Project Algorithms

Rishav Das

I. Clustering Algorithms

This section describes three clustering algorithms: K-means, Divisive Hierarchical Clustering, and DBSCAN.

A. K-means Algorithm

The K-means clustering algorithm is a partition-based method that minimizes intra-cluster variance. The detailed steps are presented below:

Algorithm 1 K-means Clustering Algorithm

Require: Dataset $X = \{x_1, x_2, \dots, x_n\} \subseteq R^d$, number of clusters k

Ensure: Cluster centroids $C = \{c_1, c_2, \dots, c_k\} \subseteq R^d$ and cluster assignments $\{S_1, S_2, \dots, S_k\}$

1: **Initialization:** Randomly initialize k centroids:

$$C^{(0)} = \{c_1^{(0)}, c_2^{(0)}, \dots, c_k^{(0)}\}$$

where $c_j^{(0)} \in R^d$ for $j = 1, 2, \dots, k$.

2: **repeat**

3: **Assignment Step:** Assign each data point x_i to the cluster of the nearest centroid:

$$S_j^{(t)} = \{x_i \in X : \|x_i - c_j^{(t)}\|^2 \leq \|x_i - c_l^{(t)}\|^2, \forall l \neq j\}$$

4: **Update Step:** Recompute the centroid of each cluster:

$$c_j^{(t+1)} = \frac{1}{|S_j^{(t)}|} \sum_{x_i \in S_j^{(t)}} x_i$$

5: **until** Convergence: The centroids stabilize or the change in the objective function:

$$J(C^{(t)}) = \sum_{j=1}^{k} \sum_{x_i \in S_j^{(t)}} \|x_i - c_j^{(t)}\|^2$$

is below a threshold ϵ.

6: **return** Final centroids $C^{(t+1)}$ and cluster assignments $\{S_1^{(t+1)}, S_2^{(t+1)}, \dots, S_k^{(t+1)}\}$.

B. Divisive Hierarchical Clustering Algorithm

Divisive hierarchical clustering starts with all data points in one cluster and recursively splits them into smaller clusters until each point forms its own cluster.

C. DBSCAN Algorithm

Density-Based Spatial Clustering of Applications with Noise (DBSCAN) identifies clusters as dense regions of points separated by sparse regions. The updated algorithm includes detailed equations.

Algorithm 2 Divisive Hierarchical Clustering Algorithm

Require: Dataset $X = \{x_1, x_2, \dots, x_n\}$, stopping criteria (e.g., number of clusters k or distance threshold δ)

Ensure: Dendrogram representing the hierarchy of clusters

1: **Initialization:** Start with a single cluster containing all points:

$$S = \{X\}$$

2: **while** Stopping criteria not met **do**

3: Select the cluster S_i with the largest intra-cluster variance:

$$\text{Variance}(S_i) = \frac{1}{|S_i|} \sum_{x_p \in S_i} \|x_p - \mu_i\|^2, \quad \mu_i = \frac{1}{|S_i|} \sum_{x_p \in S_i} x_p$$

4: Split S_i into two clusters $\{S_i^1, S_i^2\}$ by maximizing inter-cluster distance:

$$d(S_i^1, S_i^2) = \min_{x_p \in S_i^1, x_q \in S_i^2} \|x_p - x_q\|$$

5: Add $\{S_i^1, S_i^2\}$ to the set of clusters S and remove S_i.

6: **end while**

7: **return** Dendrogram showing the hierarchy of splits.

Algorithm 3 DBSCAN Algorithm

Require: Dataset $X = \{x_1, x_2, \ldots, x_n\}$, neighborhood radius ϵ, minimum points $MinPts$

Ensure: Clusters $\{C_1, C_2, \ldots, C_k\}$ and noise points

Initialization: Mark all points as unvisited.
for each point $x_i \in X$ **do**
  if x_i is unvisited **then**
    Mark x_i as visited.
    Compute the ϵ-neighborhood:

$$N_\epsilon(x_i) = \{x_j \in X : \|x_i - x_j\| \leq \epsilon\}$$

    if $|N_\epsilon(x_i)| < MinPts$ **then**
      Mark x_i as noise.
    else
      Initialize a new cluster C and add x_i to C.
      Expand the cluster C:
      while there are unvisited points $x_j \in N_\epsilon(x_i)$ **do**
        Mark x_j as visited.
        If $|N_\epsilon(x_j)| \geq MinPts$, merge $N_\epsilon(x_j)$ with C.
      end while
    end if
  end if
end for
Define the objective function for cluster density:

$$D(C) = \sum_{x_i \in C} \sum_{x_j \in N_\epsilon(x_i)} \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right)$$

return Clusters $\{C_1, C_2, \ldots, C_k\}$ and noise points.