Contents

1 Cluster Analysis

1.1 Introduction

1.1.1 Cluster Analysis: a summary and grouping method

Cluster analysis (or data segmentation) is designed to group (or segment) a set of objects in sub-sets or clusters, such that objects belonging to the same group are more homogeneous than objects belonging to different groups.

Objectives can be different: it is used to

- describe an object on the basis on its relationship with other objects, as well as through a set of measurements.
- assess whether data consist of a set of well-defined groups, each group composed of objects with substantially different properties.
- achieve a dimensionality reduction of $\Re^n$: from n observed units to g homogeneius groups ($g << n$). (Note: cluster analysis can also be applied to variables, even if it is rarer.)
- arrange groups according to a natural hierarchy: clusters are gradually grouped so that at each hierarchical level, clusters of the same group are more similar than clusters within different groups are.

The problem of classifying

The problem of unit classification into homogeneous groups can be distinguished in:

- discriminant analysis

  The different groups which the n observed units belong to are known *a priori*. Therefore, the objective is to establish, on the basis on the p variables observed for each unit, a criterion that properly assigns additional units to the right group, minimizing the attribution error.

This problem is of supervised learning type.

- cluster Analysis

  We attempt to find homogeneous groups in the n units, without knowing a priori the existence of such groups.

This is a typical explorative method.

1.1.2 Cluster analysis: main tools and methods

A big issue is that clustering methods will return clusters even if the data does not contain any clusters. Therefore, it's necessary

- to assess clustering tendency before the analysis and
- to validate the quality of the result after clustering.

1.1.2.1 Assessing Clustering Tendency

Before applying any clustering method on your data, it's important to evaluate whether the data sets contains meaningful clusters (i.e.: non-random structures) or not. If yes, then how many clusters are possibly there.

The simplest approach to assess whether data contains any meaningful clusters is a *visual inspection* of the data

- *scatter plots* of the objects *wrt* the observed variables.
  - In this regard, preliminarly we have to select the *variables* wrt which individual profiles are compared: they must be *discriminating* or relevant in order to characterize groups.
  - E.g., plots of the first two or three PCs, or MDS graphs.
- **icon plots**, e.g., *Chernoff faces*, or *Andrews curves.*
- **ordered dissimilarity matrix**

Statistical tests can also be used to assess clustering tendency. E.g., the Hopkins statistic is used to evaluate whether data are uniformly distributed or are not.

Once a sort of clustering is evident, it's useful to take advantage of algorithms specific to cluster analysis in order to make an explicit distinction or classification.

The two most popular classes of clustering methods are

- **hierarchical** algorithms which recursively find nested clusters
- **partitional** algorithms which find all the clusters simultaneously as a partition of the data and do not impose a hierarchical structure.

Other two relevant approaches to clustering are

- **density-based** (or mode seeking) clustering
- **grid based** clustering

Hierarchical algorithms recursively find nested clusters

- either in **agglomerative (bottom-up)** mode: starting with each data point in its own cluster, merge the most similar pair of clusters successively to form a cluster hierarchy
- or in **divisive (top-down)** mode: starting with all the data points in one cluster, recursively divide the cluster into smaller clusters
- Input is an $n \times n$ similarity matrix
- produce ... partitions

Partitional algorithms which find all the clusters simultaneously as a partition of the data and do not impose a hierarchical structure.

- require the number of clusters chosen at priori
- depend on the initial configuration
- generally produce one partition
- input is either an $n \times p$ pattern matrix (K-means) or an $n \times n$ similarity matrix (Spectral clustering)

1.1.2.2 Distance or dissimilarity

A key point in cluster analysis is the notion of the degree of proximity (affinity) or of **distance** or **dissimilarity** (difference or lack of affinity) between the individual objects to group.

Distances between objects can be represented by a $n \times n$ **distance matrix** or dissimilarity matrix.

$$D = \begin{pmatrix} 0 & d_{12} & \cdots & \cdots & d_{1n} \\ & 0 & \cdots & \cdots & d_{2n} \\ & & & \ddots & \vdots \\ & & & & 0 \end{pmatrix}$$

where $d_{ii'}$ is the distance between objects i e $i^{'}$.

Distance matrix, which is an input of the clustering algorithm, is symmmetric, with null diagonal elements, and generally consists in $d_{ii'} \geq 0$.

In particular, some algorithms require distances in strict sense (semidefinite positive matrix).

1.2 Hierarchical methods

Bottom-up hierarchical algorithms

step **0**. Each unit costitutes a group (*singleton* clusters).

Distance between groups is given by matrix D.

step **1**. The two groups which have minimum distance are merged.

step **2**. Distance between the new group and already exixting groups is calculated.

Dimension of D matrix decreases of one unit.

step **3**. Steps 1 and 2 are repeated until one group containing all units is constituted.

Algorithm requires $\ldots$ iterations.

1.2.1 Distance between groups

Hierarchical methods need to define distance between two groups.

On the basis of this definition, hierarchical method is called:

- **single linkage** (or *nearest neighbour*): distance between two groups is measured by the minimum distance between units belonging to one group and those belonging to the other group.

- **complete linkage** (or *furthest neighbour*): distance between two groups is measured by the maximum distance (= diameter of sphere ⊃ all points ∈ the two groups).
- **average linkage**: distance between two groups is measured by the arithmetic mean of distances between all units constituing the two groups.

- **centroid**: distance between groups is given by the distance between centroids, which are the mean profiles calculated on units belonging to either group.

  Distances at which fusions occur can be non-increasing.
- **Ward**: is based on the decomposition of total deviance into deviance between groups and deviance within groups.

  At each step the union of all possible couples of clusters is considered, and the couple leading to minimum increase in within-deviance after merging is merged.

1.2.2 Dendrogram

Clustering process can be graphically represented by a **dendrogram**.

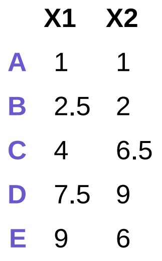
It consists in a *tree with binary ramifications* having

- units on x-axis;
- distance levels at which fusions occur, on y-axis.

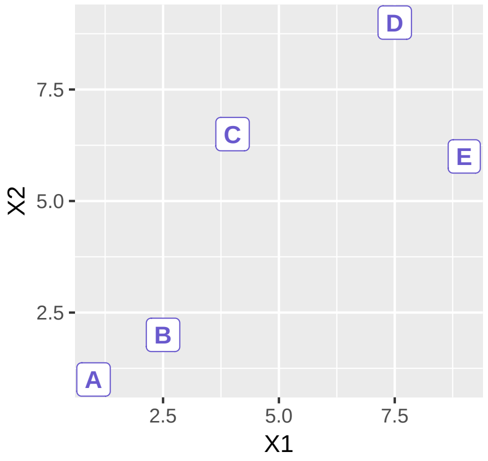
  Along the agglomerative process, the rate between a distance level and that immediately below represents a measure of separation of the clusters considered in the fusion: the larger the separation the less the proximity of one cluster to the other.
- Each level corresponds to a group partition.

1.2.3 A toy example

From Multivariate Statistical Methods, A primer by Bryan F.J.Manly (1991).



	X1	X2
A	1	1
B	2.5	2
C	4	6.5
D	7.5	9
E	9	6



The distance matrix is:

	X1	X2
A	1	1
B	2.5	2
C	4	6.5
D	7.5	9
E	9	6

	A	B	C	D	E
A	0	2	6	10	9
B	2	0	5	9	8
C	6	5	0	4	5
D	10	9	4	0	3
E	9	8	5	3	0

E.g., (euclidean) distance between A and B (rounded) is

$$\sqrt{(1-2.5)^2+(1-2)^2} \cong 2$$

Single Linkage method

Iteration	Distance	Groups
0	0	A,B,C,D,E
1	2	(A,B),C,D,E
2	3	(A,B),C,(D,E)
3	4	(A,B),(C,D,E)
4	5	(A,B,C,D,E)

---

0. 5 singleton clusters, D_0

1. The closest are A and B, compute D_1

2. The closest are D and E, compute D_2

3. The closest are C and DE, compute D_3

4. We get only one group.

D_0

	A	B	C	D
B	2			
C	6	5		
D	10	9	4	
E	9	8	5	3

D_1

	AB	C	D
C	5		
D	9	4	
E	8	5	3

D_2

	AB	C
C	5	
DE	8	4

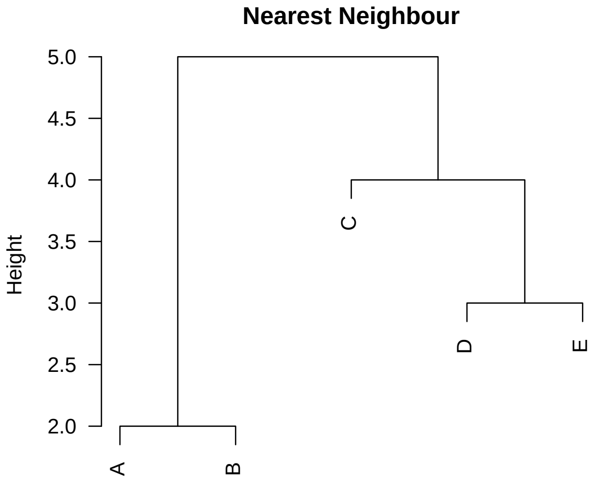
D_3

	AB
CDE	5

D_4

ABCDE

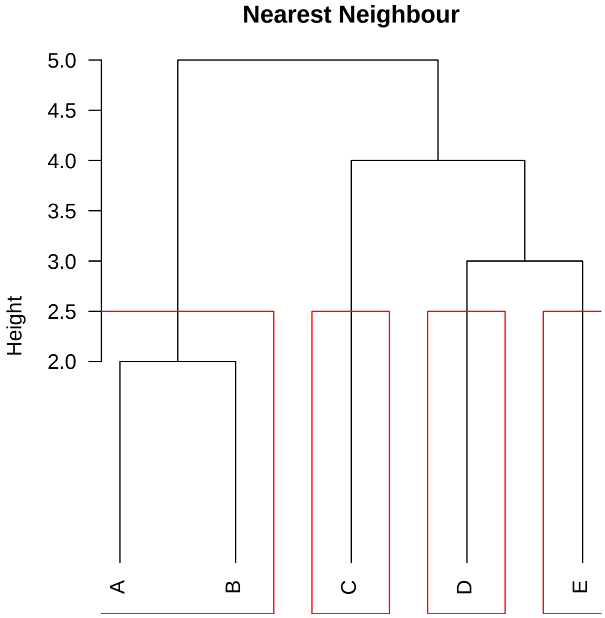
The dendrogram



A separation measure:

```
#> [1] 1.500000 1.333333 1.250000
```

The most supported partition:



Complete Linkage method

Iteration	Distance	Groups
0	0	A,B,C,D,E
1	2	(A,B),C,D,E
2	3	(A,B),C,(D,E)
3	5	(A,B),(C,D,E)
4	10	(A,B,C,D,E)

---

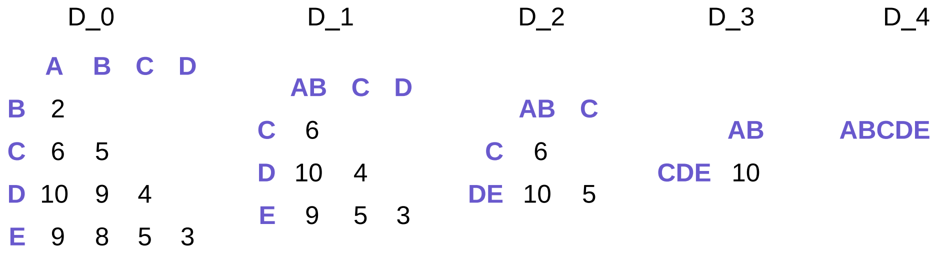
0. 5 singleton clusters, D_0

1. The closest are A and B, compute D_1

2. The closest are D and E, compute D_2

3. The closest are C and DE, compute D_3

4. We get only one group.



D_0

	A	B	C	D
B	2			
C	6	5		
D	10	9	4	
E	9	8	5	3

D_1

	AB	C	D
C	6		
D	10	4	
E	9	5	3

D_2

	AB	C
C	6	
DE	10	5

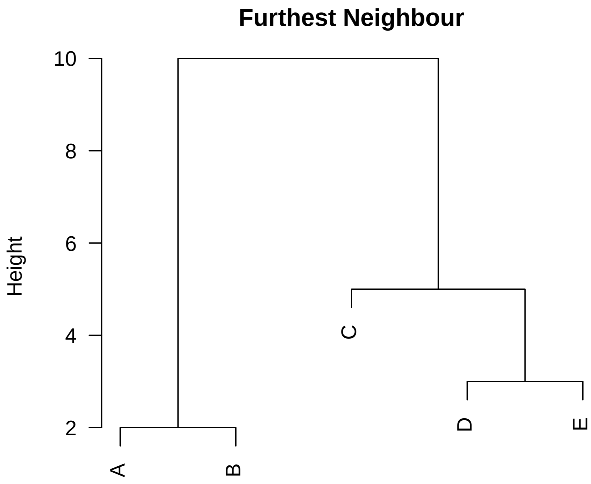
D_3

	AB
CDE	10

D_4

ABCDE

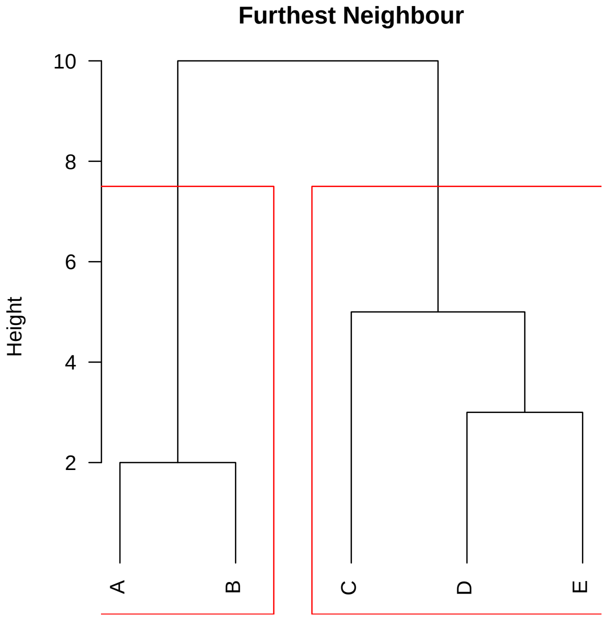
The dendrogram



A separation measure:

```
#> [1] 1.500000 1.666667 2.000000
```

The most supported partition:



1.3 Partitional methods

- Partitional methods produce as output a partition of the collective into a number of groups.
  - A partition of a set of n units consists in assigning these to disjoint and exhaustive subsets of the starting set.
- General procedure

i. Identify a temporary partition of the collective in a certain number of groups (by hierarchical cluster analysis or a priori information, whether or not specifying the number of groups);
ii. optimize an objective function by changing the assignment.

- The most popular partitional method is the algorithm k**-means** (by Hartigan and Wong)

1.3.1 K-means method

Requires in input the k number of groups to form.

The objective function to minimize is the deviance within groups.

The algorithm proceeds as follows.

---

0. Specify k initial centroids.

```
* Centroids are the cluster centers (mean profiles)
* They can be derived from a previously made hierarchical analysis;
in the absence of information one can randomly choose the profiles
of k units (as in `R` routine `kmeans`).
```

1. Each unit is assigned to the initial k centroids according to the smallest distance.

2. Calculate the centroid for each of the formed k groups and verify each unit is assigned to the group with the closest centroid.

3. If this is not true, move the unit to the group that has the nearest centroid.

4. Repeat 2. and 3. until you reach a stable configuration, or square error (deviance) decreases minimally after a number of iterations.

1.4 Clustering Validation

The term **clustering validation** is used to design the procedure of evaluating the results of a clustering algorithm.

A variety of measures has been proposed in the literature as clustering validation statistics. Generally, they can be categorized into 4 classes

- **External** clustering validation, which consists in comparing the results of a cluster analysis to an externally known result, such as externally provided class labels. Since we know the "true" cluster number in advance, this approach is mainly used for selecting the right clustering algorithm for a specific dataset.

- **Internal** clustering validation, which use the internal information of the clustering process to evaluate the goodness of a clustering structure without reference to external information. It can be also used for estimating the number of clusters and the appropriate clustering algorithm without any external data.
- **Clustering stability** validation, which is a special version of internal validation. It evaluates the consistency of a clustering result by comparing it with the clusters obtained after each column is removed, one at a time.
- **Relative** clustering validation, which evaluates the clustering structure by varying different parameter values for the same algorithm (e.g.,: varying the number of clusters k). It's generally used for determining the optimal number of clusters.

An overview of recent advances is in Jain (2010).

References

Jain, Anil K. 2010. "Data Clustering: 50 Years Beyond K-Means." *Pattern Recognition Letters* 31 (8): 651–66. doi:10.1016/j.patrec.2009.09.011.