DATA MINING 1
Density-based Clustering

Riccardo Guidotti

Revisited slides from Lecture Notes for Chapter 7 “Introduction to Data Mining”, 2nd Edition by Tan, Steinbach, Karpatne, Kumar
What is Cluster Analysis?

• Finding groups of objects such that the objects in a group will be similar (or related) to one another and different from (or unrelated to) the objects in other groups

Intra-cluster distances are minimized

Inter-cluster distances are maximized
DBSCAN

- DBSCAN is a density-based algorithm.
  - Density = number of points within a specified radius (Eps)
  - A point is a core point if it has at least a specified number of points (MinPts) within Eps
    - These are points that are at the interior of a cluster
    - Counts the point itself
  - A border point is not a core point, but is in the neighborhood of a core point
  - A noise point is any point that is not a core point or a border point
DBSCAN: Core, Border, and Noise Points

MinPts = 7
DBSCAN Algorithm

- Eliminate noise points
- Perform clustering on the remaining points

\[ current\_cluster\_label \leftarrow 1 \]

for all core points do
    if the core point has no cluster label then
        \[ current\_cluster\_label \leftarrow current\_cluster\_label + 1 \]
        Label the current core point with cluster label \( current\_cluster\_label \)
    end if
    for all points in the \( Eps \)-neighborhood, except \( i^{th} \) the point itself do
        if the point does not have a cluster label then
            Label the point with cluster label \( current\_cluster\_label \)
        end if
    end for
end for
DBSCAN: Core, Border and Noise Points

Original Points

Point types: core, border and noise

Eps = 10, MinPts = 4
When DBSCAN Works Well

- Resistant to Noise
- Can handle clusters of different shapes and sizes
When DBSCAN Does NOT Work Well

- Varying densities
- High-dimensional data

Original Points

(MinPts=4, Eps=9.75).

(MinPts=4, Eps=9.92)
DBSCAN: Determining EPS and MinPts

• Idea is that for points in a cluster, their $k^{th}$ nearest neighbors are at roughly the same distance
• Noise points have the $k^{th}$ nearest neighbor at farther distance
• So, plot sorted distance of every point to its $k^{th}$ nearest neighbor
DBSCAN Evolution

OPTICS
When DBSCAN Works Well

- Resistant to Noise
- Can handle clusters of different shapes and sizes
When DBSCAN Does NOT Work Well

- Varying densities
- High-dimensional data
• OPTICS: Ordering Points To Identify the Clustering Structure
  • Produces a special order of the dataset wrt its density-based clustering structure.
  • This cluster-ordering contains info equivalent to the density-based clusterings corresponding to a broad range of parameter settings.
  • Good for both automatic and interactive cluster analysis, including finding intrinsic clustering structure.
  • Can be represented graphically or using visualization techniques.
OPTICS: Extension from DBSCAN

- OPTICS requires two **parameters**:
  - $\varepsilon$, which describes the maximum distance (radius) to consider,
  - $\text{MinPts}$, describing the number of points required to form a cluster

- **Core point**. A point $p$ is a core point if at least $\text{MinPts}$ points are found within its $\varepsilon$-neighborhood.

- **Core Distance**. It is the **minimum** value of radius required to classify a given point as a core point. If the given point is not a Core point, then it’s Core Distance is undefined.
OPTICS: Extension from DBSCAN

• **Reachability Distance.** The reachability distance between a point $p$ and $q$ is the **maximum** of the Core Distance of $p$ and the Distance between $p$ and $q$.

• The Reachability Distance is not defined if $q$ is not a Core point. Below is the example of the Reachability Distance.

• In other words, if $q$ is within the core distance of $p$ then use the core distance, otherwise the real distance.
OPTICS Pseudo-Code

• For each point $p$ in the dataset
  • Initialize the reachability distance of $p$ as undefined
• For each unprocessed point $p$ in the dataset
  • Get the neighbors $N$ of $p$
  • Mark $p$ as processed and output to the ordered list
  • If $p$ is a core point
    • Initialize a priority queue $Q$ to get the closest point to $p$ in terms of reachability
    • Call the function $update(N, p, Q)$
    • For each point $q$ in $Q$
      • Get the neighbors $N'$ of $q$
      • Mark $q$ as processed and output to the ordered list
        • If $q$ is a core point Call the function $update(N', q, Q)$
OPTICS Pseudo-Code

• Function \textit{update}(N, p, Q)
  • Calculate the core distance for \( p \)
  • For each neighbor \( q \) in \( N \) (update the reachability)
    • If \( q \) is not processed
      • \( new\_rd \) = reachability distance between \( p \) and \( q \)
    • If \( q \) is not in \( Q \)
      • \( Q.insert(q, new\_rd) \)
    • Else
      • If \( new\_rd < q.rd \)
        • \( Q.move\_up(q, new\_rd) \)
OPTICS Output

• OPTICS outputs the points in a particular ordering, annotated with their smallest reachability distance.

• A reachability-plot (a special kind of dendrogram), the hierarchical structure of the clusters can be obtained easily.

• x-axis: the ordering of the points as processed by OPTICS

• y-axis: the reachability distance

• Points belonging to a cluster have a low reachability distance to their nearest neighbor, the clusters show up as valleys in the reachability plot. The deeper the valley, the denser the cluster.
OPTICS Output
OPTICS Output

- Clusters are extracted
  1. by selecting a range on the x-axis after visual inspection,
  2. by selecting a threshold on the y-axis
  3. by different algorithms that try to detect the valleys by steepness, knee detection, or local maxima. Clustering obtained this way usually are hierarchical, and cannot be achieved by a single DBSCAN run.

OPTICS: The Radius Parameter

• Both core-distance and reachability-distance are undefined if no sufficiently dense cluster (w.r.t. $\varepsilon$) is available.
• Given a sufficiently large $\varepsilon$, this never happens, but then every $\varepsilon$-neighborhood query returns the entire database.
• Hence, the $\varepsilon$ parameter is required to cut off the density of clusters that are no longer interesting, and to speed up the algorithm.
• The parameter $\varepsilon$ is, strictly speaking, not necessary.
• It can simply be set to the maximum possible value.
• When a spatial index is available, however, it does play a practical role with regards to complexity.
• OPTICS abstracts from DBSCAN by removing this parameter, at least to the extent of only having to give the maximum value.
References

• Clustering. Chapter 7. Introduction to Data Mining.

• Mihael Ankerst; Markus M. Breunig; Hans-Peter Kriegel; Jörg Sander (1999). OPTICS: Ordering Points To Identify the Clustering Structure.