Density Based Spatial Clustering of Applications with Noise(DBSCAN)

Introduction

First let us define Minmum points and Epsilon
How to measure density around a point ?
We define a region around point and asses the number points with in that designated

We draw unit radius circle around a point P we define a criterion a region is considered sparse if it contains fewer than 3 points and dense if it contains 3 or more points

Minimum points parameters minimum number of points required to form a dense region which is consider a cluster
The number of samples (or total weight) in a neighborhood for a point to be considered as a core point. This includes the point itself. If min_samples is set to a higher value, DBSCAN will find denser clusters, whereas if it is set to a lower value, the found clusters will be more sparse
Epsilon Key parameter define readius of the neighborhood around a give data point is maximum distance between two points for them to be considered as part of the same neighborhood .
Examining the diagram, it’s evident that within the circle surrounding a specific point, there are only two points in addition to the point itself, totaling three points. This doesn’t meet the MinPts requirement of 4, leading us to conclude that it is not a core point.

Core Points Border Points and Noise Points
Core points
A Points considered as core points if it has minimum number of other minimum points with given radius epsilon

In the depicted diagram, with ε set to 1 and MinPts to 4, let’s focus on a specific point, P. To determine if P qualifies as a core point, we create a circle with a radius of 1 unit around P. Observing the diagram, it’s evident that point P, along with three additional points within the circle, satisfies the MinPts condition. Hence, we can confidently classify point P as a core point.

Examining the diagram, it’s evident that within the circle surrounding a specific point, there are only two points in addition to the point itself, totaling three points. This doesn’t meet the MinPts requirement of 4, leading us to conclude that it is not a core point.

Border Points

1 **not Core point**s which means a border points does meet the criteria to be a core point it has fewer han minPts with in E neigbourhood .

2 **Neighbour of core points** : A border point is within e distance of one or more core points in other words it lies on the edge of the cluster with in radius of E at least once core point
Noise point which is neihter a core poinst nor a border point
Main

Why DBSCAN:

1. Clusters can be of arbitrary shape such as those shown in the figure below.
2. Data may contain noise.

Parameters Required For DBSCAN Algorithm

1 **eps**: It defines the neighborhood around a data point i.e. if the distance between two points is lower or equal to ‘eps’ then they are considered neighbors. If the eps value is chosen too small then a large part of the data will be considered as an outlier. If it is chosen very large then the clusters will merge and the majority of the data points will be in the same clusters. One way to find the eps value is based on the *k-distance graph*.
2 **MinPts**: Minimum number of neighbors (data points) within eps radius. The larger the dataset, the larger value of MinPts must be chosen. As a general rule,

4 **Core Point**: A point is a core point if it has more than MinPts points within eps.
5 **Border Point**: A point which has fewer than MinPts within eps but it is in the neighborhood of a core point.
6 **Noise or outlier**: A point which is not a core point or border point.

Advantages of DBSCAN :

1. **Robust to outliers** : It is robust to outliers as it defines clusters based on dense regions of data, and isolated points are treated as noise.

2. **No need to specify clusters** : Unlike some clustering algorithms, DBSCAN does not require the user to specify the number of clusters beforehand, making it more flexible and applicable to a variety of datasets.

3. **Can find arbitrary shaped clusters** : DBSCAN can identify clusters with complex shapes and is not constrained by assumptions of cluster shapes, making it suitable for data with irregular structures.

4. **Only 2 hyperparameters to tune :** DBSCAN has only two primary hyperparameters to tune: “eps” (distance threshold for defining neighborhood) and “min_samples” (minimum number of points required to form a dense region). This simplicity can make parameter tuning more straightforward.

Disadvantages of DBSCAN :

1. **Sensitivity to hyperparameters** : The performance of DBSCAN can be sensitive to the choice of its hyperparameters, especially the distance threshold (eps) and the minimum number of points (min_samples). Suboptimal parameter selection may lead to under-segmentation or over-segmentation.

2. **Difficulty with varying density clusters** : DBSCAN struggles with clusters of varying densities. It may fail to connect regions with lower point density to the rest of the

cluster, leading to suboptimal cluster assignments in datasets with regions of varying densities.

3. **Does not predict** : Unlike some clustering algorithms, DBSCAN does not predict the cluster membership of new, unseen data points. Once the model is trained, it is applied to the existing dataset without the ability to generalize to new observations outside the training set.

Conclusion

Difference Between DBSCAN and K-Means.

DBSCAN	KMeans
In DBSCAN we need not specify the number of clusters.	K-Means is very sensitive to the number of clusters so it need to specified
Clusters formed in DBSCAN can be of any arbitrary shape.	Clusters formed in K-Means are spherical or convex in shape
DBSCAN can work well with datasets having noise and outliers	K-Means does not work well with outliers data. Outliers can skew the clusters in K-Means to a very large extent.
In DBSCAN two parameters are required for training the Model	In K-Means only one parameter is required is for training the model