A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases in Noise

Martin Ester, Hans-Peter Kriegel, Jorg Sander, Xiawei Xu

Presenter: Chihoon, Lee
20, Nov, 2000

DBSCAN

1. Introduction
2. Density Based Notion of Clusters
3. Overview of DBSCAN
4. Performance Evaluation
5. Discussion

Introduction

Spatial Databases
- Require to detect knowledge from great amount of data
- Need to handle with arbitrary shape

Requirements of Clustering in Data Mining
- Scalability
- Dealing with different types of attributes
- Discovery of Clusters with arbitrary shape
- Minimal requirements for domain knowledge to determine input parameters
- Able to deal with noise and outliers
- Insensitive to the order of input data
- High dimensionality of data
- Interpretability and usability

Introduction (cont..)

Agglomerative approach
(D min )
Ejcluster O(n^2)

K-means (Center)
K-medoids (One of Objects)
Clarans
Focusing techniques

Hierarchical
- Termination Conditions required
- Agglomerative approach
- Divisive approach

Partitioning
- Domain Knowledge required (K)

High dimensionality of data
Density Based Notion of Clusters

Terms

- \( N_{\text{eps}}(q) \), \( \text{MinPts} \)
- \( N_{\text{eps}}(q) = \{ p \in \mathbb{D} \mid \text{dist}(p,q) \leq \text{Eps} \} \)

Definitions

1. Directly Density-reachable
2. Density - reachable
3. Density - connected

Overview of DBSCAN

1. Based on Notion of Density in N-dimensional points.
   - Best working with Point data
2. \( N_{\text{eps}} \) and \( \text{MinPts} \) parameters required.
   - Empirically determined
3. Performed to discover arbitrary shape.
4. Supported by R* tree structure
   - spatial index structures
   - \( O(\log n) \)

Overview of DBSCAN (cont..)

Basic 2 steps

1. Arbitrary selection of an point
2. Retrieve all points that are density-reachable

DBSCAN(SetOfPoints, Eps, MinPts) {
  //SetOfPoints is UNCLASSIFIED
  ClusterId := nextId(NOISE);
  FOR i FROM 1 TO SetOfPoints.size DO
    Point := SetOfPoints.get(i);
    IF Point.ClId = UNCLASSIFIED THEN
      IF ExpandCluster (SetOfPoints, Point, ClusterId, Eps, MinPts) THEN
        ClusterId := nextId(ClusterId)
      END IF
    END IF
  END FOR
  //End of DBSCAN
Performance Evaluation

- Scalability ✓
- Dealing with different types of attributes
- Discovery of Clusters with arbitrary shape ✓
- Able to deal with noise and outliers ✓
- Insensitive to the order of input data
- High dimensionality of data
- Interpretability and usability ✓
- Minimum requirements for Domain knowledge to input parameters ✓

<table>
<thead>
<tr>
<th>Numbers of points</th>
<th>1252</th>
<th>2503</th>
<th>3910</th>
<th>5213</th>
<th>6256</th>
<th>7820</th>
<th>8937</th>
<th>10426</th>
<th>12512</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBSCAN</td>
<td>3.1</td>
<td>6.7</td>
<td>11.3</td>
<td>16.0</td>
<td>17.8</td>
<td>24.5</td>
<td>28.2</td>
<td>32.7</td>
<td>41.7</td>
</tr>
<tr>
<td>CLARA NS</td>
<td>758</td>
<td>3026</td>
<td>6845</td>
<td>11745</td>
<td>18029</td>
<td>29826</td>
<td>39265</td>
<td>60540</td>
<td>80638</td>
</tr>
</tbody>
</table>

Discussion

- Requires one Global parameters.

- Need to extend object types.
- High Dimensional features need to be investigated.
- Need to explore K-dist graph
- Update clusters for new data
Performance Evaluation (cont)

Overview of DBSCAN (cont..)

2.1. Adopted Heuristic to decide $N_{eps}$ and $MinPts$ parameters
    Generates K-dist graph
    Users or the system estimate percentage of noise
    Users can evaluate the selected threshold.

Start

MinPts=4

Discovered by CLARANS

Discovered by DBSCAN

K-dist

threshold point

Eps

noise

clusters

points