



Hierarchical Clustering in Machine Learning

Last Updated : 11 Mar, 2024

In data mining and statistics, hierarchical clustering analysis is a method of clustering analysis that seeks to build a hierarchy of clusters i.e. tree-type structure based on the hierarchy.

In [machine learning](#), [clustering](#) is the [unsupervised learning](#) technique that groups the data based on similarity between the set of data. There are [different-different types of clustering algorithms](#) in machine learning.

Connectivity-based clustering: This type of clustering algorithm builds the cluster based on the connectivity between the data points. Example:
Hierarchical clustering

- **Centroid-based clustering:** This type of clustering algorithm forms around the centroids of the data points. Example: [K-Means clustering](#), [K-Mode clustering](#)
- **Distribution-based clustering:** This type of clustering algorithm is modeled generated from a particular probability distribution, and the algorithm aims to estimate the parameters of the distribution to group similar data points into clusters Example: [Gaussian Mixture Models \(GMM\)](#).
- **Density-based clustering:** This type of clustering algorithm groups together data points that are in high-density concentrations and separates points in low-concentrations regions. The basic idea is that it identifies regions in the data space that have a high density of data points and groups those points together into clusters. Example: [DBSCAN\(Density-Based Spatial Clustering of Applications with Noise\)](#).

In this article, we will discuss connectivity-based clustering algorithms i.e
Hierarchical clustering

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

Got It !

Hierarchical clustering is a connectivity-based clustering model that groups the data points together that are close to each other based on the measure of similarity or distance. The assumption is that data points that are close to each other are more similar or related than data points that are farther apart.

A dendrogram, a tree-like figure produced by hierarchical clustering, depicts the hierarchical relationships between groups. Individual data points are located at the bottom of the dendrogram, while the largest clusters, which include all the data points, are located at the top. In order to generate different numbers of clusters, the dendrogram can be sliced at various heights.

The dendrogram is created by iteratively merging or splitting clusters based on a measure of similarity or distance between data points. Clusters are divided or merged repeatedly until all data points are contained within a single cluster, or until the predetermined number of clusters is attained.

We can look at the dendrogram and measure the height at which the branches of the dendrogram form distinct clusters to calculate the ideal number of clusters. The dendrogram can be sliced at this height to determine the number of clusters.

Types of Hierarchical Clustering

Basically, there are two types of hierarchical Clustering:

1. Agglomerative Clustering
2. Divisive clustering

Hierarchical Agglomerative Clustering

It is also known as the bottom-up approach or hierarchical agglomerative clustering (HAC). A structure that is more informative than the unstructured set of clusters returned by flat clustering. This clustering algorithm does not require us to prespecify the number of clusters. Bottom-up algorithms treat each data as a singleton cluster at the outset and then successively agglomerate pairs of clusters until all clusters have been merged into a single

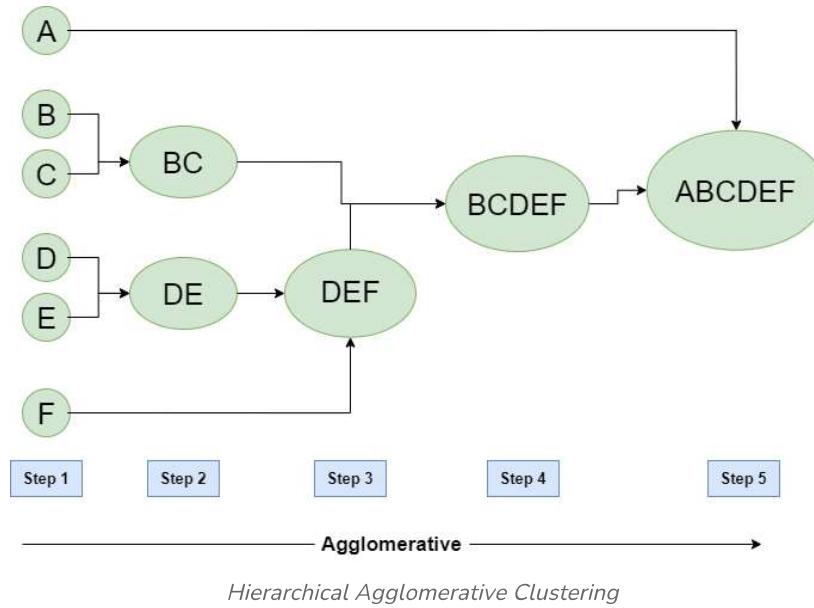
We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

Algorithm :

```

given a dataset (d1, d2, d3, ....dN) of size N
# compute the distance matrix
for i=1 to N:
    # as the distance matrix is symmetric about
    # the primary diagonal so we compute only lower
    # part of the primary diagonal
    for j=1 to i:
        dis_mat[i][j] = distance[di, dj]
each data point is a singleton cluster
repeat
    merge the two cluster having minimum distance
    update the distance matrix
until only a single cluster remains

```



Steps:

- Consider each alphabet as a single cluster and calculate the distance of one cluster from all the other clusters.
- In the second step, comparable clusters are merged together to form a single cluster. Let's say cluster (B) and cluster (C) are very similar to each other therefore we merge them in the second step similarly to cluster (D) and (E) and at last, we get the clusters [(A), (BC), (DE), (F)]

(DEF)]

- Repeating the same process; The clusters DEF and BC are comparable and merged together to form a new cluster. We're now left with clusters [(A), (BCDEF)].
- At last, the two remaining clusters are merged together to form a single cluster [(ABCDEF)].

Python implementation of the above algorithm using the scikit-learn library:

Python3

```
from sklearn.cluster import AgglomerativeClustering
import numpy as np

# randomly chosen dataset
X = np.array([[1, 2], [1, 4], [1, 0],
              [4, 2], [4, 4], [4, 0]])

# here we need to mention the number of clusters
# otherwise the result will be a single cluster
# containing all the data
clustering = AgglomerativeClustering(n_clusters=2).fit(X)

# print the class labels
print(clustering.labels_)
```

Output :

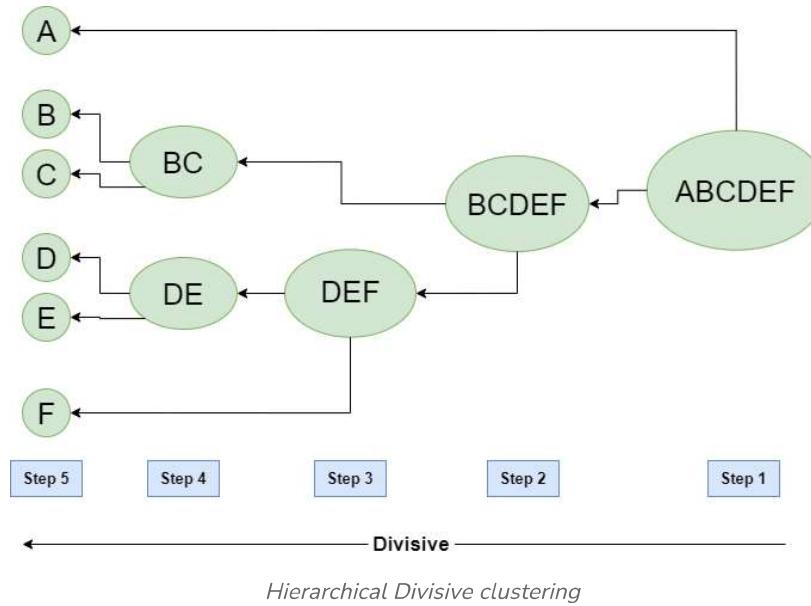
[1, 1, 1, 0, 0, 0]

Hierarchical Divisive clustering

It is also known as a top-down approach. This algorithm also does not require to prespecify the number of clusters. Top-down clustering requires a method for splitting a cluster that contains the whole data and proceeds by splitting clusters recursively until individual data have been split into singleton clusters.

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

given a dataset ($d_1, d_2, d_3, \dots, d_N$) of size N
 at the top we have all data in one cluster
 the cluster is split using a flat clustering method eg. K-Means etc
repeat
 choose the best cluster among all the clusters to split
 split that cluster by the flat clustering algorithm
until each data is in its own singleton cluster



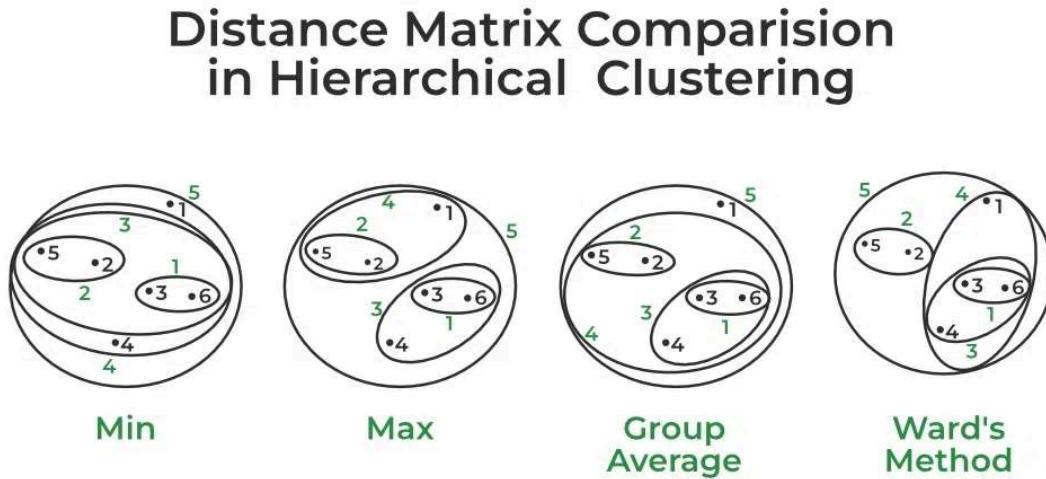
Computing Distance Matrix

While merging two clusters we check the distance between two every pair of clusters and merge the pair with the least distance/most similarity. But the question is how is that distance determined. There are different ways of defining Inter Cluster distance/similarity. Some of them are:

1. Min Distance: Find the minimum distance between any two points of the cluster.
2. Max Distance: Find the maximum distance between any two points of the cluster.
3. Group Average: Find the average distance between every two points of the clusters.
4. Ward's Method: The similarity of two clusters is based on the increase in

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

For example, if we group a given data using different methods, we may get different results:



Distance Matrix Comparision in Hierarchical Clustering

Implementations code

Python3

```

import numpy as np
from scipy.cluster.hierarchy import dendrogram, linkage
import matplotlib.pyplot as plt

# randomly chosen dataset
X = np.array([[1, 2], [1, 4], [1, 0],
              [4, 2], [4, 4], [4, 0]])

# Perform hierarchical clustering
Z = linkage(X, 'ward')

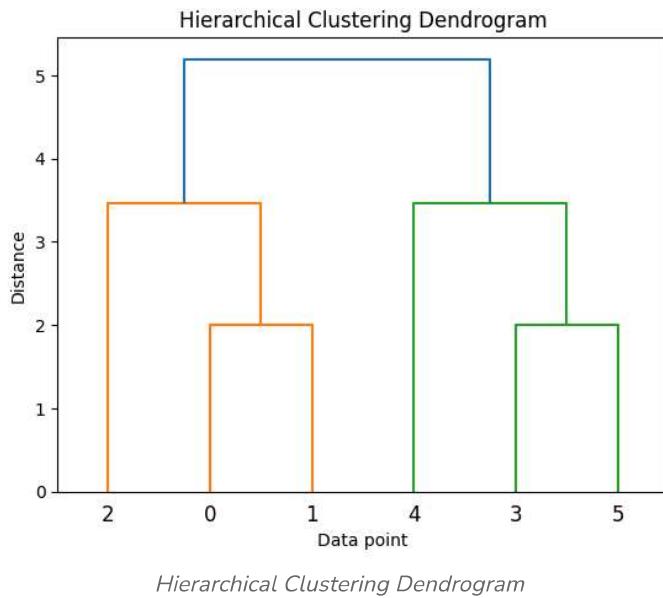
# Plot dendrogram
dendrogram(Z)

plt.title('Hierarchical Clustering Dendrogram')
plt.xlabel('Data point')
plt.ylabel('Distance')
plt.show()

```

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

Output:



Hierarchical Agglomerative vs Divisive Clustering

- Divisive clustering is more *complex* as compared to agglomerative clustering, as in the case of divisive clustering we need a flat clustering method as “subroutine” to split each cluster until we have each data having its own singleton cluster.
- Divisive clustering is more *efficient* if we do not generate a complete hierarchy all the way down to individual data leaves. The time complexity of a naive agglomerative clustering is $O(n^3)$ because we exhaustively scan the $N \times N$ matrix `dist_mat` for the lowest distance in each of $N-1$ iterations. Using priority queue data structure we can reduce this complexity to $O(n^2 \log n)$. By using some more optimizations it can be brought down to $O(n^2)$. Whereas for divisive clustering given a fixed number of top levels, using an efficient flat algorithm like K-Means, divisive algorithms are linear in the number of patterns and clusters.
- A divisive algorithm is also more *accurate*. Agglomerative clustering makes decisions by considering the local patterns or neighbor points without initially taking into account the global distribution of data. These early decisions cannot be undone. whereas divisive clustering takes into consideration the global distribution of data when making top level

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

D Debo... + Follow



24

Previous Article

ML | Expectation-Maximization Algorithm

Next Article

Implementing Agglomerative Clustering using Sklearn

Similar Reads

Difference between Hierarchical and Non Hierarchical Clustering

Hierarchical Clustering: Hierarchical clustering is basically an unsupervised clustering technique which involves creating clusters in a predefined order. The...

2 min read

Hierarchical Clustering in R Programming

Hierarchical clustering in R Programming Language is an Unsupervised non-linear algorithm in which clusters are created such that they have a hierarchy(or ...

3 min read

Difference between K means and Hierarchical Clustering

k-means is method of cluster analysis using a pre-specified no. of clusters. It requires advance knowledge of 'K'. Hierarchical clustering also known as...

2 min read

Hierarchical clustering using Weka

In this article, we will see how to utilize the Weka Explorer to perform hierarchical analysis. The sample data set for this example is based on iris data i...

3 min read

Structured vs Unstructured Ward in Hierarchical Clustering Using Scikit Learn

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

Creating Heatmaps with Hierarchical Clustering

Before diving into our actual topic, let's have an understanding of Heatmaps and Hierarchical Clustering. Heatmaps are a powerful data visualization...

9 min read

Hierarchical Clustering in Data Mining

A Hierarchical clustering method works via grouping data into a tree of clusters. Hierarchical clustering begins by treating every data point as a separate cluster....

5 min read

Hierarchical Density-Based Spatial Clustering of Applications with Noise...

Clustering is a machine-learning technique that divides data into groups, or clusters, based on similarity. By putting similar data points together and...

6 min read

Hierarchical Clustering with Scikit-Learn

Hierarchical clustering is a popular method in data science for grouping similar data points into clusters. Unlike other clustering techniques like K-means,...

4 min read

Getting started with Machine Learning || Machine Learning Roadmap

Machine Learning (ML) represents a branch of artificial intelligence (AI) focused on enabling systems to learn from data, uncover patterns, and autonomously...

11 min read

Article Tags :

AI-ML-DS

Machine Learning

AI-ML-DS With Python

Practice Tags :

Machine Learning



Corporate & Communications Address:- A-143, 9th Floor, Sovereign Corporate Tower, Sector- 136, Noida, Uttar Pradesh (201305)
 | Registered Address:- K 061, Tower K, Gulshan Vivante Apartment, Sector 137, Noida, Gautam Buddh Nagar, Uttar Pradesh, 201305



Company

- [About Us](#)
- [Legal](#)
- [Careers](#)
- [In Media](#)
- [Contact Us](#)
- [Advertise with us](#)
- [GFG Corporate Solution](#)
- [Placement Training Program](#)

Explore

- [Job-A-Thon Hiring Challenge](#)
- [Hack-A-Thon](#)
- [GfG Weekly Contest](#)
- [Offline Classes \(Delhi/NCR\)](#)
- [DSA in JAVA/C++](#)
- [Master System Design](#)
- [Master CP](#)
- [GeeksforGeeks Videos](#)
- [Geeks Community](#)

Languages

- [Python](#)
- [Java](#)
- [C++](#)
- [PHP](#)
- [GoLang](#)
- [SQL](#)
- [R Language](#)
- [Android Tutorial](#)

DSA

- [Data Structures](#)
- [Algorithms](#)
- [DSA for Beginners](#)
- [Basic DSA Problems](#)
- [DSA Roadmap](#)
- [DSA Interview Questions](#)
- [Competitive Programming](#)

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

Data Science For Beginner

CSS

Machine Learning

JavaScript

ML Maths

TypeScript

Data Visualisation

ReactJS

Pandas

NextJS

NumPy

NodeJS

NLP

Bootstrap

Deep Learning

Tailwind CSS

Python Tutorial

Python Programming Examples

Computer Science

GATE CS Notes

Django Tutorial

Operating Systems

Python Projects

Computer Network

Python Tkinter

Database Management System

Web Scraping

Software Engineering

OpenCV Tutorial

Digital Logic Design

Python Interview Question

Engineering Maths

DevOps

System Design

Git

High Level Design

AWS

Low Level Design

Docker

UML Diagrams

Kubernetes

Interview Guide

Azure

Design Patterns

GCP

OOAD

DevOps Roadmap

System Design Bootcamp

Interview Questions

School Subjects

Commerce

Mathematics

Accountancy

Physics

Business Studies

Chemistry

Economics

Biology

Management

Social Science

HR Management

English Grammar

Finance

Income Tax

Databases

Preparation Corner

SQL

Company-Wise Recruitment Process

MYSQL

Resume Templates

PostgreSQL

Aptitude Preparation

PL/SQL

Puzzles

MongoDB

Company-Wise Preparation

Companies

Colleges

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).

UPSC
SSC CGL
SBI PO
SBI Clerk
IBPS PO
IBPS Clerk

Product Management
Project Management
Linux
Excel
All Cheat Sheets
Recent Articles

Free Online Tools

Typing Test
Image Editor
Code Formatters
Code Converters
Currency Converter
Random Number Generator
Random Password Generator

Write & Earn

Write an Article
Improve an Article
Pick Topics to Write
Share your Experiences
Internships

@GeeksforGeeks, Sanchaya Education Private Limited, All rights reserved

We use cookies to ensure you have the best browsing experience on our website. By using our site, you acknowledge that you have read and understood our [Cookie Policy](#) & [Privacy Policy](#).