STAT3401: Introduction to Cluster Analysis

Paul Hewson

30th November 2006

Aims of the week

- Rationale for unsupervised classification
- Cluster analysis: types of analysis
- Methods for hierarchical analysis
- kmeans analysis
- Assessing cluster solutions

Motivation for cluster analysis

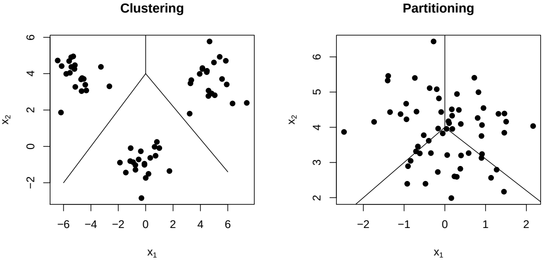


Figure: Artificial data suggesting a difference between "clustering" and "dissecting"

Clustering algorithms

There are a wide range of algorithms that have been developed to investigate clustering within data. These can be considered in a number of ways:

- Hierarchical Methods
  - Agglomerative clustering (`hclust()`, `agnes()`)
  - Divisive clustering (`diana()`, `mona()`)
- Partitioning methods (`kmeans()`, `pam()`, `clara()`)

Consider the following distance matrix

	a	b	c	d	e
a	0				
b	2	0			
c	6	5	0		
d	10	9	4	0	
e	9	8	5	3	0

Each individual is in it's own cluster!

Nearest neighbour / Single linkage

- Use `method = "single"` instruction in the call to `hclust()`
- Finds “friends of friends” to join each cluster (c.f. minimum spanning trees).
- Decision to merge groups is based on the distance of the *nearest* member of the group to the *nearest* other object.

In our example, with a distance of 2, inviduals *a* and *b* are the most similar.

Nearest neighbour clustering

	a	b	c	d	e
a	0				
b	2	0			
c	6	5	0		
d	10	9	4	0	
e	9	8	5	3	0

We therefore merge these into a cluster at level 2:

Distance	Groups
0	a b c d e
2	(ab) c d e

Next step:

and we now need to re-write our distance matrix, whereby:

$$
\begin{aligned}
d_{(ab)c} &= min(d_{ac}, d_{bc}) = d_{bc} = 5 \\
d_{(ab)d} &= min(d_{ad}, d_{bd}) = d_{bc} = 9 \\
d_{(ab)e} &= min(d_{ae}, d_{be}) = d_{bc} = 8
\end{aligned}
$$

This gives us a new distance matrix

$$
\begin{array}{r|cccc}
 & (ab) & c & d & e \\
\hline
(ab) & 0 & & & \\
c & 5 & 0 & & \\
d & 9 & 4 & 0 & \\
e & 8 & 5 & 3 & 0
\end{array}
$$

What do we merge next?

Next step

Distance	Groups
0	$a\ b\ c\ d\ e$
2	$(ab)\ c\ d\ e$
3	$(ab)\ c\ (de)$

So, find the minimum distance from d and e to the other objects and reform the distance matrix:

	(ab)	c	(de)
(ab)	0		
c	5	0	
(de)	8	4	0

And so on . . .

Clearly, the next merger is between (de) and c, at a height of 4, the final merger will take place at a height of 5.

Distance	Groups
0	$a\ b\ c\ d\ e$
2	$(ab)\ c\ d\ e$
3	$(ab)\ c\ (de)$
4	$(ab)\ (cde)$
5	$(abcde)$

Furthest neighbour / Complete linkage

- Use the `method = "complete"` instruction in the call to `hclust()`
- Finds “similar” clusters.
- Objects are merged when the *furthest* member of the group is close enough to the new object.

Working it though

	a	b	c	d	e
a	0				
b	2	0			
c	6	5	0		
d	10	9	4	0	
e	9	8	5	3	0

Starts as before, merge *a* and *b* as these are the nearest:

Distance	Groups
0	a b c d e
2	(ab) c d e

Furthest neighbour

Life changes now when we calculate the new distance matrix:

$$\begin{aligned} d_{(ab)c} &= max(d_{ac}, d_{bc}) = d_{bc} = 6 \\ d_{(ab)d} &= max(d_{ad}, d_{bd}) = d_{bc} = 10 \\ d_{(ab)e} &= max(d_{ae}, d_{be}) = d_{bc} = 9 \end{aligned}$$

	(ab)	c	d	e
(ab)	0			
c	6	0		
d	10	4	0	
e	9	5	3	0

So what do we merge next?

Actually we still merge *d* and *e*, but note the height!

Distance	Groups
0	a b c d e
2	(ab) c d e
3	(ab) c (de)

And reforming the new distance matrix:

	(ab)	c	(de)
(ab)	0		
c	6	0	
(de)	10	5	0

Compare the next merge with the same step before, but compare the heights (noting this is a very artificial example)

Completing the clustering

Distance	Groups
0	a b c d e
2	(ab) c d e
3	(ab) c (de)
5	(ab) (cde)

and the final distance matrix:

	(ab)	(cde)
(ab)	0	
(cde)	10	0

Final merge at height 10

Distance	Groups
0	a b c d e
2	(ab) c d e
3	(ab) c (de)
5	(ab) (cde)
10	(abcde)

This is a very artificial example. Merges happen in the same order, but at different heights. In more realistic examples you would expect to see some different mergers taking place

Group average link

This is the last example we will work by hand

- Requires `agnes()` in package `cluster`
- Use with the `method="average"` instruction.
- Merge two groups is the average distance between them is small enough

Continuing the clustering

Again, we start by merging a and b, but again the reduced distance matrix will be different:

$$
\begin{aligned}
d_{(ab)c} &= (d_{ac} + d_{bc})/2 = d_{bc} = 5.5 \\
d_{(ab)d} &= (d_{ad} + d_{bd})/2 = d_{bc} = 9.5 \\
d_{(ab)e} &= (d_{ae} + d_{be})/2 = d_{bc} = 8.5
\end{aligned}
$$

	(ab)	c	d	e
(ab)	0			
c	5.5	0		
d	9.5	4	0	
e	8.5	5	3	0

Next merge (same order, different height)

Merge *d* and *e*, at height 3:

Distance	Groups
0	a b c d e
2	(ab) c d e
3	(ab) c (de)

Again, need to recalculate distances:

	(ab)	c	(de)
(ab)	0		
c	5.5	0	
(de)	9	4.5	0

and leaping on a bit

after merging (*de*) and *c*:

	(ab)	(cde)
(ab)	0	
(cde)	7.8	0

our final merge will take place at height 7.8.

Distance	Groups
0	a b c d e
2	(ab) c d e
3	(ab) c (de)
4.5	(ab) (cde)
7.8	(abcde)

We can plot this information

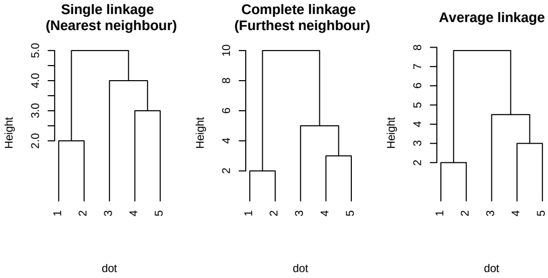


Figure: Dendrograms from three basic cluster methods

Don't be misled by this simple example!

Other methods for clustering

- Cluster "analysis" is an *algorithmically* guided exploratory data analysis
- Many other methods proposed
- Attempts to generalise the algorithm.

Lance and Williams recurrence formula

$$d_{C_k \bigcup C_l, C_m} = \alpha_l d(C_k, C_l) + \alpha_m d(C_k, C_m) + \beta d(C_k, C_l) + \gamma |d(C_k, C_m) - d(C_l, C_m)| \quad (1)$$

- $d_{C_k \bigcup C_l, C_m}$ is the distance between a cluster C_k and the merging of two groups C_l and C_m.
- Parameters are constrained:
- $\alpha_l + \alpha_m + \beta = 1$
- $\alpha_l = \alpha_m$, $\beta < 1$
- $\gamma = 0$.

Lance and Williams recurrence formula

Method	R call	α_k	β	γ
Single link (nearest neighbour)	`method = "single"`	$\frac{1}{2}$	0	$-\frac{1}{2}$
Complete link (furthest neighbour)	`method = "complete"`	$\frac{1}{2}$	0	$\frac{1}{2}$
Group average link	`method = "average"`	$N_l\|(N_l + N_m)$	0	0
Weighted average link	`method = "mcquitty"`	$\frac{1}{2}$	0	0
Centroid	`method = "centroid"`	$N_l\|(N_l + N_m)$	$-N_l N_m\|(N_l + N_m)^2$	0
Incremental sum of squares	`method = "ward"`	$\frac{N_k+N_m}{N_k+N_l+N_m}$	$\frac{N_k+N_l}{N_k+N_l+N_m}$	0
Median	`method = "median"`	$\frac{1}{2}$	$-\frac{1}{4}$	0

where N_k, N_l and N_m are the cluster sizes when C_k is joined to the other two clusters considered.

Popular alternatives

Ward's method:

- Minimises the error sum of squares:

$$ESS_k = \sum_{i+1}^{n_k} \sum_{j=1}^{p} (x_{ki,j} - \bar{\mathbf{x}}_{k,j})^2$$

  where $\bar{\mathbf{x}}_{k,j}$ is the mean of cluster k with respect to variable j and $x_{ki,j}$ is the value of j for each object i in cluster k. The total error sum of squares given by $\sum_{k=1}^{K} ESS_k$ for all clusters k.

- Ward's method tends to give spherical clusters (whether reasonable or not).

Problems with hierarchical clustering

There are several well known problems, such as reversals in the dendrogram and (with single link clustering): chaining

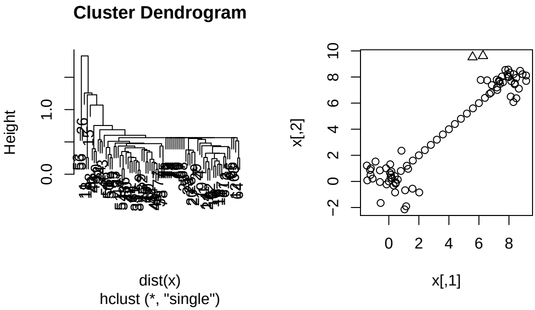


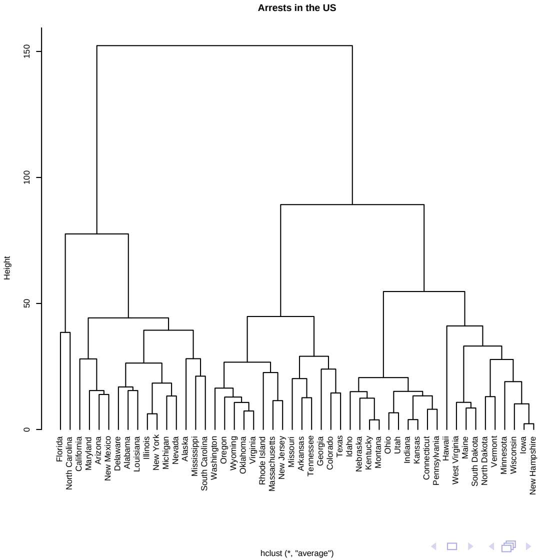
Figure: Demonstration of “chaining” with single link clustering

hierarchical clustering in R

1. Create a distance matrix
2. Apply the cluster algorithm
3. Plot the results (dendrogam)
4. Cut the tree at a suitable point if you want distinct groups, plot the original data with this classification
5. Get some measures of fit

```
> USArrests.dist <- dist(USArrests,
       method = "manhattan")
> USArrests.hclust <- hclust(USArrests.dist,
       method = "complete")
> plot(USArrests.hclust)
> US.cut <- cutree(USArrests.hclust, k=5)
> plot(USArrests, col = US.cut, pch = US.cut)
```

Dendrogram


Arrests in the US
Height
0
50
100
150
Florida
North Carolina
California
Maryland
Arizona
New Mexico
Delaware
Alabama
Louisiana
Illinois
New York
Michigan
Nevada
Alaska
Mississippi
South Carolina
Washington
Oregon
Wyoming
Oklahoma
Virginia
Rhode Island
Massachusetts
New Jersey
Missouri
Arkansas
Tennessee
Georgia
Colorado
Texas
Idaho
Nebraska
Kentucky
Montana
Ohio
Utah
Indiana
Kansas
Connecticut
Pennsylvania
Hawaii
West Virginia
Maine
South Dakota
North Dakota
Vermont
Minnesota
Wisconsin
Iowa
New Hampshire
hclust (*, "average")

Cophenetic Correlation

The cophenetic correlation can be used as some kind of measure of the goodness of fit of a particular dendrogram.

$$\rho_{Cophenetic} = \frac{\sum_{i=1,j=1,i<j}^{n}(d_{ij} - \bar{d})(h_{ij} - \bar{h})}{\left(\sum_{i=1,j=1,i<j}^{n}(d_{ij} - \bar{d})^2(h_{ij} - \bar{h})^2\right)^{0.5}} \tag{2}$$

Easily extracted in R, but less clear what it means. A value below 0.6 implies some distortion in the dendrogram.

And now for something completely un-different

- Friday 23 March 2007
- Royal Statistical Society
- 12 Errol Street, London

Sounds like a great place for a Party!

Karl Pearson

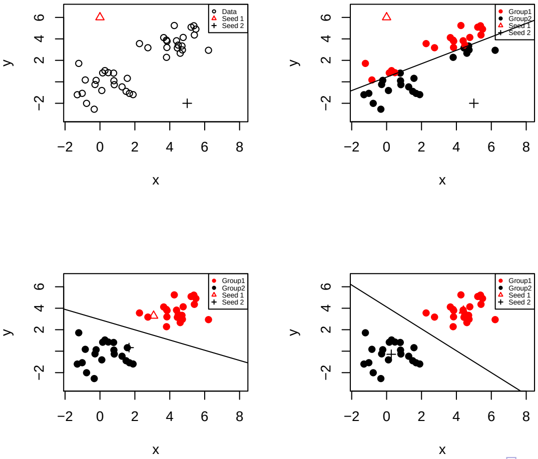
- Born March 27, 1857 Islington, died 1936
- His mother came from Hull, most of his family from the North Riding
- Einstein's Olympia Academy stated Pearson's "The Grammar of Science" was essential reading
- Many academic honours (FRS, DSc from University of London) but refused OBE and Knighthood.
- χ^2 test, work on correlation and regression, classified probability distributions (especially the exponential family distributions)

k-means clustering

A very different approach

- Aimed at finding “more homogenous” subgroups within the data.
- We specify at the start how many clusters we are looking for,
- Ideally provide some clue as to where those clusters might be
- Given a number of k starting points, the data are classified, the centroids recalculated and the process iterates until stable.

k-means demo with silly seed points



k-means clustering in R

Very easily applied

```
> US.km <- kmeans(USArrests, centers = 5)
> plot(USArrests, col = US.km$cluster,
    pch = US.km$cluster)
```

But you could compare the Rand index from this solution with that from hierarchical clustering

Rand Statistic

- Applied to the classes you identify (in the USArrests data we decided 5 classes were appropriate) and can be used with other clustering methods
- Compares cluster solutions in terms of group membership
- A value of 1 implies perfect agreement

Summary

- Cluster analysis is all about finding groups of individuals who are more "alike" than others
- Vast number of applications
- Hierarchical clustering is based upon a distance[1] matrix; - we have worked examples with nearest / furthest neighbour and group average but there are popular alternatives (Ward's)
- k-means is another approach to clustering
- Can extract and plot group memberships
- Cophenetic correlation can assess how much disortion is caused by hierarchical clustering; Rand index compares two cluster solutions
- Algorithm based approach - unpopular with some! Each method has advantages and disadvantages

[1] by whatever method

Some silly numbers

- $S_{k,n}$, the number of ways of partitioning n objects into k groups is given by:

$$S_{k,n} = \frac{1}{k!} \sum_{j=1}^{k} \left(\begin{array}{c} k \\ j \end{array} \right) (-1)^{k-j} j^n \approx_{n \to \infty} \frac{k^n}{k!}$$

a second type Stirling number.

- Where k is not specified we have $\sum_{k=1}^{K} S_{k,n}$ partitions.
- For $n = 50$ and $k = 2$ this is in the order of 6×10^{29}

Some silly numbers

- These Stirling numbers assume you have *one* method
- As you've seen, there are many different distance measures, many different clustering algorithms
- Outside our world, there are also numerous other methods for unsupervised classification, trees and forests being the most common (recursive partitioning / classification and regression trees)
- Some statisticians prefer mixture models

Further reading

- See the online reading list (soon) for a few useful resources, including links to animations
- Manly chapter 9
- Johnson and Wichern chapter 12

Common exam questions

- Explain why we use cluster analysis, give some examples where it has been useful
- Explaining about different types of cluster analysis, and demonstrate awareness of the various sub-methods
- Explain and work out distances, especially Gower's, Euclidean, Minkowski, Manhattan etc.
- Explain and work through a simple cluster analysis using nearest, furthest or group average linkage
- Justify a choice of cluster solution plots, diagnostics (cophenetic, Rand)
- Interpret some cluster results in context
- Anything else you might like to suggest . . .

Lab. work

- You've been given notes for a worked example using `milk` data in `flexclust` library on the chemical composition of mammalian milk from a number of species
- This example is a bit obvious, all methods seem to lead to the same solutions, but check out the methods - distance (choice of measure), hierarchical cluster (choice of method), graphics (dendrogram, then cut at a suitable point), graphics (look at groupings on original data), diagnostics
- Then carry out a cluster analysis using `nutrient` data from `flexclust`
- Interpret a cluster analysis of the class (use `daisy()` to get Gower's out)!