DBSTREAM: Density-Based Stream Clustering

Benjamin Ocampo

August 1, 2024

Abstract

DBSTREAM (Density-Based Stream Clustering with Evolving Centers and Core Micro-Clusters) is an advanced clustering algorithm designed for data streams. It efficiently identifies clusters in dynamically evolving data by maintaining micro-clusters and adapting to changes in the data distribution. This document provides a mathematical explanation of the DBSTREAM algorithm and references key papers in the field.

1 Introduction

Clustering data streams is a challenging task due to the continuous flow of data and the need for real-time analysis. Traditional clustering algorithms are often inadequate for data streams because they assume a static dataset. DBSTREAM is a density-based algorithm that addresses these challenges by maintaining a set of evolving micro-clusters and adapting to the data stream dynamically.

DBSTREAM builds on concepts from DBSCAN (Density-Based Spatial Clustering of Applications with Noise) and extends them to handle streaming data. It is designed to work with a continuous inflow of data, making it suitable for applications such as network monitoring, intrusion detection, and real-time recommendation systems.

2 Micro-Cluster Representation

In DBSTREAM, the data is summarized using micro-clusters. Each micro-cluster MC_i is characterized by the following parameters:

- C_i: The center of the micro-cluster, representing the mean position of the data points within the cluster.
- r_i: The radius of the micro-cluster, indicating the spread of data points around the center.
- d_i: The density of the micro-cluster, defined as the number of points within the micro-cluster.

- t_i: The timestamp of the last update to the micro-cluster, used for decay calculations.

Mathematically, a micro-cluster MC_i can be represented as:

$$MC_i = (C_i, r_i, d_i, t_i) \tag{1}$$

3 Micro-Cluster Update

When a new data point p_t arrives at time t, the densities of existing micro-clusters are first adjusted using a decay function to reflect the passage of time. This decay is calculated as:

$$d_i(t) = d_i(\text{last\_update}) \cdot e^{-\lambda \cdot (t - t_{\text{last\_update}})} \tag{2}$$

where λ is the decay rate, ensuring that older clusters gradually lose their influence unless reinforced by new data.

Following the decay adjustment, the algorithm checks if the new point p_t is within the radius r_i of any micro-cluster:

$$C_i(t+1) = \frac{d_i(t) \cdot C_i(t) + p_t}{d_i(t) + 1} \tag{3}$$

$$d_i(t+1) = d_i(t) + 1 \tag{4}$$

$$r_i(t+1) = \max(r_i(t), \text{distance}(p_t, C_i(t+1))) \tag{5}$$

If no suitable micro-cluster is found, a new one is created with the new point as its center, an initial radius, and a starting density, all marked with the current time as their last update time.

This mechanism ensures that the clustering model continuously adapts to the latest data while the impact of older, less relevant data diminishes over time.

4 Cluster Formation

Clusters are formed by grouping micro-clusters based on their density. In DBSTREAM, unlike some traditional clustering algorithms, the volume V_i of a micro-cluster does not play a direct role in the calculation of its density for clustering decisions. Instead, density is considered in terms of point count within the micro-cluster relative to its spatial extent, defined by its radius.

The density of a micro-cluster MC_i at any given time t is traditionally calculated as:

$$d_i = \frac{d_i(t)}{V_i} \tag{6}$$

where V_i is often conceptualized as the volume of the n-dimensional sphere defined by the micro-cluster's radius:

$$V_i = \frac{\pi^{n/2}}{\Gamma(n/2+1)} r_i^n \tag{7}$$

for a micro-cluster in an n-dimensional space. Here, Γ represents the Gamma function, which extends the factorial to real and complex numbers.

However, DBSTREAM simplifies this approach by focusing more on the radius and count of points, adjusting cluster parameters without explicitly calculating the volume. This approach emphasizes the adaptability and efficiency of handling data streams where computational simplicity and speed are crucial.

Micro-clusters with densities above a threshold are considered core micro-clusters. These core micro-clusters are then connected based on their proximity to form larger clusters. Non-core micro-clusters that are within a certain distance of core micro-clusters are assigned to the nearest core cluster, based on their geographical closeness and the density criterion.

This method allows DBSTREAM to dynamically adjust to changes in the data stream and maintain high-performance clustering in real-time applications.

5 Decay Function

To handle the evolving nature of data streams, DBSTREAM uses a decay function to reduce the influence of older data points. The density of a micro-cluster MC_i decays over time according to the following function:

$$d_i(t) = d_i(t) \cdot \exp(-\lambda \cdot (t - t_i)) \tag{8}$$

where λ is the decay rate and t_i is the timestamp of the last update. This decay function ensures that the algorithm adapts to changes in the data stream by gradually reducing the influence of older data points.

6 References

For further reading on the DBSTREAM algorithm and related work, refer to the following papers:

- Hahsler, M., & Bolanos, M. (2016). *Clustering Data Streams Based on Shared Density Between Micro-Clusters*. IEEE Transactions on Knowledge and Data Engineering, 28(6), 1449-1461.