Clustering: Hierarchical Clustering, K-Means, and DBSCAN

Prepared by Luu Minh Sao Khue

Introduction

Clustering is an unsupervised machine learning technique used to group data points into clusters such that points in the same cluster are more similar to each other than to those in different clusters. This document covers three popular clustering algorithms: hierarchical clustering, k-means, and DBSCAN.

1 Hierarchical Clustering

Hierarchical clustering builds a hierarchy of clusters either through a bottom-up (agglomerative) approach or a top-down (divisive) approach.

1.1 Agglomerative Clustering

Agglomerative clustering starts with each data point as a single cluster and merges clusters iteratively based on a similarity criterion until all points belong to one cluster or a stopping condition is met.

Steps:

1. Start with N clusters, where each cluster contains one data point. 2. Compute the distance between all pairs of clusters (e.g., Euclidean distance). 3. Merge the two closest clusters. 4. Update the distance matrix to reflect the new cluster. 5. Repeat steps 2-4 until the desired number of clusters is reached or all points form one cluster.

Mathematical Formulation:

- **Distance Measures:**

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

$$d_{\text{manhattan}}(x, y) = \sum_{i=1}^{n} |x_i - y_i|.$$

- **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:* $d(A, B) = \frac{1}{|A||B|} \sum_{a \in A} \sum_{b \in B} d(a, b)$.

Applications:

Hierarchical clustering is often used in fields like bioinformatics, social network analysis, and market research to uncover hierarchical relationships among data points.

1.2 Divisive Clustering

Divisive clustering starts with all data points in a single cluster and recursively splits them into smaller clusters.

Steps:

1. Start with all points in a single cluster. 2. Compute the distance matrix for points within the cluster. 3. Split the cluster into two smaller clusters based on a criterion (e.g., maximizing inter-cluster distance). 4. Repeat steps 2-3 until each point is its own cluster or a stopping condition is met (e.g., a desired number of clusters).

Advantages and Disadvantages:

- **Advantages:** Can yield a global view of the clustering structure early in the process.
- **Disadvantages:** Computationally expensive as it requires recalculating distances at every step.

2 K-Means Clustering

K-means clustering aims to partition data into k clusters by minimizing the variance within each cluster.

2.1 Steps:

1. Initialize k cluster centroids randomly or using a heuristic (e.g., k-means++). 2. Assign each data point to the nearest centroid. 3. Recalculate the centroids as the mean of points assigned to each cluster:

$$\mu_j = \frac{1}{|C_j|} \sum_{x \in C_j} x,$$

where C_j is the set of points in cluster j. 4. Repeat steps 2-3 until centroids stabilize or a stopping criterion (e.g., maximum iterations) is met.

2.2 Mathematical Formulation:

K-means minimizes the following objective function:

$$J = \sum_{j=1}^{k} \sum_{x \in C_j} \|x - \mu_j\|^2.$$

Here, $\|x - \mu_j\|$ is the Euclidean distance between a point x and the cluster centroid μ_j.

2.3 Strengths and Weaknesses:

- **Strengths:** Simple to implement, efficient for large datasets.
- **Weaknesses:** Sensitive to the initial selection of centroids, struggles with non-spherical clusters and outliers.

2.4 Applications:

K-means is commonly used in image compression, customer segmentation, and anomaly detection.

3 DBSCAN (Density-Based Spatial Clustering of Applications with Noise)

DBSCAN is a density-based clustering algorithm that groups points based on their density and can identify noise points.

3.1 Definitions:

- **Epsilon (ε):** The radius of the neighborhood around a point.
- **MinPts:** Minimum number of points required to form a dense region.
- **Core Point:** A point with at least MinPts points within its ε-neighborhood.
- **Border Point:** A point that is not a core point but lies within the ε-neighborhood of a core point.
- **Noise Point:** A point that is neither a core point nor a border point.

3.2 Steps:

1. Select an unvisited point. 2. If the point is a core point, form a cluster by including all density-reachable points. 3. Mark the point as visited. 4. Repeat until all points are visited.

3.3 Mathematical Formulation:

- **ε-Neighborhood:**
$$N_\varepsilon(x) = \{y \in \mathbb{R}^n : \|x - y\| \leq \varepsilon\}.$$
- **Density Reachability:** Point y is density-reachable from x if there is a chain of points $x_1, x_2, \ldots, x_k$ where $x_1 = x$ and $x_k = y$, such that $x_{i+1} \in N_\varepsilon(x_i)$.

3.4 Strengths and Weaknesses:

- **Strengths:** Identifies clusters of arbitrary shape, robust to noise and outliers.
- **Weaknesses:** Performance depends on the choice of ε and MinPts, struggles with varying density clusters.

3.5 Applications:

DBSCAN is widely used in geospatial data analysis, pattern recognition, and fraud detection.

4 Comparison of Clustering Algorithms

Feature	**Hierarchical Clustering**	**K-Means**	
Data Assumptions	None	Spherical clusters	
Scalability	Poor for large datasets	Efficient for large datasets	
Robustness to Outliers	Moderate	Poor	
Cluster Shapes	Any shape	Spherical	
Input Parameters	Linkage criteria	k (number of clusters)	
Applications	Bioinformatics, social networks	Image compression, segmentation	Geospa

Table 1: Comparison of Clustering Algorithms

Conclusion

These clustering methods provide different perspectives on grouping data. Hierarchical clustering builds a tree-like structure, k-means minimizes variance within clusters, and DBSCAN identifies clusters based on density.