+1

Cmd> species <- factor(species)

Cmd> X <- X[,-8]

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

Cmd> S <- matrix(SS[3,,])/DF[3]

Cmd> xbar1 <- tabs(X[species==1,],mean:T)

Cmd> xbar2 <- tabs(X[species==2,],mean:T)

Cmd> cfs <- (xbar1-xbar2)'%*%solve(S);cfs
(1) 0.0062356   0.15059  -0.85382   0.26757
(5)  -0.38279   -2.1873    2.9707

Cmd> const <- (xbar1-xbar2)'%*%solve(S)%*%\
(xbar1+xbar2)/2;  const
(1,1)      -24.173

Cmd> scores <- X %*% cfs' - const

Cmd> hist(scores[species==1]);\
hist(scores[species==2],add:T);\
```

```
showplot(xmin:?,ymin:?,xmax:?,ymax:?)
```




```
Cmd> discrim(species,X)
component: coefs
        species1     species2
(1)       1.7558       1.7496
(2)    -0.022239     -0.17283
(3)       3.8413       4.6951
(4)       1.6597       1.3922
(5)     -0.18566      0.19712
(6)       5.7122       7.8995
(7)      -3.0088      -5.9795
component: addcon
        species1     species2
(1)      -174.53       -198.7

Cmd> cfs <- discrim(species,X)$coefs

Cmd> con <- discrim(species,X)$addcon

Cmd> score1 <- X %*% cfs[,1]+con[1]

Cmd> score2 <- X %*% cfs[,2]+con[2]

Cmd> cor(score1-score2,scores)
(1,1)            1            1
(2,1)            1            1

Cmd> both <- hconcat(score1,score2)'
```

```
Cmd> both <- both-max(both)

Cmd> both <- exp(both)

Cmd> pp1 <- both[1,]/sum(both)

Cmd> pp1 <- pp1’

Cmd> chplot(scores,pp1,species)
```