

Contents

- Preliminaries
- 0: Introduction
- 1: Graphical Displays
- 2: Reduction of Dimensionality
- 3: Multidimensional Scaling Techniques
- 4: Discriminant Analysis
- 5: Multivariate Regression Analysis
- 6: Canonical Correlation Analysis
- 7: Partial Least Squares
- 8: Statistical Analysis of Multivariate Data
- 9: Statistical Discriminant Analysis**

463

Statistical Discriminant Analysis

Introduction

- ◆ Setup:-
 - p-dimensional data on objects from k groups
 - c.f. crimcoords
 - data x observations of a random variable X whose density **depends** on which group the object belongs to
 - e.g. $X \sim N_p(\mu_i, \Sigma)$; $i=1, \dots, k$
- ◆ want a rule for deciding unambiguously which category an object x belongs to just on the basis of the measurements x
 - NB **unambiguously** so not 'fuzzy' rules

464

- ◆ What are the [statistical] properties of the rule?
 - how to evaluate one rule **d** in relation to another **d***
- Easiest case when distribution of X known
 - unrealistic in practice but useful to consider
 - ◆ If distribution not known then may have parametric form
 - e.g. $N_p(\mu_i, \Sigma)$
 - and can estimate parameters from available data

465

- ◆ If no parametric form then possibilities are
 - non-parametric estimate of distribution (e.g. kernel)
 - use any of techniques in § 4.6
- ◆ Still enables construction of a rule so can consider its statistical properties
- ◆ Key properties relate to 'success rates'
 - How accurate is the rule?
 - What is probability of correct classification?
 - [on new cases]
- ◆ If all densities are known then can calculate such properties exactly and so find 'the best rule' for specific criteria.
 - realistically they need estimation

466

Known densities

- ◆ Maximum Likelihood Discriminant Rule
 - allocate x to category with highest likelihood for x
- Example:- 2 univariate Normal populations
 - ◆ $p=1, k=2, x \sim N(\mu_i, \sigma_i^2)$ if in category $i, i=1,2$: allocate x to category 1 if $f_1(x) > f_2(x)$, i.e. if

$$\frac{\sigma_2}{\sigma_1} \exp \left\{ -\frac{1}{2} \left(\frac{x-\mu_1}{\sigma_1} \right)^2 + \frac{1}{2} \left(\frac{x-\mu_2}{\sigma_2} \right)^2 \right\} > 1$$
 - i.e. if [taking logs and assuming $\log(\sigma_2/\sigma_1) \neq 0$]
 - $$Q(x) = x^2 \left(\frac{1}{\sigma_1^2} - \frac{1}{\sigma_2^2} \right) - 2x \left(\frac{\mu_1}{\sigma_1^2} - \frac{\mu_2}{\sigma_2^2} \right) + \left\{ \left(\frac{\mu_1^2}{\sigma_1^2} - \frac{\mu_2^2}{\sigma_2^2} \right) - 2 \log \left(\frac{\sigma_2}{\sigma_1} \right) \right\} < 0$$

467

- Suppose $\sigma_1 > \sigma_2$ then coefficient of $x^2 < 0$ then $Q(x) < 0$ if x is small or big & so x is classified as group 1

468



◆ If $\sigma_1 = \sigma_2$ then $\log(\sigma_2/\sigma_1) = 0$ and rule becomes allocate to 1 if $|x - \mu_1| < |x - \mu_2|$
 i.e. if $\mu_1 < \mu_2$ then allocate to 1 if $x < \frac{1}{2}(\mu_1 + \mu_2)$

μ_1 μ_2

© NRJF, University of Sheffield, 2011/12, Semester 1 Multivariate Data Analysis 469

■ **p dimensions, k Normal populations**

- ◆ means μ_i , common variance Σ
 - $f_i(x) = (2\pi)^{-p/2} |\Sigma|^{-1/2} \exp\{-\frac{1}{2}(x - \mu_i)' \Sigma^{-1} (x - \mu_i)\}$
 - $f_i(x)$ maximized when $(x - \mu_i)' \Sigma^{-1} (x - \mu_i)$ minimized
- ◆ i.e. allocate x to the category whose mean has the smallest Mahalanobis distance from x

© NRJF, University of Sheffield, 2011/12, Semester 1 Multivariate Data Analysis 470

■ When $k=2$ this becomes:
 allocate x to category 1
 if $(\mu_1 - \mu_2)' \Sigma^{-1} (x - \mu) > 0$, where $\mu = \frac{1}{2}(\mu_1 + \mu_2)$

■ Maximum Likelihood Discriminant Function for two Normal populations with the **same** variances is **linear**

© NRJF, University of Sheffield, 2011/12, Semester 1 Multivariate Data Analysis 471

■ **Discrimination Under Estimation**

- ◆ Need to estimate parameters:
- ◆ 2 possibilities:
 - ◆ Sample ML Discriminant Rule
 - Estimate parameters just from 'training data'
 - Plug in estimates & proceed as if they were known
 - ◆ Likelihood Ratio Discriminant Rule
 - Estimate parameters assuming new observation is from i^{th} group ($i=1, \dots, k$)
 - Plug in estimates & proceed as if they were known
- ◆ Different for small & unequal sample sizes

© NRJF, University of Sheffield, 2011/12, Semester 1 Multivariate Data Analysis 472

■ Two Normal groups case:

- ◆ ML rule is allocate to 1
 if $(\mu_1 - \mu_2)' \Sigma^{-1} (x - \frac{1}{2}(\mu_1 + \mu_2)) > 0$
- ◆ Sample ML rule is allocate to 1 if
 $(\bar{x}_1 - \bar{x}_2)' W^{-1} \{x - \frac{1}{2}(\bar{x}_1 + \bar{x}_2)\} > 0$
- ◆ This is a special case of Fisher's **Linear Discriminant Function**

© NRJF, University of Sheffield, 2011/12, Semester 1 Multivariate Data Analysis 473

■ **Fisher's Linear Discriminant Function**

$h(x) = a'x$

where a is the first eigenvector of $W^{-1}B$

- ◆ calculate the discriminant score $a'x$ allocate to that group j where

$$\min_{i=1}^k |a'x - a'\bar{x}_i| = a'x - a'\bar{x}_j$$
 - i.e. $j = \arg(\min_{i=1}^k |a'x - a'\bar{x}_i|)$

© NRJF, University of Sheffield, 2011/12, Semester 1 Multivariate Data Analysis 474



Probabilities of Misclassification

- $p_{ij} = P[\text{a type } j \text{ object is classified as type } i]$
- the **performance** of a rule described by $\{p_{ij}\}$
- Good rules:– p_{ij} small for $i \neq j$ & big for $i=j$
 - i.e. low probabilities of *misclassification* and high probabilities of *correct classification*
- If we **know** the densities then we can calculate p_{ij} if not then we must estimate them

475

Calculating probabilities when densities known:

- Rule:– classify x as category j if $x \in R_j$
- Density of x is $f_j(\cdot)$ when x is of type j

476

$p_{ij} = P[x \in R_i \text{ when } x \text{ is of type } j]$
 $= P[x \in R_i \text{ when } x \text{ has density } f_j(\cdot)]$

- i.e. a type j object with density $f_j(\cdot)$ has landed in R_i

$$= \int_{R_i} f_j(x) dx$$

$$= \int_{R_i} \phi_i(x) f_j(x) dx$$

- where $\phi_i(x)$ is the indicator function of R_i (i.e. $\phi_i(x)=1$ if $x \in R_i$ and $\phi_i(x)=0$ if $x \notin R_i$)

$$= \int_{\mathbb{R}^p} \phi_i(x) f_j(x) dx$$

- since integrand is 0 outside R_i

477

Example: 2 regions in 1 dimension:

- If x is of type 1 then x has density $N(\mu_1, 1)$
- if x is of type 2 then it has density $N(\mu_2, 1)$
- Regions R_1 and R_2 so if $x \in R_1$ classify as 1
- Say $R_1 = \{x : x < c\}$, $R_2 = \{x : x > c\}$ for some value c

478

Example: 2 regions in 1 dimension:

- $p_{11} = P[x \in R_1 | x \text{ is of type 1}] = P[x \in R_1 | x \text{ is } N(\mu_1, 1)]$
 $= P[x < c | x \sim \phi(x - \mu_1)] = \Phi(c - \mu_1)$
- $p_{12} = P[x \in R_1 | x \text{ is of type 2}] = P[x \in R_1 | x \text{ is } N(\mu_2, 1)]$
 $= P[x < c | x \sim \phi(x - \mu_2)] = \Phi(c - \mu_2)$
- $p_{22} = P[x > c | x \sim \phi(x - \mu_2)] = 1 - \Phi(c - \mu_2)$
- $p_{21} = P[x > c | x \sim \phi(x - \mu_1)] = 1 - \Phi(c - \mu_1)$

479

- If d and d^* are two discriminant rules with $\{p_{ij}\}$ and $\{p^*_{ij}\}$ then d is **better** than d^* if $p_{ii} \geq p^*_{ii}$ for all $i=1,2,\dots,k$ and $p_{ij} > p^*_{ij}$ for at least one j , $1 \leq j \leq k$
 - i.e. probability of correct classification *never worse and sometimes better*
- If d is a discriminant rule with no better rule then d is **admissible**
- If d is better than d^* then d^* is **inadmissible**
 - NB could be that $p_{11} > p^*_{11}$ but $p_{22} < p^*_{22}$ so cannot compare d and d^*

480



- **Main Results** (if densities are known) :
 - ◆ The rule which minimizes the total misclassification probability is the ML rule
 - equivalently maximizes total correct classification probability
 - See § 9.4.2
 - ◆ All Bayes rules are admissible
 - ◆ There are arbitrarily many admissible rules
- Anticipate these properties hold when parameters are estimated
 - perhaps 'asymptotically'

© NRJF, University of Sheffield, 2011/12, Semester 1 Multivariate Data Analysis 481

- **Particular Cases**
 - ◆ Two Normal Populations $N_p(\mu_i, \Sigma)$
 - Remember classify as group 1 if $(\mu_1 - \mu_2)' \Sigma^{-1} (x - \frac{1}{2}(\mu_1 + \mu_2)) > 0$
 - Want $P[\text{Classify as 1 when really from 2}]$
 - i.e. use $N_p(\mu_2, \Sigma)$ when calculating probability
 - If $X \sim N_p(\mu_2, \Sigma)$ then $\alpha'(x - \mu) \sim N(\alpha'(\mu_2 - \mu), \alpha' \Sigma \alpha)$
 - any p-vectors μ and α
 - » Univariate Normal note since $\alpha'x$ is a scalar
 - put $\mu = \frac{1}{2}(\mu_1 + \mu_2)$ & $\alpha = (\mu_1 - \mu_2)' \Sigma^{-1}$
 - Then $h(x) = (\mu_1 - \mu_2)' \Sigma^{-1} (x - \frac{1}{2}(\mu_1 + \mu_2)) \sim N(-\frac{1}{2} \Delta^2, \Delta^2)$
 - where $\Delta^2 = (\mu_1 - \mu_2)' \Sigma^{-1} (\mu_1 - \mu_2)$

© NRJF, University of Sheffield, 2011/12, Semester 1 Multivariate Data Analysis 482

- ◆ so, want $P[h(x) > 0]$ when $h(x) \sim N(-\frac{1}{2} \Delta^2, \Delta^2)$

$$= \Phi\left(\frac{-\frac{1}{2} \Delta^2}{\sqrt{\Delta^2}}\right) = \Phi(-\frac{1}{2} \Delta)$$
- ◆ Similarly, when x is from 1 $h(x) \sim N(\frac{1}{2} \Delta^2, \Delta^2)$
 - x is classified as type 2 if $h(x) < 0$
 - so $p_{21} = P[h(x) < 0]$ when $h(x) \sim N(\frac{1}{2} \Delta^2, \Delta^2)$

$$= \Phi\left(\frac{-\frac{1}{2} \Delta^2}{\sqrt{\Delta^2}}\right) = \Phi(-\frac{1}{2} \Delta)$$
- ◆ **the misclassification probabilities are equal**

© NRJF, University of Sheffield, 2011/12, Semester 1 Multivariate Data Analysis 483

- **Estimating Misclassification Probabilities**
 - ◆ If assuming a parametric model then plug in estimates of parameters
 - ◆ e.g. in two group Normal case have

$$\hat{p}_{21} = \hat{p}_{12} = \Phi(-\frac{1}{2} \hat{\Delta})$$
 where

$$\hat{\Delta}^2 = (\bar{x}_1 - \bar{x}_2)' S^{-1} (\bar{x}_1 - \bar{x}_2)$$

© NRJF, University of Sheffield, 2011/12, Semester 1 Multivariate Data Analysis 484

- ◆ Estimate is biased downwards
 - i.e. over-optimistic, says rule performs better than it actually will on future data
 - **Reason:**
 - $h(x)$ is constructed to separate the observed data as best it can
 - Future data would not give the same $h(x)$ i.e. the calculated $h(x)$ will not separate out new data as perfectly as calculating a new $h(x)$

© NRJF, University of Sheffield, 2011/12, Semester 1 Multivariate Data Analysis 485

- ◆ If not parametric (or for any rule) then can look at a table of true category vs predicted category
 - (a confusion matrix)
 - ◆ If $n_{ij} = \#\{x \in \text{Pop}^n_j, x \in R_i\}$ then we can estimate p_{ij} by

$$\hat{p}_{ij} = \frac{n_{ij}}{\sum_i n_{ij}} = \frac{n_{ij}}{n_j}$$
 - proportion from popⁿ j misclassified as i

© NRJF, University of Sheffield, 2011/12, Semester 1 Multivariate Data Analysis 486



- ◆ Also tends to be over-optimistic
 - improve by jackknifing
 - assess by permutation tests
- ◆ **Jackknifing**
 - (or **cross-validation**) 'Leave one out'
 - Calculate rule for n-1 objects and then classify nth
 - Jackknifed or cross-validated estimate is proportion correctly classified
- ◆ Implementation in R (& S-PLUS)
 - Use CV=TRUE option
 - gives cross-validated predicted classes in value `$class`
 - With CV=FALSE (the default) need to use `predict.lda(.)` to get raw classification matrix:

- Example: dogmandibles, just using `length & height`

```

> attach(dogmandibles)
> species = factor(species)
> library(MASS)
> dog.lda=lda(species~length+height)

```

- Now use `dog.lda` to predict class membership of raw data

```

> dog.pred = predict(dog.lda,
+ data.frame(cbind(length,height)))

```

- Predicted class membership are held in `dog.pred$class`

```

> dog.pred$class
[1] 5 1 1 1 4 1 1 1 1 1 2 1 1 1 1 1 3 2 2 2 2 2 2 2 2 2 2 2
[30] 2 2 2 2 2 2 2 1 1 3 3 3 3 3 3 3 3 3 3 3 1 3 3 1 4 4 4 3 4
[59] 4 4 4 3 4 4 3 4 4 2 5 3 1 5 5 1 5 5 5

```

- Now tabulate against known classes:

```

> table(species,dog.pred$class)
  1  2  3  4  5
1 12  1  1  1  1
2  0 20  0  0  0
3  4  0 13  0  0
4  0  0  3 11  0
5  2  1  1  0  6

```

- Cross-validated predictions:

```

> dogCV.lda=lda(species~length+height,CV=TRUE)

```

- Cross-validated predictions held in `dogCV.lda$class`:

```

> dogCV.lda$class
[1] 5 1 1 1 4 1 1 1 1 1 2 1 1 1 1 1 3 2 2 2 2 2 2 2 2 2 2
[30] 2 2 2 2 2 2 2 1 1 3 3 3 3 3 3 3 3 3 3 3 1 3 3 1 4 4 4 3 4
[59] 4 4 4 3 4 4 3 4 4 2 5 3 1 5 5 1 5 5 5

```

- Now tabulate against known classes:

```

> table(species,dogCV.lda$class)
  1  2  3  4  5
1 12  1  1  1  1
2  0 20  0  0  0
3  4  0 13  0  0
4  0  0  3 11  0
5  2  1  2  0  5

```

- Cross-validated predictions held in `dogCV.lda$class`:

```

> table(dogCV.lda$class,dog.pred$class)
  1  2  3  4  5
1 18  0  0  0  0
2  0 22  0  0  0
3  0  0 18  0  1
4  0  0  0 12  0
5  0  0  0  0  6

```

- Now tabulate against known classes:

```

> table(species,dogCV.lda$class)
  1  2  3  4  5
1 12  1  1  1  1
2  0 20  0  0  0
3  4  0 13  0  0
4  0  0  3 11  0
5  2  1  2  0  5

```

- Cross-validated predictions held in `dogCV.lda$class`:

```

> table(dogCV.lda$class,dog.pred$class)
  1  2  3  4  5
1 18  0  0  0  0
2  0 22  0  0  0
3  0  0 18  0  1
4  0  0  0 12  0
5  0  0  0  0  6

```

- (one classification different)

- **How good is the discrimination?**
 - ◆ Note that the more dimensions the better the discrimination — see § 9.4.3.3
 - c.f. 'support vector machines' [SVMs] of computer scientists which derive an arbitrary number of dimensions until discrimination is perfect
 - ◆ One way to assess 'significance' of discrimination is by a permutation or randomization test
 - ◆ Randomly permute group labels on training data and calculate 'observed' correct classification rate
 - ◆ Compare actual rate with empirical distribution



- Random permutation by `sample(.)` :

```
> sp=sample(species)
> table(sp,predict.lda(lda(sp~length+height),
  data.frame(cbind(length,height)))$class)
  1  2  3  4  5
1  0  7  4  4  1
2  0  8  6  6  0
3  0 10  6  1  0
4  0  4  2  7  1
5  0  2  4  3  1
```

- Summary and Conclusions**

- formal problem of classifying new observations with rules constructed from training data
- The ideal is *Maximum Likelihood Rule*
 - two sample versions:– the *sample discriminant rule* and the *likelihood ratio discriminant rule*
 - can give different results for small & different sample sizes
- Maximum Likelihood Rule minimizes the total probability of misclassification
- Admissible and inadmissible rules were defined. Bayes rules are always admissible

- Summary and Conclusions (ct^d)**

- Methods for estimating misclassification probabilities were outlined, in particular by jackknifing and by using randomisation tests
- Some illustrations of the use of randomisation tests are given in Appendices 1 & 2.
 - Appendix 7 on Neural Networks gives some examples of simulation methods similar to (but more general than) jackknifing

