WIKIPEDIA
The Free Encyclopedia

Main page
Contents
Featured content
Current events
Random article
Donate to Wikipedia
Wikipedia store

Interaction
Help
About Wikipedia
Community portal
Recent changes
Contact page

Tools
What links here
Related changes
Upload file
Special pages
Permanent link
Page information
Wikidata item
Cite this page

Print/export
Create a book
Download as PDF
Printable version

Languages
Русский
Edit links

Complete-linkage clustering

From Wikipedia, the free encyclopedia

This article **needs additional citations for verification**. Please help improve this article by adding citations to reliable sources. Unsourced material may be challenged and removed. *(September 2010)*

This article **needs attention from an expert in Statistics**. Please add a *reason* or a *talk* parameter to this template to explain the issue with the article. WikiProject Statistics (or its Portal) may be able to help recruit an expert. *(February 2011)*

Complete-linkage clustering is one of several methods of agglomerative hierarchical clustering. At the beginning of the process, each element is in a cluster of its own. The clusters are then sequentially combined into larger clusters until all elements end up being in the same cluster. At each step, the two clusters separated by the shortest distance are combined. The definition of 'shortest distance' is what differentiates between the different agglomerative clustering methods. In complete-linkage clustering, the link between two clusters contains all element pairs, and the distance between clusters equals the distance between those two elements (one in each cluster) that are farthest away from each other. The shortest of these links that remains at any step causes the fusion of the two clusters whose elements are involved. The method is also known as **farthest neighbour clustering**. The result of the clustering can be visualized as a dendrogram, which shows the sequence of cluster fusion and the distance at which each fusion took place.[1][2][3]

Mathematically, the complete linkage function — the distance $D(X,Y)$ between clusters X and Y — is described by the following expression : $D(X,Y)=\max_{x\in X,y\in Y} d(x,y)$

where

- $d(x,y)$ is the distance between elements $x\in X$ and $y\in Y$;
- X and Y are two sets of elements (clusters)

Complete linkage clustering avoids a drawback of the alternative single linkage method - the so-called *chaining phenomenon*, where clusters formed via single linkage clustering may be forced together due to single elements being close to each other, even though many of the elements in each cluster may be very distant to each other. Complete linkage tends to find compact clusters of approximately equal diameters.[4]

Contents [hide]

Naive Algorithm [edit]

The following algorithm is an agglomerative scheme that erases rows and columns in a proximity matrix as old clusters are merged into new ones. The $N\times N$ proximity matrix *D* contains all distances *d*(*i*,*j*). The clusterings are assigned sequence numbers 0,1,......, (*n* − 1) and *L*(*k*) is the level of the kth clustering. A cluster with sequence number *m* is denoted (*m*) and the proximity between clusters (*r*) and (*s*) is denoted *d*[(*r*),(*s*)].

The algorithm is composed of the following steps:

1. Begin with the disjoint clustering having level *L*(0) = 0 and sequence number m = 0.
2. Find the most similar pair of clusters in the current clustering, say pair (r), (s), according to *d*[(*r*),(*s*)] = max *d*[(*i*),(*j*)] where the maximum is over all pairs of clusters in the current clustering.
3. Increment the sequence number: *m* = *m* + 1. Merge clusters (*r*) and (*s*) into a single cluster to form the next clustering *m*. Set the level of this clustering to *L*(*m*) = *d*[(*r*),(*s*)]
4. Update the proximity matrix, *D*, by deleting the rows and columns corresponding to clusters (*r*) and (*s*)

and adding a row and column corresponding to the newly formed cluster. The proximity between the new cluster, denoted (*r*,*s*) and old cluster (*k*) is defined as *d*[(*k*), (*r*,*s*)] = **max *d*[(*k*),(*r*)], *d*[(*k*),(*s*)]**.

5. If all objects are in one cluster, stop. Else, go to step 2.

Optimally efficient algorithm [edit]

The algorithm explained above is easy to understand but of complexity $\mathcal{O}(n^3)$. In May 1976, D. Defays proposed an optimally efficient algorithm of only complexity $\mathcal{O}(n^2)$ known as CLINK (published 1977)[5] inspired by the similar algorithm SLINK for single-linkage clustering.

This section requires expansion. *(October 2011)*

Other linkages [edit]

Alternative linkage schemes include single linkage and average linkage clustering - implementing a different linkage in the naive algorithm is simply a matter of using a different formula to calculate inter-cluster distances in the initial computation of the proximity matrix and in step 4 of the above algorithm. An optimally efficient algorithm is however not available for arbitrary linkages. The formula that should be adjusted has been highlighted using bold text.

References [edit]

1. ^ T. Sorensen (1948). "A method of establishing groups of equal amplitude in plant sociology based on similarity of species and its application to analyses of the vegetation on Danish commons.". *Biologiske Skrifter* **5**: 1–34.
2. ^ Legendre, P. & Legendre, L. 1998. Numerical Ecology. Second English Edition. 853 pages.
3. ^ Brian S. Everitt; Sabine Landau; Morven Leese (2001). *Cluster Analysis* (Fourth ed.). London: Arnold. ISBN 0-340-76119-9.
4. ^ Everitt, Landau and Leese (2001), pp. 62-64.
5. ^ D. Defays (1977). "An efficient algorithm for a complete link method" (PDF). *The Computer Journal* (British Computer Society) **20** (4): 364–366. doi:10.1093/comjnl/20.4.364.

Other literature [edit]

- H. Späth (1980). *Cluster Analysis Algorithms*. Chichester: Ellis Horwood.

Categories: Data clustering algorithms