1

Multivariate Methods

- Multivariate data
- Data display
- Principal component analysis
  - *Unsupervised learning technique*
- Discriminant analysis
  - *Supervised learning technique*
- Cluster analysis
  - *Unsupervised learning technique*
    - (Read notes on this)

2

- Measurements of ***p*** variables on each of ***n*** objects
  - e.g. lengths & widths of petals & sepals of each of 150 iris flowers
- key feature is that variables are **correlated** & observations **independent**

3

- **Data Display**
  - Scatterplots of pairs of components
    - Need to choose which components
  - Matrix plots
  - Star plots
  - etc. etc. etc.
- None is very satisfactory when ***p*** is big
  - Need to ***select*** best components to plot
  - i.e. need to ***reduce dimensionality***

4

- Digression on **R** language details:
  - Many multivariate routines in library `mva`
  - So far only considered data in a *dataframe*
  - Multivariate methods in **R** often need data in a ***matrix***
  - Use commands such as
    - `as.matrix(.)`
    - `rbind(.)`
    - `cbind(.)`
  - Which create matrices (see `help`)

5

- **Principal Component Analysis (PCA)**
  - Technique for finding which linear combinations of variables contain most information.
  - Produces a new coordinate system
    - Plots on the first few components are like to show structure in data (i.e. information)
  - Example:
    - Iris data

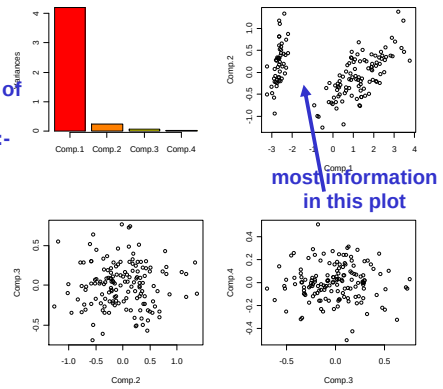
6

```
> library(mva)
> library(MASS)
> par(mfrow=c(2,2))
> data(iris)
> attach(iris)
> ir<-cbind(Sepal.Length, Sepal.Width, Petal.Length,
+ Petal.Width)
> ir.pca<-princomp(ir)
> plot(ir.pca)
> ir.pc<-predict(ir.pca)
> plot(ir.pca$scores[,1:2])
> plot(ir.pca$scores[,2:3])
> plot(ir.pca$scores[,3:4])
```

This creates a matrix `ir` of the iris data, performs pca, uses the generic `predict` function to calculate the coordinates of the data on the principal components and plots the first three pairs of components

7

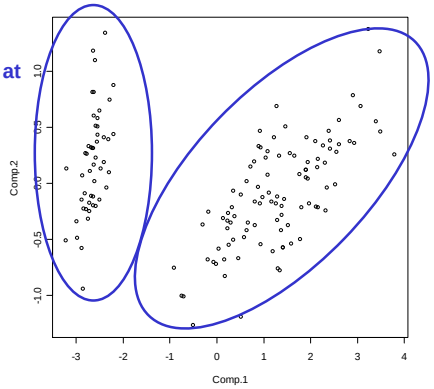
Shows importance of each component:- most information in first component



most information in this plot

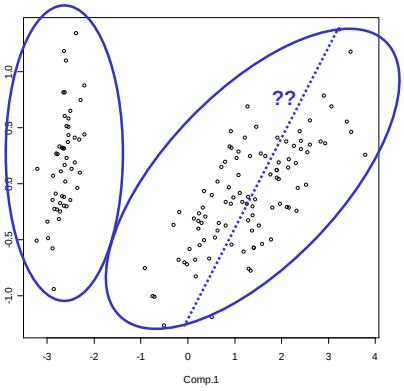
8

Can detect at least two separate groups in data and maybe……



9

Can detect at least two separate groups in data and maybe one group divides into two



10

- Can interpret principal components as reflecting features in the data by examining loadings
  - away from the main theme of course
  - see example in notes.

- Principal component analysis is a useful basic tool for investigating data structure and reducing dimensionality

11

- **Discriminant Analysis**
  - Key problem is to use multivariate data on different types of objects to classify future observations.
  - e.g. the iris flowers are actually from 3 different species (50 of each)
  - What combinations of sepal & petal length & width are most useful in distinguishing between the species and for classifying new cases

12

- e.g. consider a plot of petal length *vs* width
  - First set up a vector to label the three varieties as **s** or **c** or **v**

```
> ir.species<-factor(c(rep("s",50),
+ rep("c",50),rep("v",50)))
```

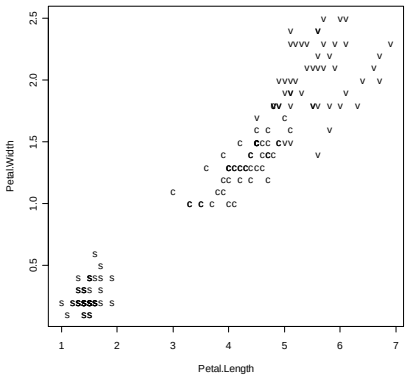
  - Then create a matrix with the petal measurements

```
> petals<-cbind(Petal.Length,
+ Petal.Width)
```

  - Then plot the data with the labels

```
> plot(petals,type="n")
> text(petals,
+ labels=as.character(ir.species))
```

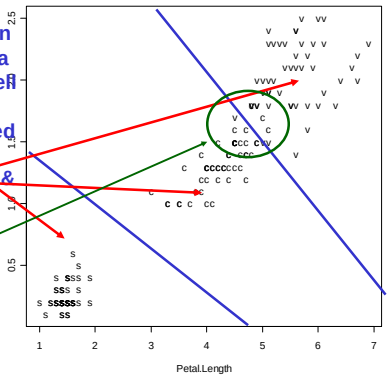

13



14

Informally we can see some separation between the species and if a new observation fell in this region it should be classified as type "s" & here as type "c" & here as type "v"

However, some misclassification here between "c" and "v"



15

- This method uses just petal length & width
  - makes some mistakes
- Could we do better with all measurements?
- ***linear discriminant analysis (LDA)*** will give the best method when boundaries between regions are straight lines
  - And ***quadratic discriminant analysis (QDA)*** when boundaries are quadratic curves

16

- How do we evaluate how well LDA performs?
  - Could use rule to classify data we actually have
    - Can use generic function **`predict(.)`**

```
> ir.lda<- lda(ir,ir.species)
> ir.ld<- predict(ir.lda,ir)
> ir.ld$class
  [1] s s s s s s s s s s s s s s s s s s s s s s s s s s s s s s s s s s s s s s
 [38] s s s s s s s s s s s s s s c c c c c c c c c c c c c c c c c c c c c c c c
 [75] c c c c c c c c c v c c c c c c c c c c c c c c c c c c c c c c c c c c c v
[112] v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v v
[149] v v
Levels: c s v
```


17

- Or in a table:

```
> table(ir.species, ir.ld$class)
ir.species  c  s  v
         c 48  0  2
         s  0 50  0
         v  1  0 49
```

- Which shows that
  - Correct classification rate 147/150
  - 2 species C classified as V
  - 1 species V classified as a C

18

- However, this is optimistic ('cheating')
  - Using same data for calculating the `lda` and evaluating its performance
  - Better would be ***cross-validation***
    - Omit one observation
    - Calculate lda
    - Use lda to classify this observation
    - Repeat on each observation in turn
  - Also better would be a ***randomization test***
    - Randomly permute the species labels

19

- Use `sample(ir.species)` to permute labels
  - Note:- sampling ***without replacement***

```
> randomspecies<-sample(ir.species)
> irrand.lda<-lda(ir,randomspecies)
> irrand.ld<-predict(irrand.lda,ir)
> table(randomspecies,irrand.ld$class)
randomspecies  c  s  v
            c 29 17  4
            s 17 28  5
            v 17 20 13
```

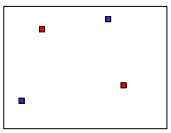
  - Which shows that only 70 out of 150 would be correctly classified
    - (compare 147 out of 150)

20

- This could be repeated many times and the ***permutation distribution*** of the correct classification rate obtained.
  - (or strictly the ***randomization distribution***)
- The observed rate of 147/150 would be in the extreme tail of the distribution
  - i.e. the observed rate of 147/150 is much higher than could occur by chance

21

- **General Comment**
  - If we have **high** number of dimensions & **small** number of points then *always* easy to get ***near perfect*** discrimination
  - A randomization test will show if a high classification rate is a result of a real difference between cases or just geometry.
    - e.g. 2 groups, 2 dimensions
      3 points ➔ always perfect
      4 points ➔ 75% chance of perfect discrimination
    - 3 dimensions always perfect with 2 groups 4 points



22

- To estimate the true classification rate we should apply the rule to new data
  - e.g. to construct the rule on a random sample and apply it to the other observations

```
> samp<- c(sample(1:50,25),
+ sample(51:100,25), sample(101:150,25))
```

  - `samp` will contain
    - 25 numbers from 1 to 50
    - 25 from 51 to 100
    - 25 from 101 to 150

23

```
> samp
 [1]  43   7  46  10  19  47   5  49  45  37  33   8  12  28
27  11   2  29   1
[20]  32   3  14   4  25   6  54  92  67  74  89  71  81  97
62  73  93  99  60
[39]  58  70  51  94  83  72  66  59  65  86  98  82 132 101
139 108 138 112 125
[58] 146 103 129 109 124 102 137 121 147 144 128 116 131 113
104 148 115 122
```

- So `ir[samp,]`will have just these cases
  - With 25 from each species
  - `ir[-samp,]`will have the others
- Use `ir[samp,]` to construct the lda and then predict on `ir[-samp,]`

24

```
> irsamp.lda<-
  lda(ir[samp,],ir.species[samp])
> irsamp.ld<-predict(irsamp.lda, ir[-
  samp,])
> table(ir.species[-samp], irsamp.ld$class)
     c  s  v
  c 22  0  3
  s  0 25  0
  v  1  0 24
```

- So rule classifies correctly 71 out of 75
- Other examples in notes

25

- **Summary**
  - PCA was introduced
  - Ideas of discrimination & classification with lda and qda outlined
  - Ideas of using analyses to ***predict*** illustrated
  - Taking random permutations & random samples illustrated
- Predictions and random samples will be used in other methods for discrimination & classification using neural networks etc.

26

27

28

29

30