Machine Learning 101

Rajdeep Chatterjee, Ph.D.
Amygdala AI, Bhubaneswar, India *

March 2025

Hierarchical Clustering: Agglomerative and Divisive Approaches

CSE RAIDEEP

1 Introduction

Hierarchical clustering is a method of cluster analysis that builds a hierarchy of clusters. It is divided into two main approaches:

- **Agglomerative Hierarchical Clustering**: A bottom-up approach where each data point starts as its own cluster, and pairs of clusters are merged as one moves up the hierarchy.
- **Divisive Hierarchical Clustering**: A top-down approach where all data points start in one cluster, and splits are performed recursively as one moves down the hierarchy.

1.1 Agglomerative Hierarchical Clustering

Agglomerative Hierarchical Clustering (AHC) is a bottom-up approach where each data point initially forms its own cluster. The algorithm proceeds by iteratively merging the most similar clusters until a stopping criterion is met (e.g., a single cluster remains or a predefined number of clusters is achieved).

Steps of Agglomerative Hierarchical Clustering

1. **Initialization**: Start by treating each data point as a single cluster. If there are n data points, there will be n clusters initially.
2. **Compute Proximity Matrix**: Compute the proximity (distance or similarity) between every pair of clusters.
3. **Merge Clusters**: Merge the two closest clusters based on the proximity matrix.
4. **Update Proximity Matrix**: Update the proximity matrix to reflect the new cluster.
5. **Repeat**: Repeat steps 3 and 4 until all data points are in a single cluster or a stopping criterion is met.

*Amygdala AI, is an international volunteer-run research group that advocates for *AI for a better tomorrow* `http://amygdalaai.org/`.

1.1.1 Types of Agglomerative Hierarchical Clustering

The key difference between the types of AHC lies in how the distance between clusters is computed. Common methods include:

Single Linkage

The distance between two clusters is defined as the shortest distance between any single pair of points from the two clusters:

$$d(C_1, C_2) = \min_{\mathbf{x} \in C_1, \mathbf{y} \in C_2} d(\mathbf{x}, \mathbf{y})$$

where $d(\mathbf{x}, \mathbf{y})$ is the distance between points $\mathbf{x}$ and $\mathbf{y}$.

Complete Linkage

The distance between two clusters is defined as the longest distance between any single pair of points from the two clusters:

$$d(C_1, C_2) = \max_{\mathbf{x} \in C_1, \mathbf{y} \in C_2} d(\mathbf{x}, \mathbf{y})$$

Average Linkage

The distance between two clusters is defined as the average distance between all pairs of points from the two clusters:

$$d(C_1, C_2) = \frac{1}{|C_1| \cdot |C_2|} \sum_{\mathbf{x} \in C_1} \sum_{\mathbf{y} \in C_2} d(\mathbf{x}, \mathbf{y})$$

Ward's Method

Ward's method minimizes the total within-cluster variance. The distance between two clusters is defined as the increase in the sum of squared errors (SSE) when the two clusters are merged:

$$d(C_1, C_2) = \text{SSE}(C_1 \cup C_2) - \text{SSE}(C_1) - \text{SSE}(C_2)$$

1.2 Dendrogram Example

A dendrogram is a tree diagram used to represent the arrangement of clusters produced by hierarchical clustering. Below is an example of a dendrogram.

refer to code in github.com
cserajdeep

Explanation

- The dendrogram shows the hierarchical clustering of four data points: 1, 2, 3, and 4.
- At the first level, points 1 and 2 are merged into cluster A, and points 3 and 4 are merged into cluster B.
- At the second level, clusters A and B are merged into the final cluster C.
- The dashed horizontal lines represent the distances at which the clusters are merged.
- The vertical lines represent the merging steps.

CSERAJDEEP

Copyright ©cserajdeep, 2025

2 Divisive Hierarchical Clustering

Divisive Hierarchical Clustering (DHC) is a top-down approach where all data points start in a single cluster. The algorithm proceeds by recursively splitting the clusters until each data point is in its own cluster or a stopping criterion is met.

2.1 Steps of Divisive Hierarchical Clustering

1. **Initialization**: Start with all data points in a single cluster.
2. **Compute Proximity Matrix**: Compute the proximity (distance or similarity) between every pair of data points.
3. **Split Cluster**: Split the cluster into two subclusters based on a criterion (e.g., maximizing the distance between subclusters).
4. **Repeat**: Repeat steps 2 and 3 for each subcluster until all data points are in their own clusters or a stopping criterion is met.

2.2 Comparison with Agglomerative Clustering

- **Complexity**: Agglomerative clustering is generally more computationally efficient than divisive clustering.
- **Approach**: Agglomerative clustering is bottom-up, while divisive clustering is top-down.
- **Use Cases**: Agglomerative clustering is more commonly used due to its simplicity and efficiency.

Agglomerative clustering is more widely used due to its simplicity and efficiency, while divisive clustering can be useful in specific scenarios where a top-down approach is preferred. Both methods produce a dendrogram, which provides a visual representation of the clustering process.

In brief, the concept is as follows:

Agglomerative Hierarchical Clustering (Bottom-Up)

- Each data point starts as its own cluster.
- Clusters are iteratively merged based on similarity (or distance) until all points belong to a single cluster.
- This is the most commonly used method.

Divisive Hierarchical Clustering (Top-Down)

- Starts with all data points in a single cluster.
- Splits are performed recursively until each point is its own cluster.

CSERAJDEEP

Copyright ©cserajdeep, 2025

Key Steps in Hierarchical Clustering

1. Compute the Distance Matrix

Calculate pairwise distances between data points using metrics like:

- Euclidean distance
- Manhattan distance
- Cosine similarity

2. Linkage Criteria

Determines how the distance between clusters is calculated when merging them:

- **Single Linkage:** Distance between the closest points in two clusters.
- **Complete Linkage:** Distance between the farthest points in two clusters.
- **Average Linkage:** Average distance between all points in the two clusters.
- **Centroid Linkage (ward):** Distance between the centroids (mean points) of the clusters.

Explanation of Agglomerative Clustering Code

The following Python code initializes an agglomerative clustering model using the `AgglomerativeClustering` class from the `sklearn.cluster` library:

```
cluster = AgglomerativeClustering(n_clusters=2, metric='euclidean', linkage='ward')
```

This code specifies the following parameters:

- `n_clusters = 2`
  Specifies the number of clusters to form. In this case, the data points will be grouped into two clusters.
- `metric = 'euclidean'`
  Defines the distance metric used to calculate the similarity (or dissimilarity) between points. The Euclidean distance is given by:

  $$d(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

  This is the most common distance metric used for clustering in continuous space.
- `linkage = 'ward'`
  Specifies the linkage criterion used to determine the distance between clusters. The Ward linkage minimizes the variance between clusters during merging. Formally, Ward's linkage minimizes the increase in total within-cluster variance when merging two clusters.

  If A and B are two clusters being merged, Ward's linkage minimizes:

CSERAJDEEP

Copyright ©cserajdeep, 2025

$$D(A,B) = \sum_{i \in A \cup B} \|x_i - \mu_{A \cup B}\|^2$$

where $\mu_{A \cup B}$ is the mean of the combined cluster.

In this setup:

- The model aims to form **two clusters**.
- The **Euclidean distance** is used to compute pairwise distances.
- The **Ward linkage** minimizes variance to ensure optimal cluster merging.

This combination is well-suited for balanced clusters and minimizes distortion effectively.

3. Build the Dendrogram

The dendrogram visually represents the merging (or splitting) process. The height of each merge indicates the distance (dissimilarity) between the merged clusters.

4. Cutting the Dendrogram

To form final clusters, you can cut the dendrogram at a desired height.

Advantages

- No need to predefine the number of clusters.
- Provides a clear visualization of the clustering structure through the dendrogram.
- Suitable for hierarchical data structures.

Disadvantages

- Computationally expensive (time complexity: $\mathcal{O}(n^3)$) and memory-intensive.
- Sensitive to noisy data and outliers.
- Difficult to handle very large datasets efficiently.

Applications

- **Bioinformatics:** Gene sequencing and taxonomy.
- **Marketing:** Customer segmentation.
- **Image Processing:** Grouping similar images for content-based retrieval.
- **Anomaly Detection:** Identifying outlier data points.

CSERAJDEEP

Copyright ©cserajdeep, 2025