Today: Overview of Ch 11: Discrimination and Classification

**11.1 Intro**

Multivariate Observations on two or more groups.

Discrimination (or Separation)
How to characterize the g groups using information in the g n x p data matrices alone

Classification (or Allocation)
How to tell what group and item is from, based on its 1 x p data alone

Obviously, we can use our Discrimination result to do Classification, so the two objectives overlap.

**11.2 Two Populations:**

See Example 11.1 and Fig 11.1 pp 584-585. Better discrimination using both variables than if we had used only one (or even one at a time).

Classification probabilities P(2|1), P(1|2), P(1|1), P(2|2)

Note how to represent based on densities. p 586

Misclassification errors. See Fig 11.3. Often impossible to eliminate completely.

Max density allocation ....

Need to take into account priors and costs. Result 11.1 p 587

Special cases.... p 588

**11.3 Two Normal Populations**

Case: Equal Covariance (so easy to measure statistical distance between populations)

Result 11.2 – project centroid and x onto rotated line joining means. See which pop x is close to. (But must modify this to take account of priors and costs see box p 592.)

Example 11.3 illustrating practical utility of classification rule.

What happens when group covariances cannot be assumed equal? Concept of statistical distance becomes murky! Back to basics. pp 596-597. Result 11.3 Quadratic index.

Shows how to minimize misclassification errors (actually, expected cost of misclassification, ECM.)

Note assumption of normality for this – is not robust.

Exercise: Try using simulated data, non-normal populations, unequal covariances, with the classification rule on p 597. Check robustness.

Example of Software "lda"

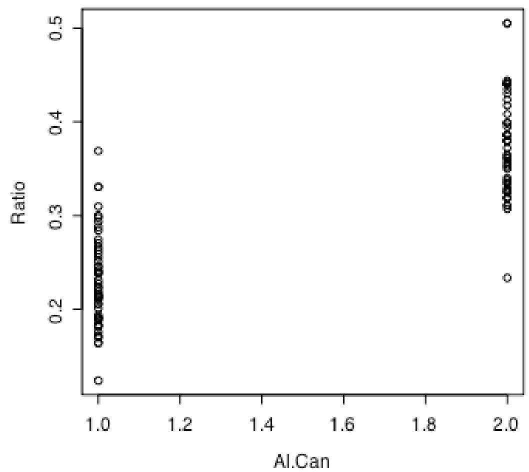
```
> T11.2.df
   Al.Can Gender Fresh Marine
1       1      2   108    368
2       1      1   131    355
3       1      1   105    469
4       1      2    86    506
5       1      1    99    402
6       1      2    87    423
7       1      1    94    440
8       1      2   117    489
9       1      2    79    432
10      1      1    99    403
11      1      1   114    428
12      1      2   123    372
13      1      1   123    372
14      1      2   109    420
15      1      2   112    394
16      1      1   104    407
17      1      2   111    422
18      1      2   126    423
19      1      2   105    434
20      1      1   119    474
21      1      1   114    396
22      1      2   100    470
23      1      2    84    399
24      1      2   102    429
25      1      2   101    469
26      1      2    85    444
27      1      1   109    397
28      1      2   106    442
29      1      1    82    431
30      1      2   118    381
31      1      1   105    388
32      1      1   121    403
33      1      1    85    451
34      1      1    83    453
35      1      1    53    427
36      1      1    95    411
37      1      1    76    442
38      1      1    95    426
39      1      2    87    402
40      1      1    70    397
41      1      2    84    511
42      1      2    91    469
43      1      1    74    451
44      1      2   101    474
45      1      1    80    398
46      1      1    95    433
47      1      2    92    404
48      1      1    99    481
49      1      2    94    491
50      1      1    87    480
51      2      1   129    420
52      2      1   148    371
53      2      1   179    407
54      2      2   152    381
55      2      2   166    377
56      2      2   124    389
57      2      1   156    419
58      2      2   131    345
59      2      1   140    362
60      2      2   144    345
61      2      2   149    393
62      2      1   108    330
63      2      1   135    355
64      2      2   170    386
65      2      1   152    301
66      2      1   153    397
67      2      1   152    301
68      2      2   136    438
69      2      2   122    306
70      2      1   148    383
71      2      2    90    385
72      2      1   145    337
73      2      1   123    364
74      2      2   145    376
```

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 75 | 2 | 2 | 115 | 354 | 88 | 2 | 1 | 144 | 403 |
| 76 | 2 | 2 | 134 | 383 | 89 | 2 | 2 | 163 | 370 |
| 77 | 2 | 1 | 117 | 355 | 90 | 2 | 2 | 145 | 355 |
| 78 | 2 | 2 | 126 | 345 | 91 | 2 | 1 | 133 | 375 |
| 79 | 2 | 1 | 118 | 379 | 92 | 2 | 1 | 128 | 383 |
| 80 | 2 | 2 | 120 | 369 | 93 | 2 | 2 | 123 | 349 |
| 81 | 2 | 1 | 153 | 403 | 94 | 2 | 1 | 144 | 373 |
| 82 | 2 | 2 | 150 | 354 | 95 | 2 | 2 | 140 | 388 |
| 83 | 2 | 1 | 154 | 390 | 96 | 2 | 2 | 150 | 339 |
| 84 | 2 | 1 | 155 | 349 | 97 | 2 | 2 | 124 | 341 |
| 85 | 2 | 2 | 109 | 325 | 98 | 2 | 1 | 125 | 346 |
| 86 | 2 | 2 | 117 | 344 | 99 | 2 | 1 | 153 | 352 |
| 87 | 2 | 1 | 128 | 400 | 100 | 2 | 1 | 108 | 339 |

```
> out=lda(Al.Can~Fresh+Marine)
> pred=predict(out)
> attributes(out)
$names
 [1] "prior"   "counts"  "means"   "scaling"
 [5] "lev"     "svd"     "N"       "call"
 [9] "terms"   "xlevels"

$class
[1] "lda"

> attributes(pred)
$names
[1] "class"     "posterior" "x"

> pred[[1]]
  [1] 2 2 1 1 1 1 1 1 1 1 1 2 2 1 1 1 1 1 1 1 1 1 1
 [24] 1 1 1 1 1 1 2 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 [47] 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
 [70] 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
 [93] 2 2 2 2 2 2 2 2
Levels: 1 2

> table(pred[[1]],Al.Can)
   Al.Can
     1  2
  1 44  1
  2  6 49
```

Alternate Approach

```
> Ratio=Fresh/Marine
> plot(Al.Can,Ratio)
```




```
> newclass=ifelse(Ratio<.3,2,1)
> compare=c(Al.Can,newclass)
> compare=cbind(Al.Can,newclass)
> compare
      Al.Can newclass
 [1,]      1        2
 [2,]      1        1
 [3,]      1        2
 [4,]      1        2
 [5,]      1        2
 [6,]      1        2
 [7,]      1        2
 [8,]      1        2
 [9,]      1        2
[10,]      1        2
[11,]      1        2
[12,]      1        1
[13,]      1        1
[14,]      1        2
[15,]      1        2
[16,]      1        2
[17,]      1        2
[18,]      1        2
[19,]      1        2
[20,]      1        2
[21,]      1        2
[22,]      1        2
[23,]      1        2
[24,]      1        2
[25,]      1        2
```

| | | |
|---|---|---|
| [26,] | 1 | 2 |
| [27,] | 1 | 2 |
| [28,] | 1 | 2 |
| [29,] | 1 | 2 |
| [30,] | 1 | 1 |
| [31,] | 1 | 2 |
| [32,] | 1 | 1 |
| [33,] | 1 | 2 |
| [34,] | 1 | 2 |
| [35,] | 1 | 2 |
| [36,] | 1 | 2 |
| [37,] | 1 | 2 |
| [38,] | 1 | 2 |
| [39,] | 1 | 2 |
| [40,] | 1 | 2 |
| [41,] | 1 | 2 |
| [42,] | 1 | 2 |
| [43,] | 1 | 2 |
| [44,] | 1 | 2 |
| [45,] | 1 | 2 |
| [46,] | 1 | 2 |
| [47,] | 1 | 2 |
| [48,] | 1 | 2 |
| [49,] | 1 | 2 |
| [50,] | 1 | 2 |
| [51,] | 2 | 1 |
| [52,] | 2 | 1 |
| [53,] | 2 | 1 |
| [54,] | 2 | 1 |
| [55,] | 2 | 1 |
| [56,] | 2 | 1 |
| [57,] | 2 | 1 |
| [58,] | 2 | 1 |
| [59,] | 2 | 1 |
| [60,] | 2 | 1 |
| [61,] | 2 | 1 |
| [62,] | 2 | 1 |
| [63,] | 2 | 1 |
| [64,] | 2 | 1 |
| [65,] | 2 | 1 |
| [66,] | 2 | 1 |
| [67,] | 2 | 1 |
| [68,] | 2 | 1 |
| [69,] | 2 | 1 |
| [70,] | 2 | 1 |
| [71,] | 2 | 2 |

```
 [72,]      2        1
 [73,]      2        1
 [74,]      2        1
 [75,]      2        1
 [76,]      2        1
 [77,]      2        1
 [78,]      2        1
 [79,]      2        1
 [80,]      2        1
 [81,]      2        1
 [82,]      2        1
 [83,]      2        1
 [84,]      2        1
 [85,]      2        1
 [86,]      2        1
 [87,]      2        1
 [88,]      2        1
 [89,]      2        1
 [90,]      2        1
 [91,]      2        1
 [92,]      2        1
 [93,]      2        1
 [94,]      2        1
 [95,]      2        1
 [96,]      2        1
 [97,]      2        1
 [98,]      2        1
 [99,]      2        1
[100,]      2        1

table(Al.Can,newclass)
      newclass
Al.Can  1  2
     1  5 45
     2 49  1
```

**11.4 Evaluating Classification Rules**

AER – actual error rate (uses estimated classification function, but known population densities)
APER – apparent actual error rate (uses estimated classification function and estimated population densities)
estimate of AER – based on data but adjusts for bias created by using data twice. p 603
– uses holdout procedure.

Example 11.7 Eyeball data to see difference in groups. See figure 11.7 p 609 and why unequal variances causes midpoint to indicate wrong balance.

**11.5 Fisher's Discriminant Function**

maximizes (linear function between variance/linear function within variance) Fig 11.8
Note sensible with equal covariances (and statistical distance is possible). Like a simple ECM rule with equal costs and priors.

**11.6 Classification with Several Populations**

General result 11.5 p 614 - same idea as with two population
see box p 614 - equal costs case

use of normality assumption for unequal cov (p 617) and equal cov (p 618)

Example 11.11 illustrating discriminant analysis, equal cov, 3 groups, normal populations

**11.7 Fisher's Discriminant Function for Several Populations**

Result see box p 630. and examples 11.13,11.14,11.15 and classification p 638.

**11.8 Comments**

qualitative variables
CART
Neural Networks
Fig 11.19 p 646/