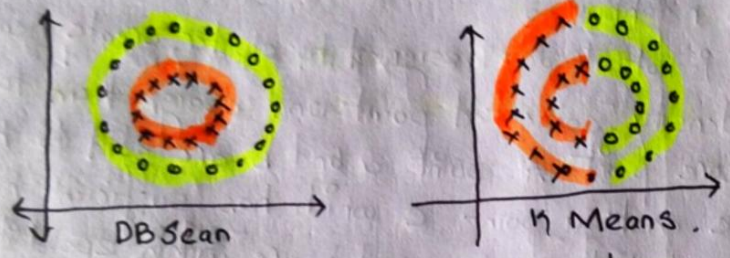
DBScan → Density based clustering

→ Clusters are dense region in the data space, seperated by regions of the lower density of points. The key idea is that for each cluster, the neighbourhood of a given radius has to contain atleast the minimum number of point.

Why DBScan?

→ Partitioning methods like K means and hierarchical clustering work for finding spherical-shaped clusters or convex clusters. In other word, they are suitable only for compact and well seperated clusters. More over, they are also severly affected by presence of noise and outliers in the data.



DBScan algorithim requires two parameters -

① eps - It defines the neighborhood around the data point i.e. if the distance between two points is lower or equal to 'eps' then they are considered as neighbours. If the eps value is choosen too small then large part of the data will be considered as outliers. If it is chosen very large then the clusters will merge and majority of the data points will be in same clusters. One way to find eps value is based on the K-distance graph.

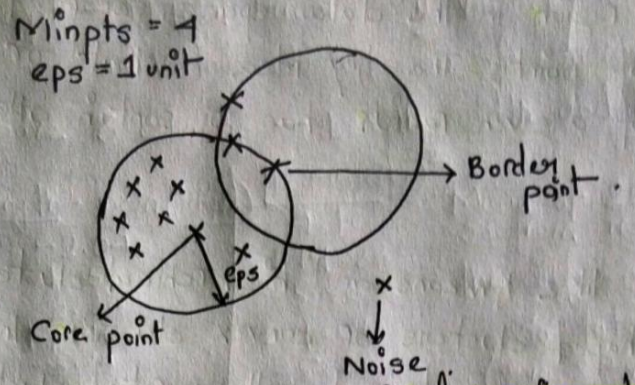
② MinPts - Minimum number of neighbors (data point) with eps radius. Larger the dataset, the larger value of MinPts should be chosen. As a general rule, the minimum MinPts can be derived from the number of dimension D in the dataset as MinPts >= D+1. The minPts value, minimum value must be choosen atleast 3.

In this algorithim, we have 3 type of data points.

① Core point → A point is a core point if it has more than MinPts point with eps.

② Border point → A point which has fewer than MinPts within eps but it is in the neighborhood of a core point.

③ Noise or outlier → A point which is not a core point / a border point.



DBScan algorithm can be abstracted in following steps -

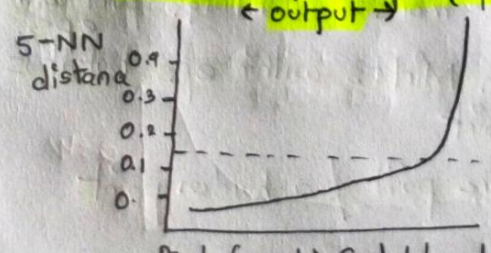
1) Find all the neighbor points within eps and identify the core points or visited with more than MinPts neighbour.
2) For each core point if it is not already assigned to cluster, create new cluster.
3) Find recursively all its density connected point and assign them to the same cluster as the core point. A point a and b are said to be density connected if there exist a point c which has sufficient number of point in its neighbour and both point a and b are within eps distance. This is a changing process. So if b is neighbour of c, c is neighbour of d, d is neighbour of e, which inturn is neighbour of a implies a is neighbour of a.
4) Iterate through the remaining unvisited point in dataset. Those point that do not belong to any cluster are noise.

Disadvantages of K Means - ① K means forms spherical clusters only. This algo fails when data is not spherical (i.e, same variance in all direction)
② K means algorithm is sensitive toward outliers. Outliers can skew the clusters in K-means to very large extent.
③ K means algo require one to specify number of cluster etc.

- DBScan algo overcome all issue, it identifies dense region by grouping together datapoint that are closed to each other based on distance measurement.

Method to determine optimal eps value - Use K distance plot, normally it is found in KNNdistplot

dbscan::KNNdistplot(df, K=5)
← output →



→ Idea is to calculate the average of distances of every points to its K Nearest neighbour.
→ K will be specified by user & correspon MinPts.
→ K distance plotted in ascending order. Aim is determine Knee (sharp change occurs)
→ It can be seen that optimal eps value is around a distance of 0.15.

DBSCAN

- Density Based Spatial Clustering of Application With Noise.
- Density based clustering algorithm make an assumption that clusters are dense region in space seperated by region of low density.
- A dense cluster is a region which is "density connected", i.e, the density of points in that region is greater than a minimum. required.

→ Since these algorithms expand clusters based on dense connectivity, they can find clusters of ~~arbitrary~~ arbitray shapes.

→ The algorithm find dense areas and expand these recursively to find dense arbitarily shaped clusters.

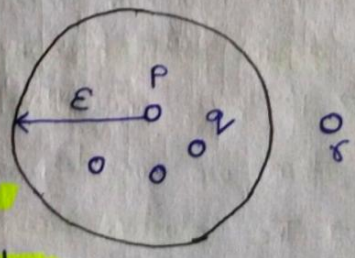
→ 2 main points to DBScan are (parameters) i) ε → Epsilon.
ii) MinPoints.

→ ε defines radius of the neighborhood region.

→ MinPoints defines the minimum number of points that should be contained in the neighborhood.

→ Since it has a concept of noise, it works well even with noise dataset.

→ Epsilon Neighborhood (N_E):
Set of all points within a distance 'ε'.



→ Core point: A point that has atleast 'minpoint' (including itself) points within its N_ε.
So, core points are 4 in the above figure.
Let the minpoint is 4, then P is the core point.

So after minpoint is found, we need to find links point. (It finds the ~~links~~ links to the core point).

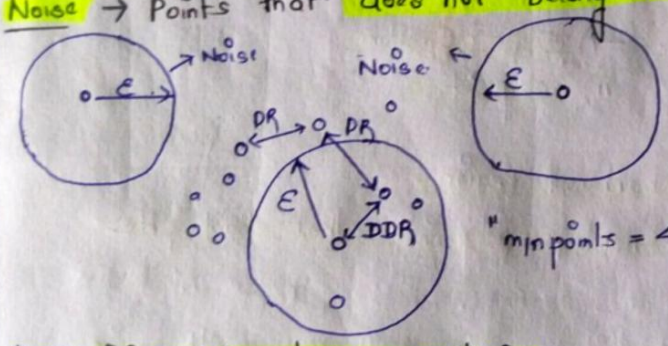
→ 2 types of link points are there → i) Direct Density Reachable (DDR)
ii) Density Reachable (DR).

→ Directly Density Reachable (DDR)
A point q is directly density reachable from a point P if p is core point and $q \in N_\epsilon$. (Refer to figure).

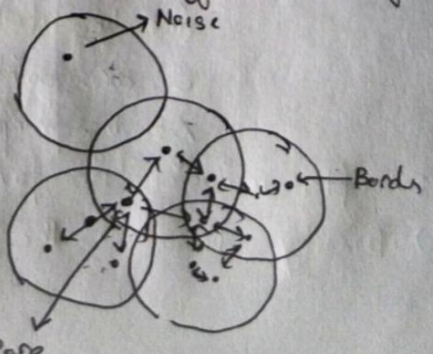
→ Density Reachable (DR).
Two points are DR if there is a chain of DDR points that links these two points. Eg → q is DDR to P.
r is DDR to q.
then r is DR to p.

Border point → Point that is DDR but not a core point. (Point that does not have sufficient min points in its N_ϵ but has atleast 'core point.')

Noise → Points that does not belong to any N_ϵ.



How DBSCAN actually work?
→ The algorithm proceeds by arbitrary picking up points in the dataset (until all the points have been visited)
→ If there are atleast 'min points' points within radius of ε to the point then we consider all these points to be part of same cluster.
→ The clusters are then expanded by recursively repeating to the neighborhood calculation for each neighboring points.
→ The complexity of this algorithm is $O(n^2)$ where n is the number of points



ADVANTAGE OF DB SCAN →
→ Can handle outlier easily.
DISADVANTAGE OF DBSCAN →
→ Struggle with high dimensional data.