Hierarchical Clustering

A Comprehensive Guide to Understanding Hierarchical Clustering

What is it?

Hierarchical clustering is an unsupervised machine learning algorithm that groups similar data points into clusters by building a hierarchy of clusters. Unlike other clustering methods, it creates a tree-like structure called a dendrogram that shows how clusters merge or split at different levels. This approach provides a multi-scale view of the data, allowing you to see relationships between data points at various levels of granularity.

The Big Idea

The core concept is to organize data into a nested hierarchy of clusters, either by starting with individual points and merging them together (agglomerative approach) or by starting with all points in one cluster and recursively splitting them (divisive approach). This creates a multi-level view of data relationships that captures both fine-grained and coarse-grained similarities. The resulting dendrogram visualization allows you to explore the data structure at any level of detail you choose.

How It Works (Simple Version)

1. **Step 1: Start** - Treat each data point as its own individual cluster
2. **Step 2: Find** - Identify the two closest clusters based on a distance metric
3. **Step 3: Merge** - Combine these two closest clusters into a single cluster
4. **Step 4: Repeat** - Continue merging the closest clusters until all points are in one cluster
5. **Step 5: Cut** - Choose where to "cut" the dendrogram tree to get your desired number of clusters

Key Formulas

Distance Metrics

Euclidean Distance:

$$d = \sqrt{\Sigma(x_i - y_i)^2}$$

Manhattan Distance:

$$d = \Sigma|x_i - y_i|$$

Linkage Criteria

Single Linkage:

$$d(A,B) = \min(d(a,b) : a \in A, b \in B)$$

Complete Linkage:

$$d(A,B) = \max(d(a,b) : a \in A, b \in B)$$

Average Linkage:

$$d(A,B) = (1/|A||B|) \Sigma d(a,b)$$

Pros & Cons

Advantages

- No need to specify the number of clusters upfront
- Creates intuitive dendrogram visualization
- Works well with any distance metric
- Deterministic - produces same results every time
- Captures hierarchical relationships in data

Disadvantages

- Computationally expensive: $O(n^3)$ time complexity
- Requires $O(n^2)$ space for distance matrix
- Sensitive to outliers and noise
- Cannot undo previous merges or splits
- Doesn't scale well to large datasets

When Should You Use It?

Ideal Scenarios

Hierarchical clustering is your best choice when you want to explore data at multiple granularity levels, when your dataset is relatively small (typically less than 10,000 points), when you need a visual hierarchy of relationships, when you don't know the optimal number of clusters beforehand, or when working with taxonomies or phylogenetic trees. It's particularly valuable for exploratory data analysis where understanding the structure of relationships is more important than computational efficiency.

Common Uses

- Customer Segmentation
- Gene Sequence Analysis
- Document Categorization
- Image Segmentation
- Social Network Analysis
- Taxonomy Creation
- Anomaly Detection
- Bioinformatics
- Market Research
- Pattern Recognition