SNS COLLEGE OF ENGINEERING

Kurumbapalayam(Po), Coimbatore – 641 107
Accredited by NAAC-UGC with 'A' Grade
Approved by AICTE, Recognized by UGC & Affiliated to Anna University, Chennai

Department of Information Technology

Course Name – Data Warehouse & Mining

II Year / IV Semester

Topic – Hierarchical Method

Hierarchical Method

- Hierarchical clustering refers to an unsupervised learning procedure that determines successive clusters based on previously defined clusters.
- It works via grouping data into a tree of clusters.
- Hierarchical clustering stats by treating each data points as an individual cluster.
- The endpoint refers to a different set of clusters, where each cluster is different from the other cluster, and the objects within each cluster are the same as one another.
- There are two types of hierarchical clustering :

  Agglomerative Hierarchical Clustering
  Divisive Clustering

Agglomerative Hierarchical Clustering

- Agglomerative clustering is one of the most common types of hierarchical clustering used to group similar objects in clusters.

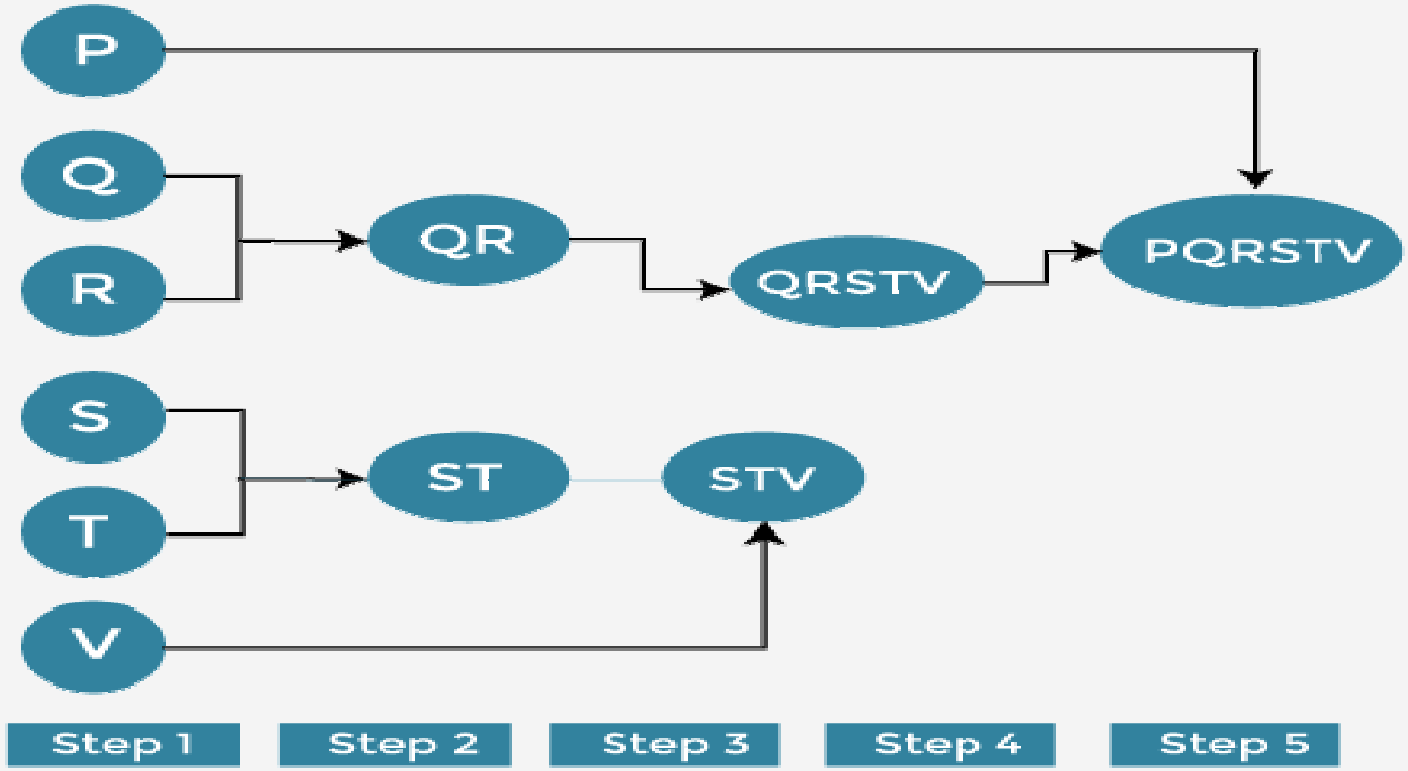
- Agglomerative clustering is also known as AGNES (Agglomerative Nesting).

- Agglomerative hierarchical clustering algorithm

1. Determine the similarity between individuals and all other clusters. (Find proximity matrix).
2. Consider each data point as an individual cluster.
3. Combine similar clusters.
4. Recalculate the proximity matrix for each cluster.
5. Repeat step 3 and step 4 until you get a single cluster.

Agglomerative Hierarchical Clustering

Let's suppose we have six different data points P, Q, R, S, T, V.

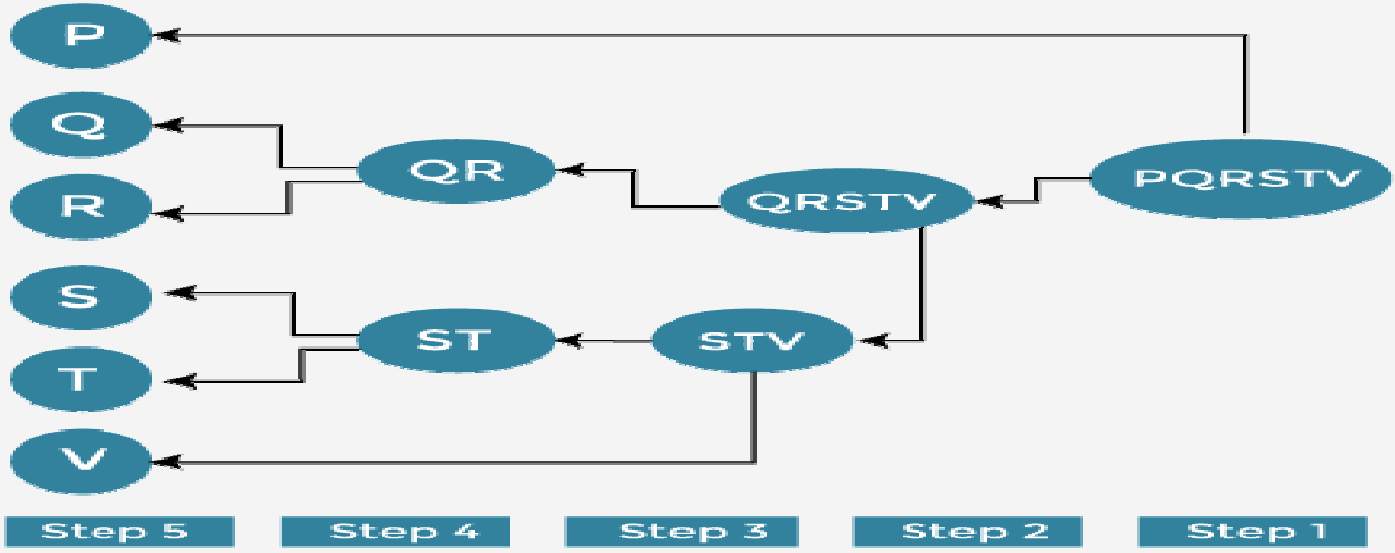


Agglomerative Hierarchical Clustering

Step 1:
Consider each alphabet (P, Q, R, S, T, V) as an individual cluster and find the distance between the individual cluster from all other clusters.
Step 2:
Now, merge the comparable clusters in a single cluster. Let's say cluster Q and Cluster R are similar to each other so that we can merge them in the second step. Finally, we get the clusters [(P), (QR), (ST), (V)]
Step 3:
Here, we recalculate the proximity as per the algorithm and combine the two closest clusters [(ST), (V)] together to form new clusters as [(P), (QR), (STV)]
Step 4:
Repeat the same process. The clusters STV and PQ are comparable and combined together to form a new cluster. Now we have [(P), (QQRSTV)].
Step 5:
Finally, the remaining two clusters are merged together to form a single cluster [(PQRSTV)]

Divisive Hierarchical Clustering

- Divisive hierarchical clustering is exactly the opposite of Agglomerative Hierarchical clustering.
- In Divisive Hierarchical clustering, all the data points are considered an individual cluster, and in every iteration, the data points that are not similar are separated from the cluster.
- The separated data points are treated as an individual cluster.
- Finally, we are left with N clusters.



Hierarchical Method

- Advantages of Hierarchical clustering

- It is simple to implement and gives the best output in some cases.
- It is easy and results in a hierarchy, a structure that contains more information.
- It does not need us to pre-specify the number of clusters.

- Disadvantages of hierarchical clustering

- It breaks the large clusters.
- It is Difficult to handle different sized clusters and convex shapes.
- It is sensitive to noise and outliers.
- The algorithm can never be changed or deleted once it was done previously.

BIRCH

- BIRCH stands for Balanced Iterative Reducing & Clustering using Hierarchy.
- It is multi-phase hierarchical clustering based on Clustering Features (CFs).
- It is designed for clustering large amount of numeric data by integrating hierarchical clustering in initial phase, called micro-clustering and other clustering methods in later phase, called macro-clustering.
- The Clustering Feature (CF) of a cluster is a 3-D vector summarizing information about clusters of objects. It is defines as,

CF = (n, LS, SS)

where n is the number of objects in the cluster, LS is the linear sum of the objects and SS is the squared sum of the objects.

Cluster's centroid, $X_0 = \frac{LS}{n}$

Cluster's radius, $R = \sqrt{\frac{\sum_{i=1}^{n}(x_i - X_0)^2}{n}} = \sqrt{\frac{n(SS) - 2LS^2 - n(LS)}{n^2}}$

Cluster's diameter, $D = \sqrt{\frac{\sum_{i=1}^{n}\sum_{j=1}^{n}(x_i - x_j)^2}{n(n-1)}} = \sqrt{\frac{2n(SS) - 2(LS)^2}{n(n-1)}}$

BIRCH

For example, consider a cluster C1={9,12,10,8,11} then CF(C1)=(5,50,510) where n=5, LS=9+12+10+8+11=50 and SS=9^2+12^2+10^2+8^2+11^2=510
Another example with 2-D objects, C2={(1,1),(2,1),(3,2)} then CF(C2)=(3,(6,4),(14,6)) where n=3,LL=(1+2+3,1+1+2)=(6,4) and SS=(1^2+2^2+3^2, 1^2+1^2+2^2)=(14,6)

Another important property of the CFs is that they are additive. That is, two disjoint clusters C1 and C2 with CFs CF1=(n1,LS2,SS1) and CF2=(n2,LS2,SS2) respectively, the CF of the cluster formed by merging C1 and C2 is given as, CF1+CF2=(n1+n2,LS1+LS2,SS1+SS2)

For example, C1={(2,5),(3,2),(4,3)} and C2={(1,1),(2,1),(3,1)} then

CF1=(3,(2+3+4,5+2+3),(2^2+3^2+4^2,5^2+2^2+3^2))=(3,(9,10),(29,38)) and
CF2=(3,(1+2+3,1+1+1),(1^2+2^2+3^2,1^2+1^2+1^2))=(3,(6,3), (14,3)) now, if C3=C1UC2 then
CF3=CF1+CF2=(6,(15,13),(43,41))

BIRCH

The Phases:
There are two primary phases of the algorithm:

Phase 1 — The algorithm scans the objects and constructs an initial in-memory CF tree, which can be viewed as multilevel compression of the data that tries to preserve the inherent clustering structure of the data.

Phase 2 — The algorithm uses a selected clustering method to cluster the leaf nodes of the CF tree.
The most important advantage of this algorithm is that its time complexity is only O(n), where n is the number of objects.

Chameleon

- dynamic modelling to determine similarity b/w pair of clusters.
- Clustering Based on
  1) how well connected objects are within a cluster.
  2) proximity of clusters.
- 2 cluster merged if their interconnectivity high & close together.
- does not depend on static cluster.
- k-nearest - neighbor graph approach.
- Relative interconnectivity:-

$$RI(C_i, C_j) = \frac{|EC_{\{C_i, C_j\}}|}{\frac{1}{2}(|EC_{C_i}| + |EC_{C_j}|)}$$

$EC_{\{C_i, C_j\}}$ → edge cut

- relative closeness:-

$$RC(C_i, C_j) = \frac{\bar{S}_{EC_{\{C_i, C_j\}}}}{\frac{|C_i|}{|C_i| + |C_j|}\bar{S}_{EC_{C_i}} + \frac{|C_j|}{|C_i| + |C_j|}\bar{S}_{EC_{C_j}}}$$

$\bar{S}_{EC_{\{C_i, C_j\}}}$ - avg weight of edges.

→ $O(n^2)$ time complexity

THANK YOU