Density-based clustering with DBSCAN and OPTICS

Izabela Anna Wowczko
Institute of Technology Blanchardstown

Abstract

This paper presents two density-based algorithms: Density Based Spatial Clustering of Applications with Noise (DBSCAN) and Ordering Points to Identify the Clustering Structure (OPTICS). The notion of density, as well as its various estimators, is explained. We compare two methods of identifying similar objects based on their density, of which one produces clusters and the other outputs augmented ordering representing density-based structure of a database. The parameters and their optimisations are also discussed.

1. Introduction

One of the primary data mining techniques is clustering. It can be a stand-alone tool in getting insight into a database, or it can be a pre-processing step in further application of other mining operators. Clustering is a process of discovering similar objects within a database and grouping them together into meaningful clusters. The notion of a cluster is quite ambiguous, thus the process itself may result in different outcomes. This is heavily dependent on the nature of the data and the implemented technique, as well as the domain expertise employed in determining the relevance of clusters.

Various practices are best suited for different data structures. We are looking at DBSCAN and OPTICS, clustering methods applied to detecting clusters of various densities, shapes and sizes in spatial data sets with noise.

The reminder of this paper is structured as follows. Section 2 presents density-based clustering by example of DBSCAN. Section 3 discusses OPTICS in detail. Section 4 reflects on parameters with regard to optimising performance of density-based algorithms. Conclusions are summarised in section 5.

2. Density-Based Clustering with DBSCAN

Density-based clustering is one of the portioning methods, in which similar objects are grouped depending on their density within a given data set. In this approach, clusters are defined as regions in which objects are dense, separated by regions of lower concentration. The most popular density-based technique is DBSCAN. It lays the foundations for density assessment, introducing terms such as ε -neighboorhood (Nε), minimal number of objects (MinPts), core and border points, as well as various measures of reachability between two items in a database (Ester et al 1996). The same terminology, explained further in this section, is followed in other algorithms, including OPTICS.

2.1 Definitions

Clustering with density-based algorithms identifies regions in which objects (also referred to as points) are close to one another. This closeness is understood in terms of proximity (dissimilarity) measure best suited for a given set of attributes. In that respect, the variations of density in a data set are being estimated based on the number of points in a neighbourhood of every single object belonging to that data set.

Definition: ε-neighboorhood

Let *p* and *q* be two arbitrary points within a dataset. Let ε be an arbitrary radius, and Nε(*p*) be the ε-neighbourhood of a point *p*.

A point *q* belongs to the ε- neighbourhood of a point *p* if the distance between these two is not greater than ε.

Definition: MinPts, core and border points

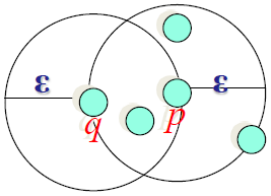
Depending on the number of objects within a given radius, the ε-neighbourhood of any point can be of high or low density. If high density constitutes a cluster and low density represents noise, a threshold value needs to be identified to distinguish between these two. This is specified by MinPts parameter, which defines the lowest number of points in ε-radius region required to form a cluster. However, every cluster contains two types of objects: core points (inside points) and border points. Core points will always have at least MinPts in their ε-neighbourhood. Border points, on the other hand, are located at the edge of a dense region and will have significantly less points around them

(Gray 2013). If a point was assessed on MinPts measure exclusively, any border point would be discarded from a cluster as, sensu stricto, its ε-neighbourhood is not sufficiently dense. To avoid that misclassification, density-based clustering supports different measures of reachability between points in a database. Roughly speaking, even though a border point is in a region considered to be of low density, it can still belong to a cluster if it is close enough to another point located in an adequately dense region. The notions of reachability as introduced by Ester et al (1996) are presented below.

Definition: direct density-reachable point, density-reachable point, density-connected point

A point *q* is directly density-reachable from a point *p* if *q* belongs to the ε -neighboorhood of *p* and *p* is a core point.

As illustrated in Figure1, the border point *q* (3 points in its ε-neighboorhood) is directly density-reachable from a core point *p* (4 points in its ε-neighboorhood). This relation would be symmetric, if both were core points.



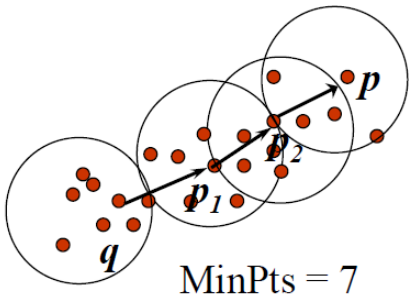
- *q* is directly density-reachable from *p*
- *p* is not directly density-reachable from *q*
- Density-reachability is asymmetric

MinPts = 4

Figure1. Directly density-reachable points.

A point *q* is density-reachable from a point *p* if there is a sequence of directly density-reachable points connecting point *q* with point *p*.

As illustrated in Figure 2, the border point *p* (6 points in its ε-neighboorhood) is density-reachable from the core point *p* (8 points in its ε-neighboorhood). In other words, the points are joined by a chain of core points. This relation would be symmetric, if both points, *p* and *q*, were core points.



- *p* is (indirectly) density-reachable from *q*
- *q* is not density-reachable from *p*

MinPts = 7

Figure 2. Density-reachable points.

A point *p* is density-connected to a point *q* if they are both density-reachable from a point *o*.

As illustrated in Figure 3, point *p* is density-connected to point *q* and that relation is always symmetric. Density connected points are always belonging to the same cluster. In fact, a cluster in DBSCAN is simply a set of density-connected points.

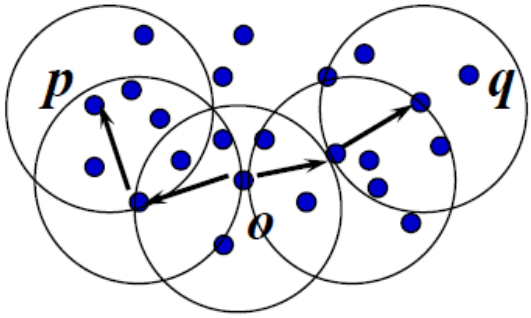


Figure 3. Density-connected points.

2.2. Clustering with DBSCAN

Based on variations in density within a data set, DBSCAN can successfully extract clusters of irregular shapes and sizes. To perform partitioning, the algorithm requires two global parameters ε and MinPts introduced in section 2.1.

As explained by Ester et al (1996), the clustering process begins with visiting an arbitrary point within a dataset. The ε-neighbourhood of the point is examined and, if the density of its ε-neighbourhood satisfies the MinPts property, the point is identified to be a core point and a cluster is started. In the next step, all density-reachable points are being retrieved and included in the cluster. If any collected point is verified to be a core point (MinPts condition is satisfied), its ε -neighbourhood also becomes part of the cluster. The logic continues until there are no more core points identified in retrieved neighbourhoods. This means that the cluster is finished and DBSCAN moves on to a new arbitrary point. At any stage, if an arbitrary is not identified to be a core point, the algorithm simply visits another object. The point itself, however, can become a border point of another cluster, should it be located in a sufficiently dense ε -neighbourhood of a core point discovered in further processing. This is possible since points are visited multiple times as candidates to different clusters in partitioning a data set.

The main disadvantage of DBSCAN is the use of global parameter values. Although optimal for clustering the entire data set, they do not reflect the structure of clusters in-depth. Real-world datasets contain regions of various densities, which form many levels of clusters. Intuitively, not all

of them are possible to identify using only global settings. OPTIC addresses that issue by outputting augmented ordering of objects in a database which represents its density-based clustering structure.

3. OPTICS

OPTIC is not a clustering technique per se as it does not output clusters. Instead, it presents intrinsic clustering structure that could otherwise be identified only in a process of repeated clustering with different parameter settings (Ankerst et al 1999). OPTICS successfully differentiates significant objects from noise, identifying all possible levels of clusters within a data set as explained in this section.

3.1. Definitions

Following the logic implemented in DBSCAN, creators of OPTICS further explore the structure of clusters in a data set. It is noted, that for the same MinPts value, clusters with larger radius (high value of ε and lower density) completely contain density-connected clusters with smaller radius (lower value of ε and higher density). With this assumption, OPTICS applies the principles of DBSCAN but the algorithm is processed for an infinite number of $\varepsilon_{i,}$ where $0 =< \varepsilon_i =< \varepsilon$. Unlike DBSCAN, however, OPTICS does not produce clusters. The output of this technique is the processing order of objects and two variables stored for each one of them: core-distance and reachability-distance. Those two values, as explained further, are sufficient information in extracting clusters for any value of $\varepsilon_i =< \varepsilon$ (Ankerst et la 1999).

Definition: core-distance of an object, reachability-distance object

For any given MinPts and ε, the core-distance of a point *o* is a minimal distance ε_i such that point *o* is a core point.

The reachability-distance of a point *p* with respect to a point *o* is the smallest distance such that *o* is a core point and *p* belongs to its ε-neighbourhood.

As illustrated in Figure 4, the core-distance of a point *o* is, in fact, the distance to its farthest MinPts neighbour. In other words, it is the smallest value of ε_i for which point *o* is a core point with respect to MinPts. Reachability-distance between two points, however, can take different measures. If a point *p* is located inside the ε_i -neighbourhood of a point *o*, the reachability-distance of a point *p* is equal to

the core-distance of a point *o*. It cannot be smaller, as core-distance ε_i is already the smallest radius for which *o* is considered to be a core point. Alternatively, if another point *q* falls outside the ε_i-neighbourhood of a point *o* (but belongs to its neighbourhood at generating distance ε), its reachability-distance is the actual distance from a point *o*.

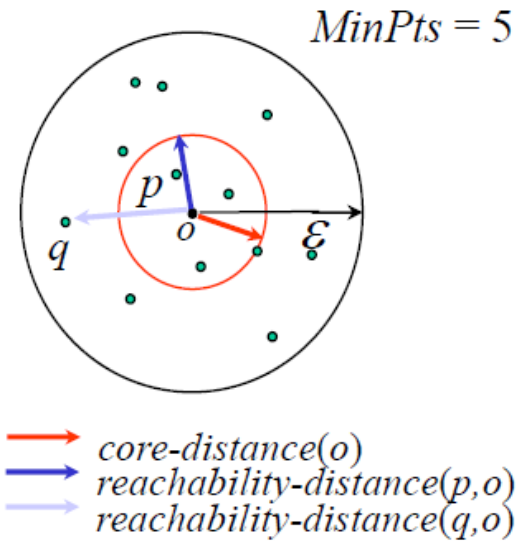


Figure 4. Core-distance and reachability-distance.

The two variables represent the relationship between points and illustrate multileveled clustering structure of a data set. The process of obtaining their values is explained in the next section.

3.2. Clustering with OPTICS

For two parameters ε and MinPts, Ankerst et al (1999) explain the performance of OPTIC in the following manner.

Firstly, a file OrderedFile is open. SetOfObjects, a database, passes an unprocessed object to a procedure ExpandClusterObject. The procedure retrieves ε-neighbourhood of the object, sets its reachability-distance to UNDEFINED (the first object processed in OPTICS is always not currently a reachable point) and calculates its core-distance. The object has been processed and is written to a file OrderedFile. If verified not to be a core object (at the generating distance ε with respect to MinPts), the next object in a database is visited. However, if the object is a core object, all directly density-reachable points are collected and written to an OrderSeeds list as candidates for further expansion of a cluster.

The OrderSeeds list in OPTICS is sorted according to the reachability-distances and it is updatable. It serves as a priority queue that guarantees hierarchical processing in such way that the closest

points are visited first and the clusters with lower value of ε_i (where $\varepsilon_i =< \varepsilon$) are finished first. A point with the smallest reachability-distance is always at the top of the list. When accessed, its ε-neighbourhood and core-distance are determined and if identified to be a core point (at the generating distance ε with respect to MinPts), its neighbours are also retrieved and written to the OrderSeeds. If any collected point is already on the list, the reachability-distances are compared and the value is corrected should the new distance be smaller (the reachability-distance is always a measure to the closest core point). The priority queue is re-sorted and the next point is being evaluated in the same manner, until all points have been processed. Then, the algorithm moves on to a new arbitrary point.

With this method, varying densities are identified without multiple processing of a database with different parameters. The process of identifying clusters with use of core- and reachability-distances is explained in the next section.

3.3 Extracting clusters

Once the density cluster-ordering has been generated, the clusters can be extracted. The supporting procedure, ExtractDBSCAN-Clustering algorithm, operates as follows.

For any given radius ε_i and MinPts, a point can belong to a current cluster, be a core point of a different cluster or constitute noise. To verify its nature, we need to look at the reachability-distance. If that value is smaller than ε_i, the object can be allocated to the cluster. If it is larger than ε_i, it cannot belong to the current cluster, as the maximal reachability-distance cannot exceed ε_i (see Figure 4). However, if verified by its core-distance, that the point is a core point with respect to ε_i and MinPts, a new cluster is started. Otherwise, the point is considered to be noise.

3.4. Clustering structure

Although, as explained in the previous section, it is possible to extract clusters by generating hierarchical cluster-ordering, the key application of OPTICS is to better understand the intrinsic density related structure of a data set (Ankerst et al1999).

The cluster-ordering is illustrated graphically as a distribution of clusters within a dataset. The main visualisation technique is reachability plot representing objects and their corresponding reachability-distance values. As seen in Figure 5, the order of processed objects is presented on the axis-x, while the reachability-distances are presented on the axis-y. Low reachability-distances indicate higher density, thus represent clusters in a form of ‘valleys’. Depending on the value of ε, more or less

clusters can be identified. This is indicated by the red bar – the higher it is placed (the higher value of ε, thus the lower density), the fewer clusters are visualised.

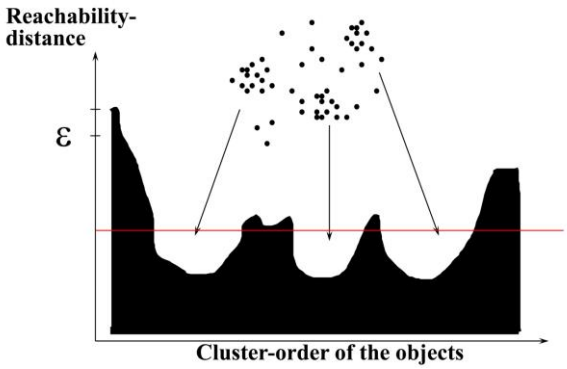


Figure 5. Reachability plot.

The influence of MinPts parameter is illustrated in Figure 6. For a constant radius ε, the structure of a clusters appears more jagged for lower value of MinPts.

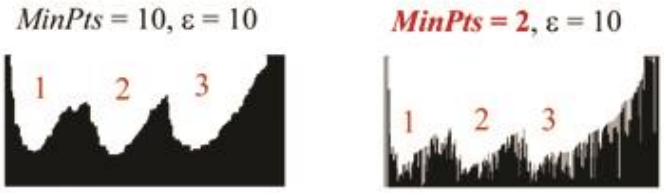


Figure 6. Reachability plot for different values of MinPts.

4. Parameter optimisation

DBSCAN and OPTICS are effective methods of density-based clustering. As presented in previous sections, they both require two parameters. The first parameter sets the minimal number of objects considered to form a cluster and is denoted as MinPts. The second parameter bounds the size of a neighbourhood in which the MinPts have to be identified and is denoted as ε.

DBSCAN offers a way of determining best parameter settings based on the values of the least dense cluster. By setting MinPts to *k* and measuring distances of an arbitrary point to its *k* nearest neighbours, the value of ε for that point is equal to the largest *k*dist. If we calculate *k*dist for all points in a data set and sort them in descending order, we can identify the maximal value of *k*dist.

This constitutes the ε value for the least dense cluster in a data set with respect to MinPts equal to *k* (Gray 2013). To achieve the best results in clustering with DBSCAN, the optimal value of $k = 4$ has been experimentally proven (Ester et al 1996).

As presented in Figure 5 and Figure 6, the reachabillity plot of OPTICS is independent of parameter settings. Although different levels of clustering might be noticeable for different values of ε and MinPts, the cluster-ordered structure of a data set itself demonstrates the same characteristics. It is, however, advisable to optimise the generating radius ε (Ankerst et al 1999). Its value influences the number of levels visualised in a reachability plot and clusters extraction with ExtractDBSCAN-Clustering as already explained in Section 3.3 and Section 3.4.

5. Conclusion

DBSCAN and OPTICS are algorithms best suited for discovering clusters of arbitrary shapes in spatial spaces with noise. Although similar methods, using the same parameters and operating in a similar manner, their approach to partitioning data set is quite different.

As noted in Figure 7, a density of a region in DBSCAN is always considered to be either high or low. This is due to a single threshold value of ε, whereas OPTICS identifies an infinite number of ε_i calculating core-distances of all objects in a dataset. Similarly, a Boolean variable identifies a relationship between points and their cluster membership in DBSCAN. In OPTICS, however, multiple reachability-distances are calculated for every single object to reflect most accurately the hierarchical structure of clusters in a data set.

	DBSCAN	**OPTICS**
Density	Boolean value (*high / low*)	Numerical value (*core-distance*)
Cluster Membership	Boolean value (*yes / no*)	Numerical value (*reachability-distance*)

Figure 7. DBSCAN vs OPTICS.

References

Ankerst, M & Breunig, MM & Kriegel, HP & Sander, J, 'OPTICS: Ordering points to identify the clustering structure', 1999, *ACM SIGMO International Conference on Management of Data*, 1999, pp. 49–60

Ester, M & Kriegel, HP & Sander, J & Xu, X, 'A density-based algorithm for discovering clusters in large spatial databases with noise', 1996, *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96),* pp. 226–231

Gray, G 'Lecture 7&8: Proximity measures & clustering', 2013, *Lecture Notes*