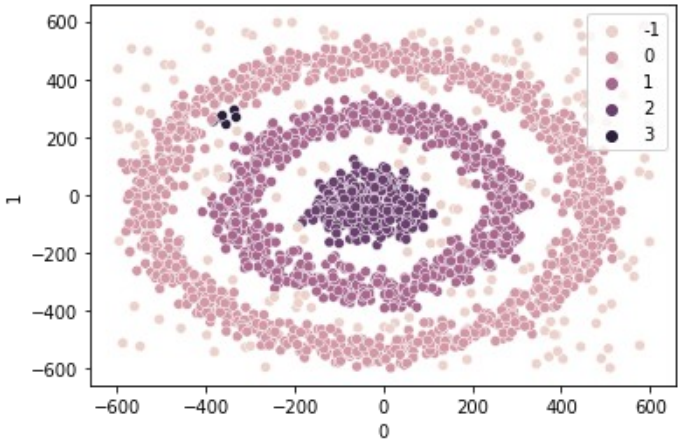
DBSCAN Clustering

DBSCAN Overview

DBSCAN (**D**ensity-**B**ased **S**patial **C**lustering of **A**pplications with **N**oise) is a clustering algorithm that works by looking for densely grouped data.

It assumes that clusters are dense regions separated by regions of lower density.

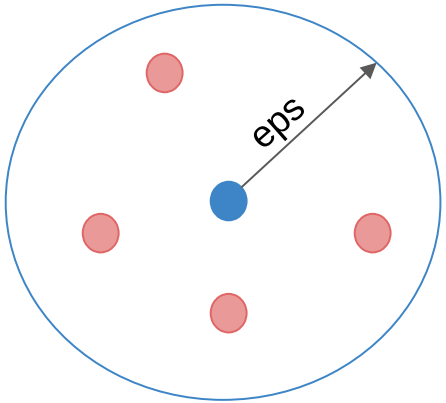
DBSCAN is very efficient in finding non-spherical shapes.



DBSCAN Parameters

The DBSCAN algorithm uses two parameters:

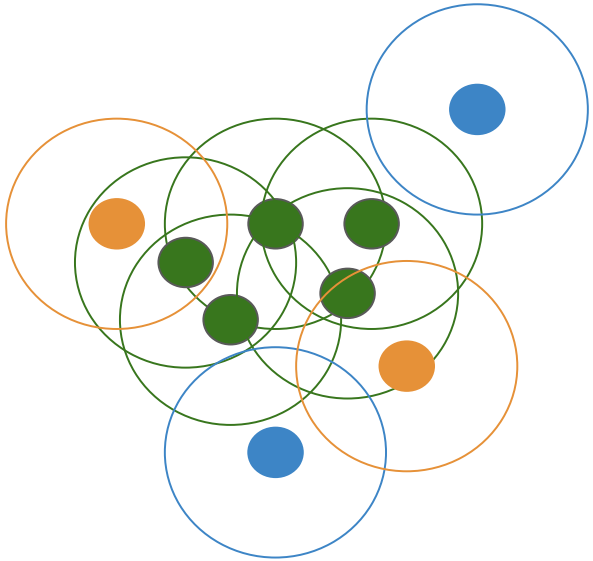
- **minPts:** The minimum number of points clustered together for a region to be considered dense.
- **eps (ε):** A distance measure used to locate the points in the neighborhood of any point.



If minPts = 4, this region would be considered dense

DBSCAN Process

1. User specifies minPts and eps.
2. An initial point is selected at random and it is determined whether the point is a core point, border point or noise point.
3. If it is a core point, all points within the neighborhood of that point become a part of its cluster. If the new points are also core points, their neighbors also join the cluster.
4. A next point that hasn't been visited yet is randomly chosen and the same process takes place.
5. The algorithm is finished when all points have been visited.



Core Point - a point that has at least minPts number of points in neighborhood
Border Point - a point that has >0 but <minPts in neighborhood
Noise Point - a point that has no points in neighborhood

Example

https://www.naftaliharris.com/blog/visualizing-dbscan-clustering/

DBSCAN vs K-Means

- DBSCAN does not require that you specify the number of clusters beforehand.
- DBSCAN is sensitive to its two parameters.
- K-means works well for spherical clusters. DBSCAN works well for non-spherical clusters.
- K-means requires that all points be assigned to a cluster. DBSCAN will assign some points as being “noise”.