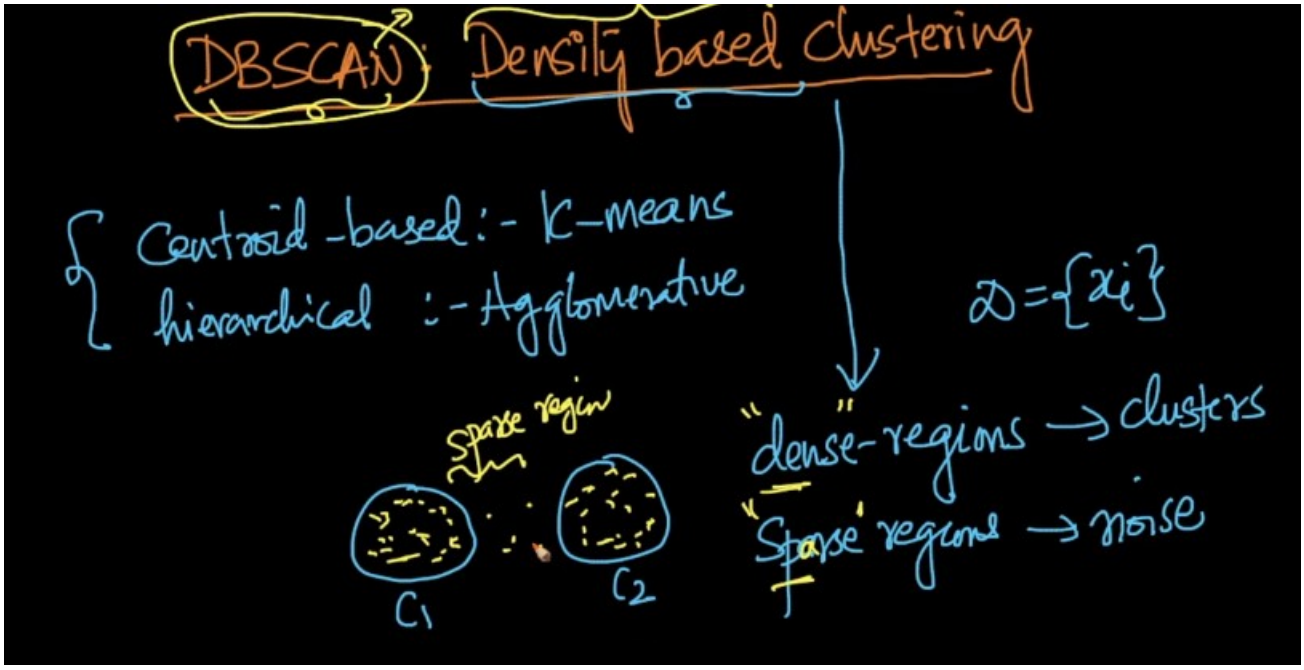
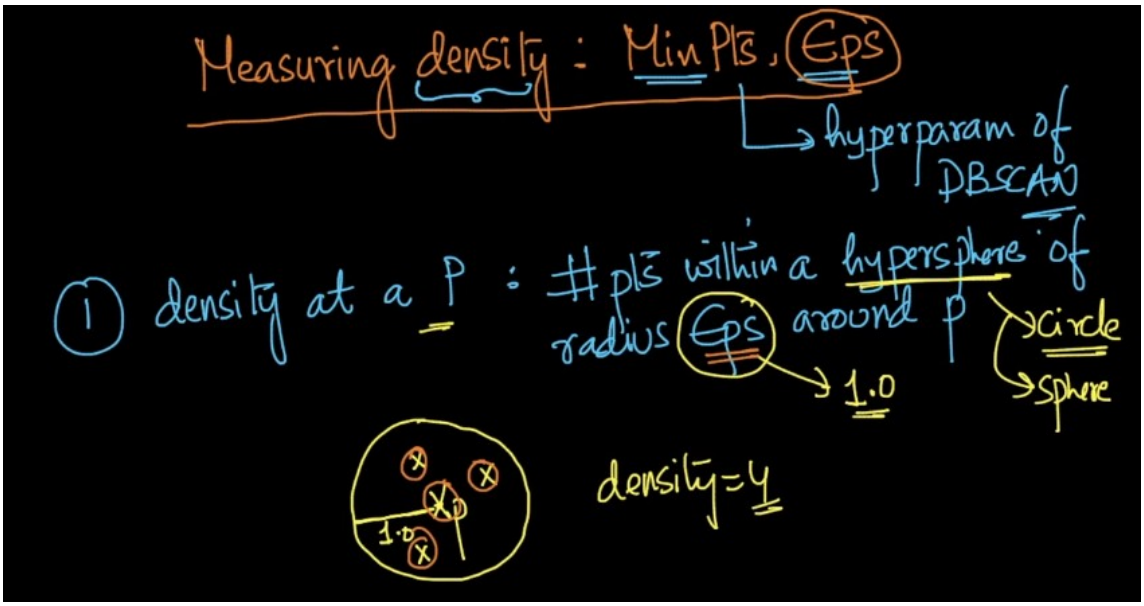
Density based Clustering
Centroid – based: K-Means
hierarchical – Agglomorative.
DBSCAN:



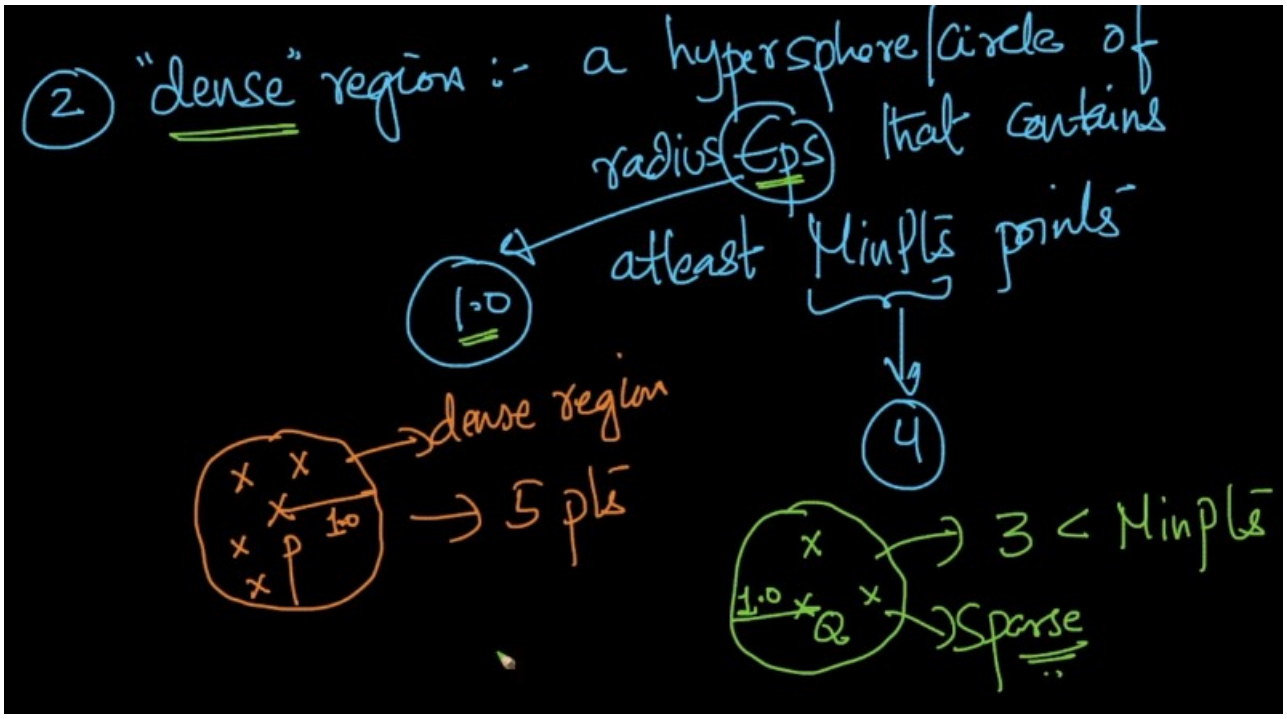
How to measure density?
Key – Idea: Min points, epsilon, core point, border point, noise point.

Min points and epsilon: Density
The Min pts and epsilon are the hyper parameters.
Density at a point P: # Pts within a hyper sphere of radius eps around p. The density at that point is 4.
Density:



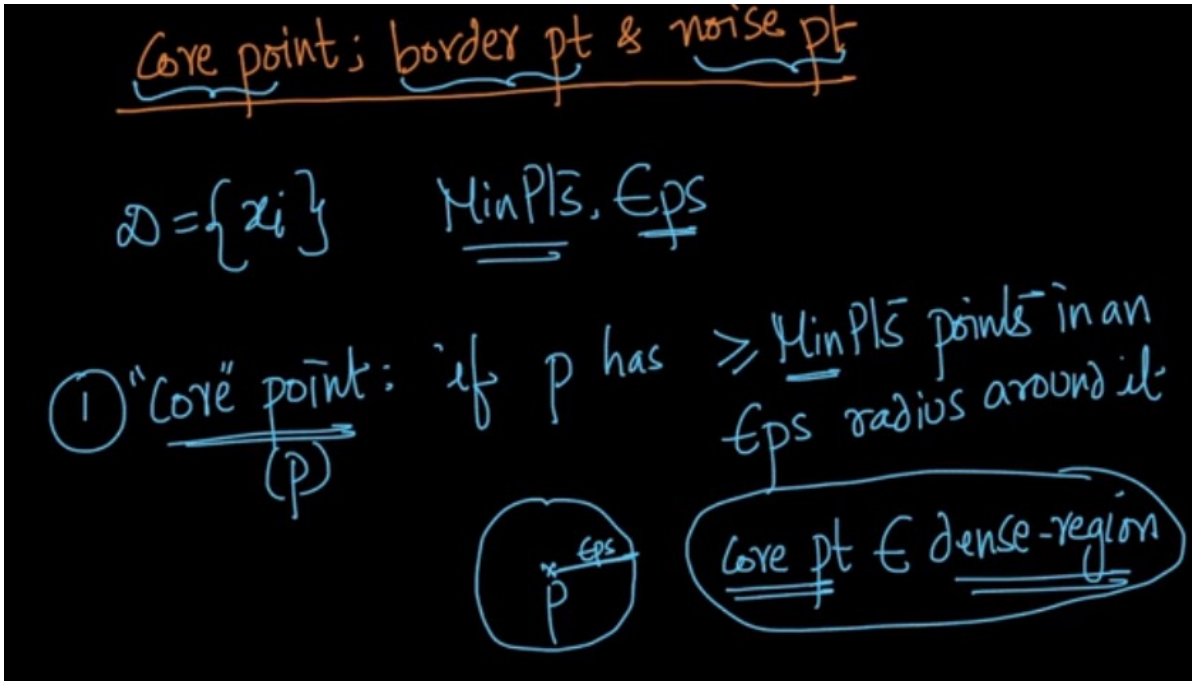
2. Dense region: A hypersphere/circle of radius eps that contains atleast min points.

Dense region is decided by the number of points in the dense circle. Hence this region is dense or sparse.



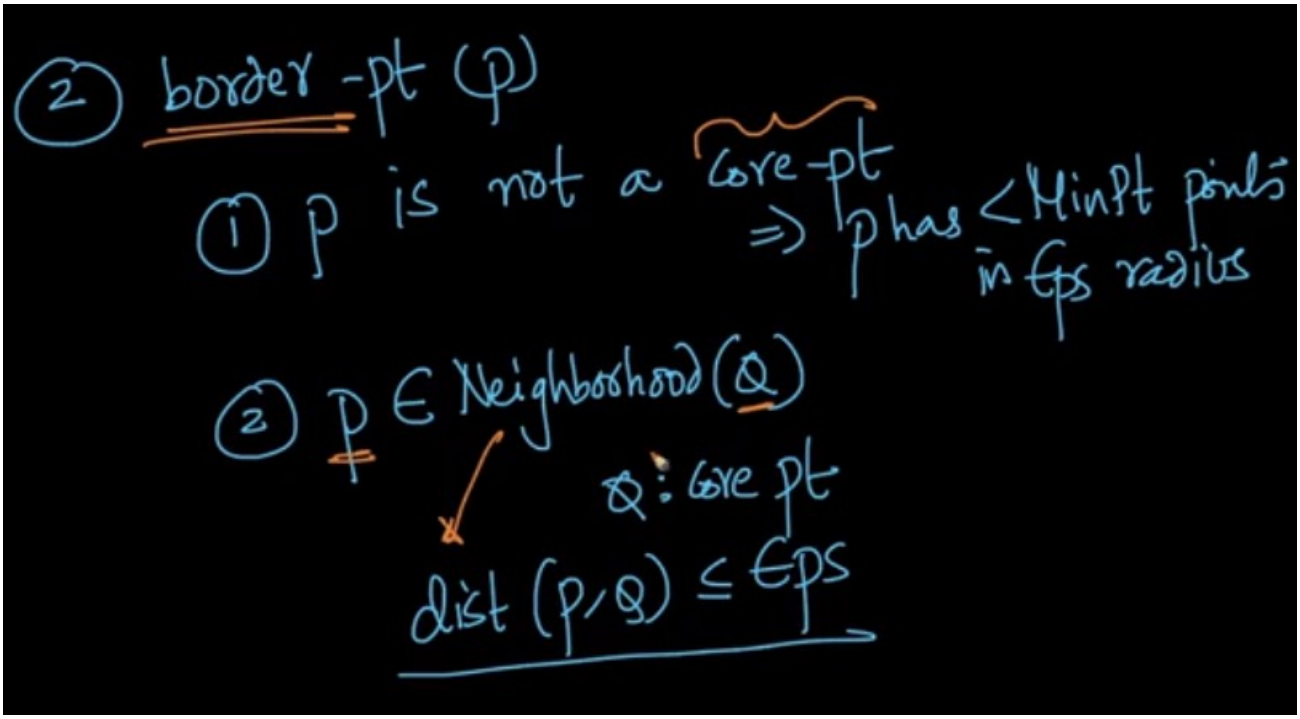
Core, border and noise points:

A core point always belongs to dense region.



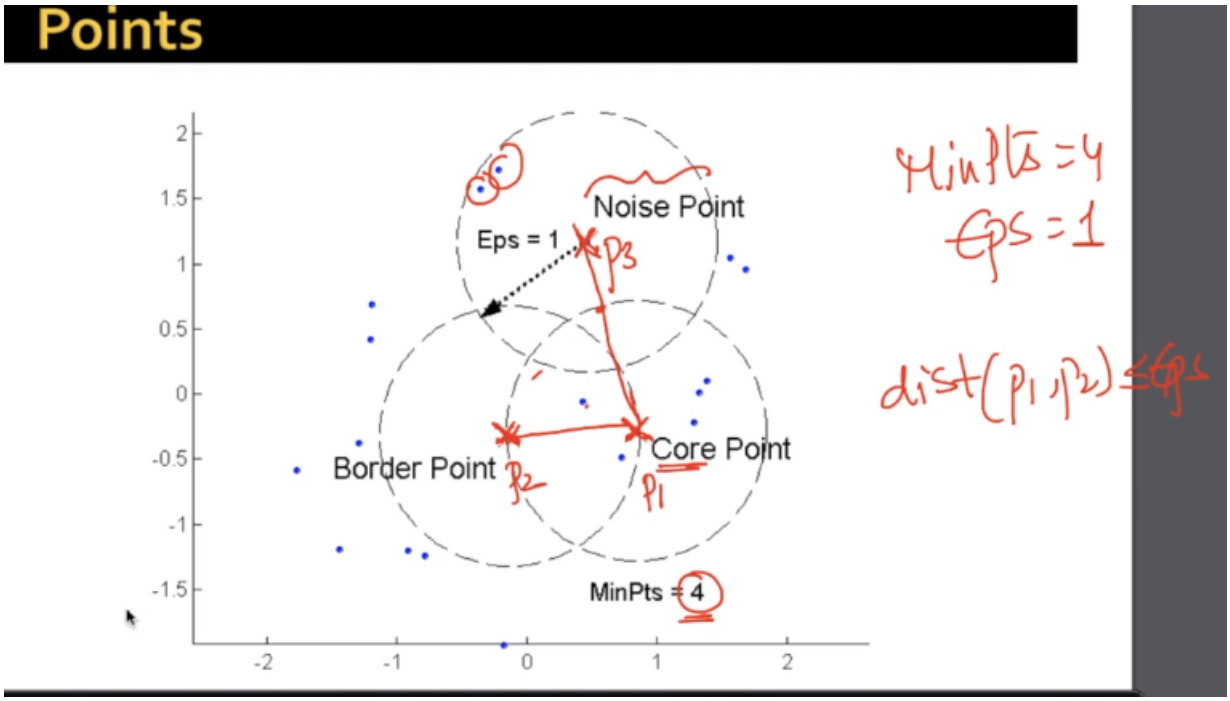
Border – point(p):

1. P is not a core – point => p has < Minpt points in eps radius.
2. p belongs neigh(Q).



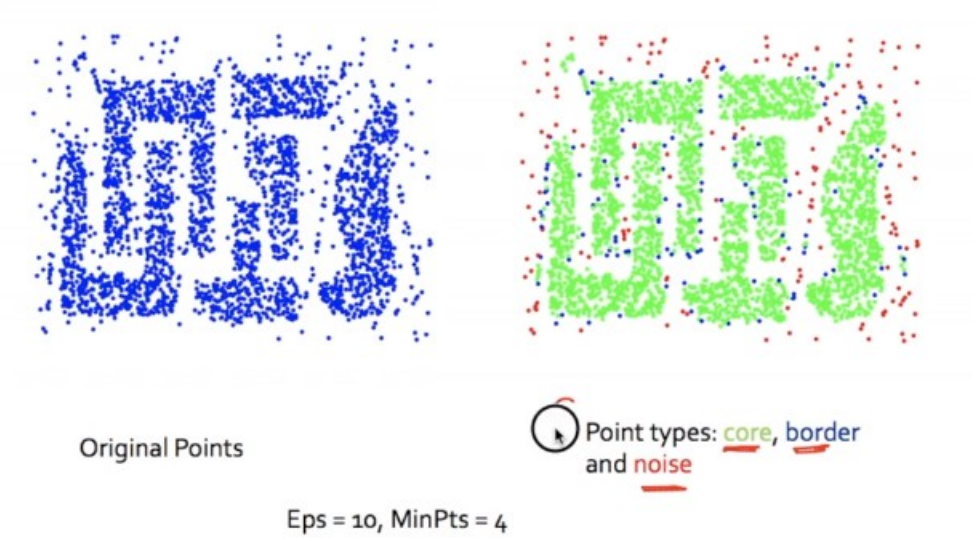
Noise point:

1. Neither core point nor a border are called noise points.

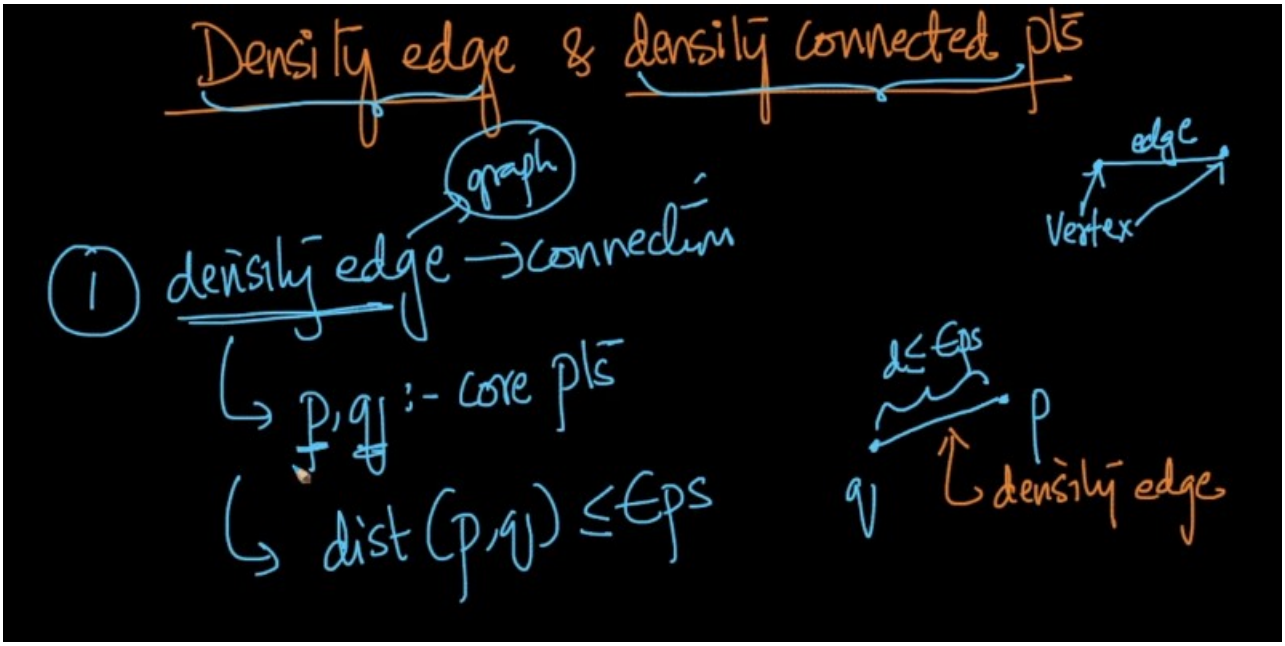


Example:

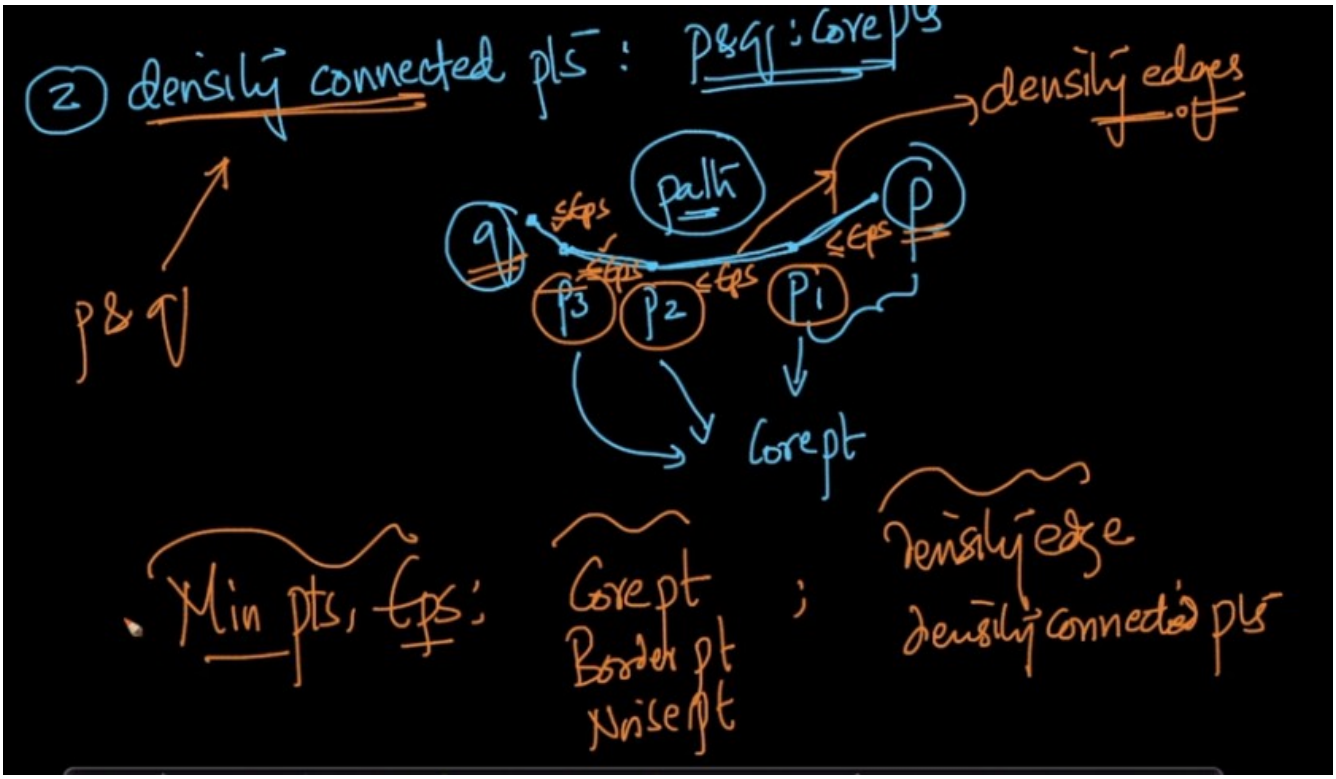
All the green points are the core points, blue points are the border points, red are the noise points.



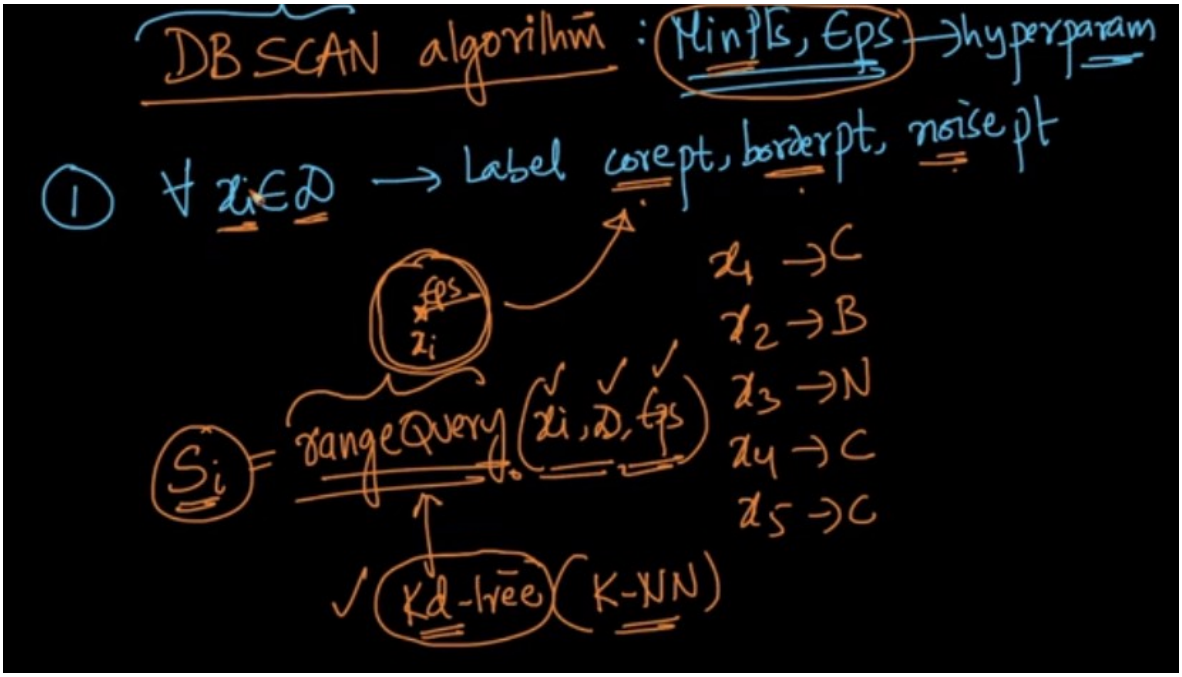
Density edge and Density connected points:
Density edge: When p and q are core points and distance(p, q) <= eps.



Density connected points:
There is a path connecting the density edges. This is called the density connected points.

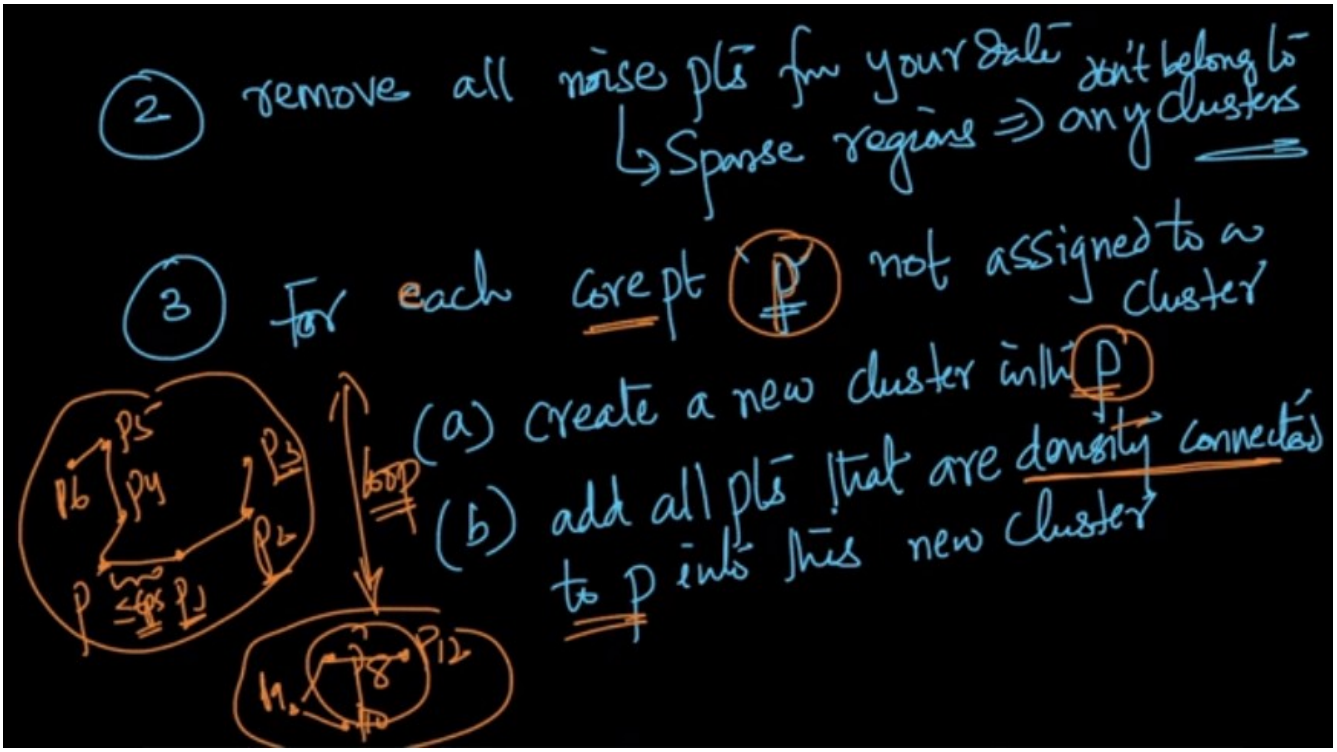


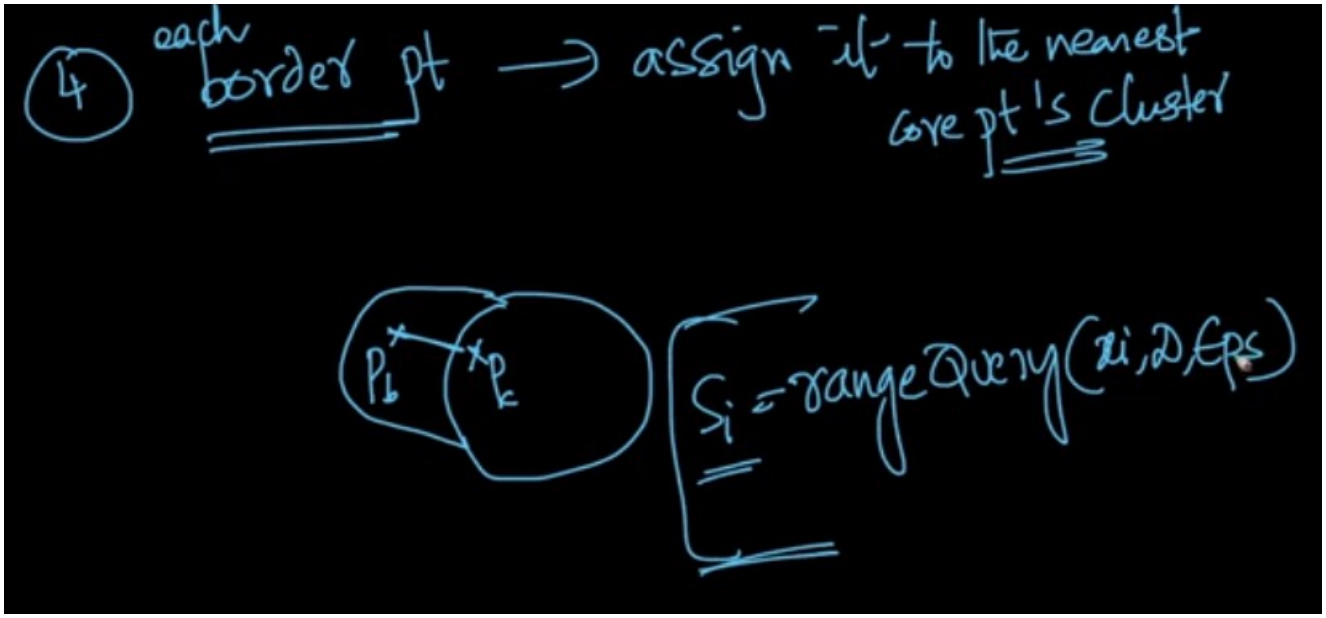
DBSCAN Algorithm:
The min pts, eps are the hyper parameters. We will label every point as core, border (or) neighbor point.
To implement this, we use the range query over the data points, min points and epsilon.



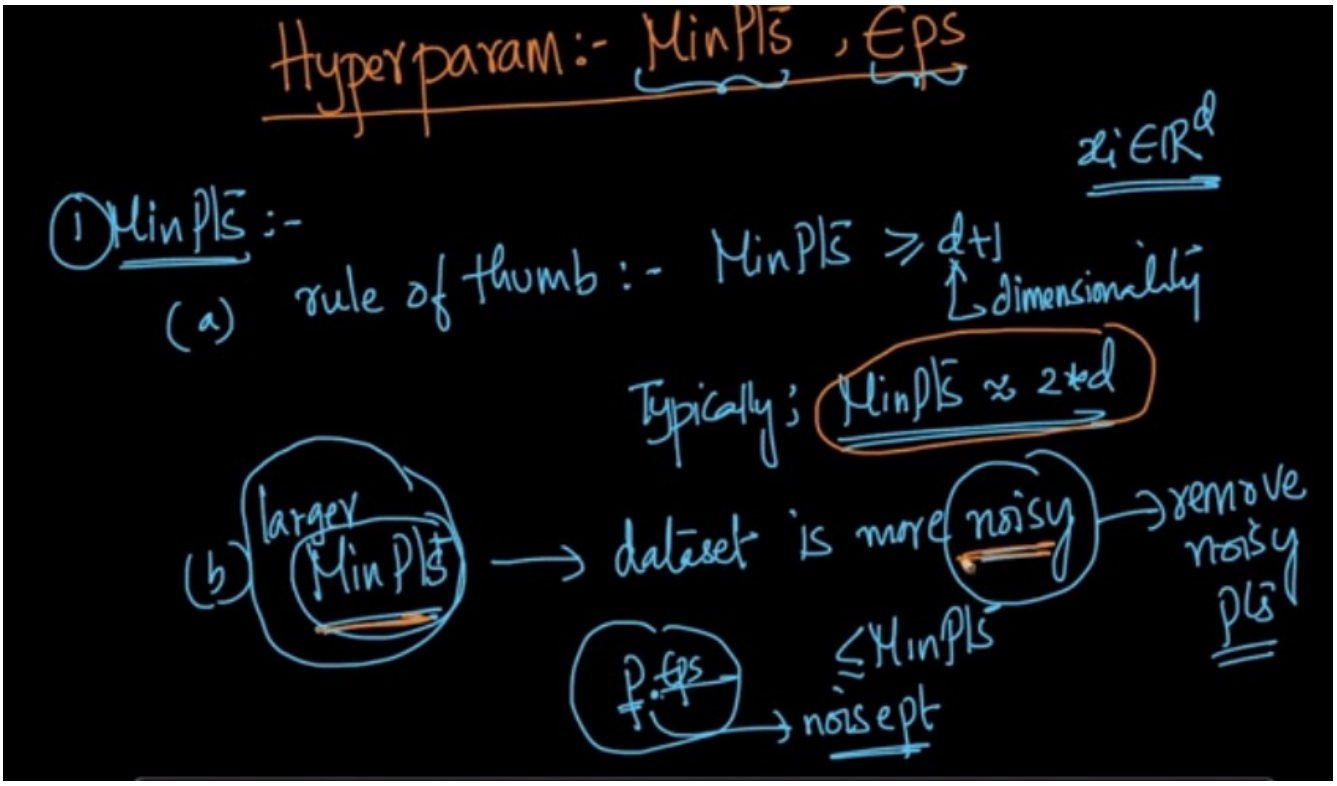
This is implemented by kd – tree.

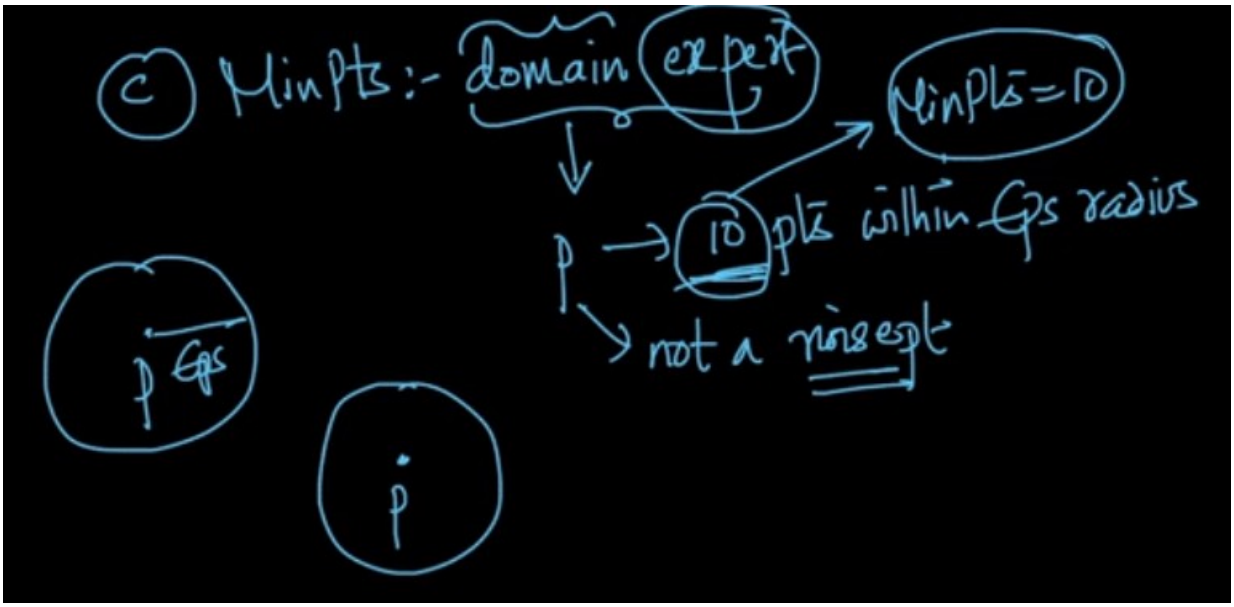
2. Remove all the noise points, that are obtained from the step – 1. They don't belong to any cluster.
   For each core point 'p' not assigned to a cluster.



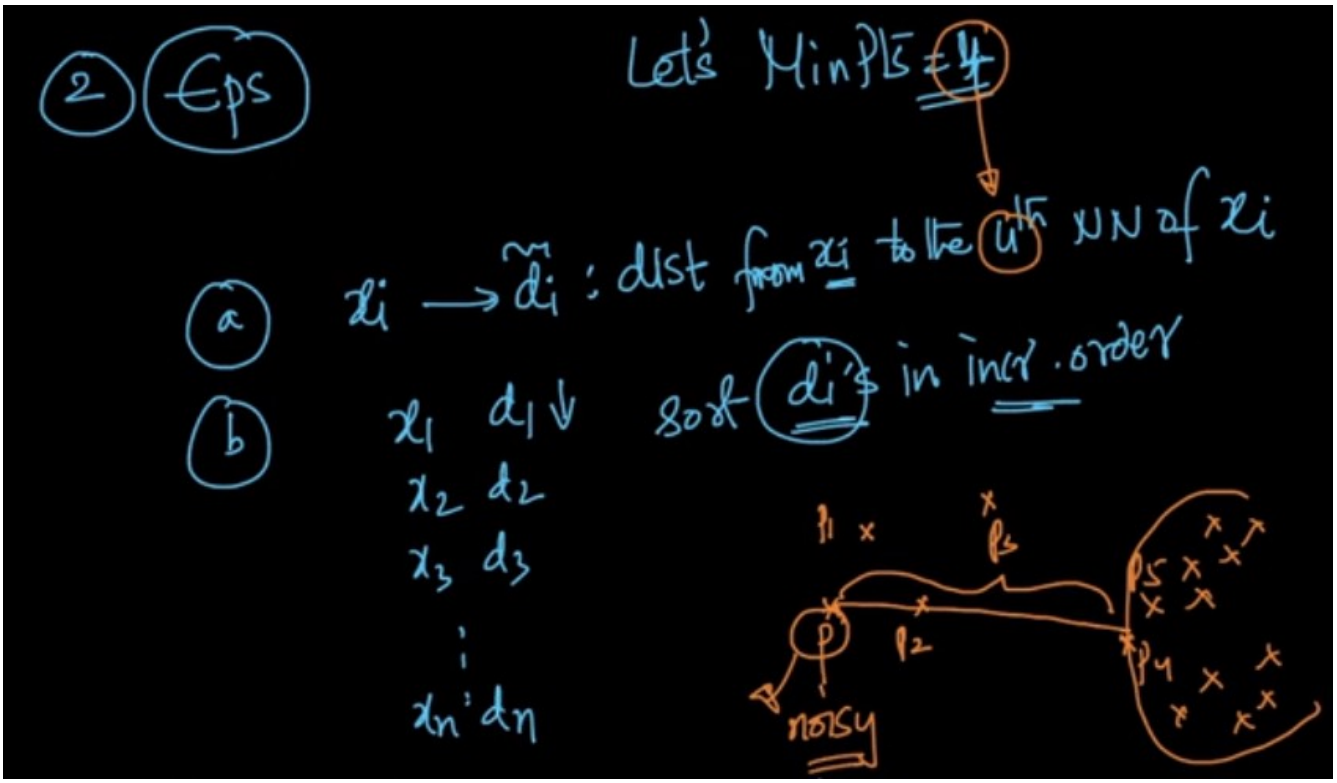


Hyper parameters: MinPts and Eps.

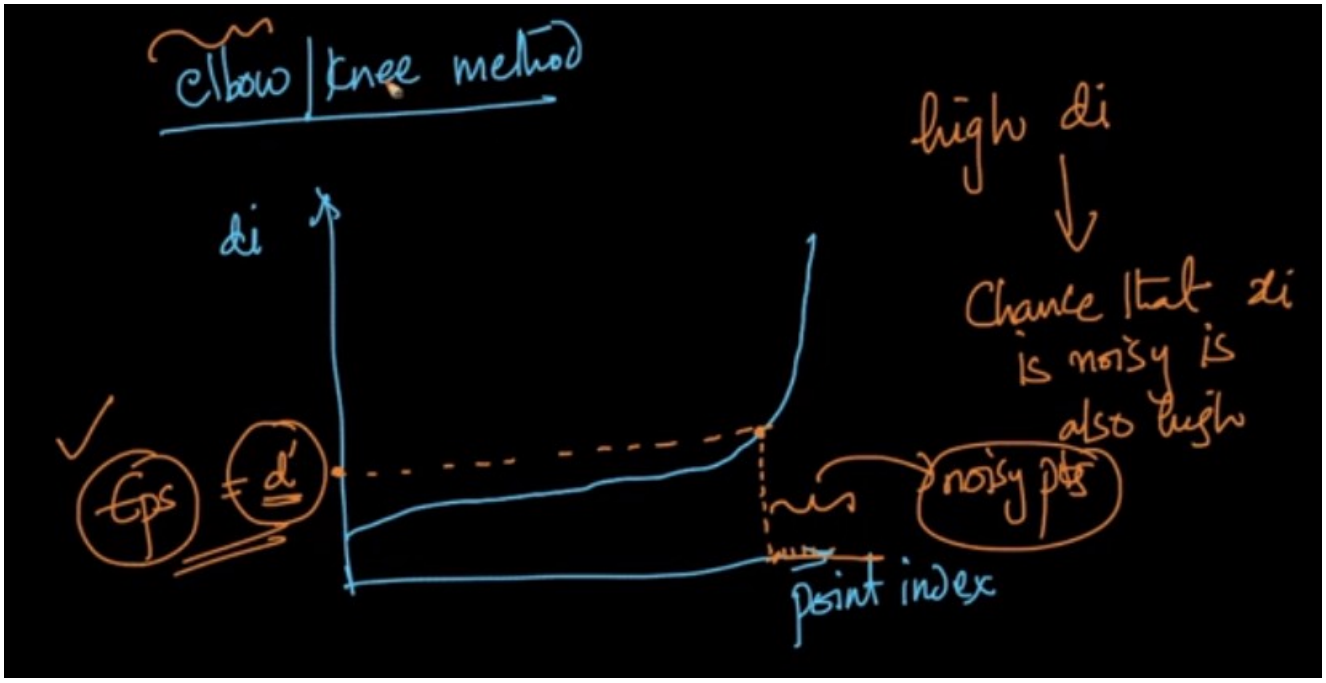




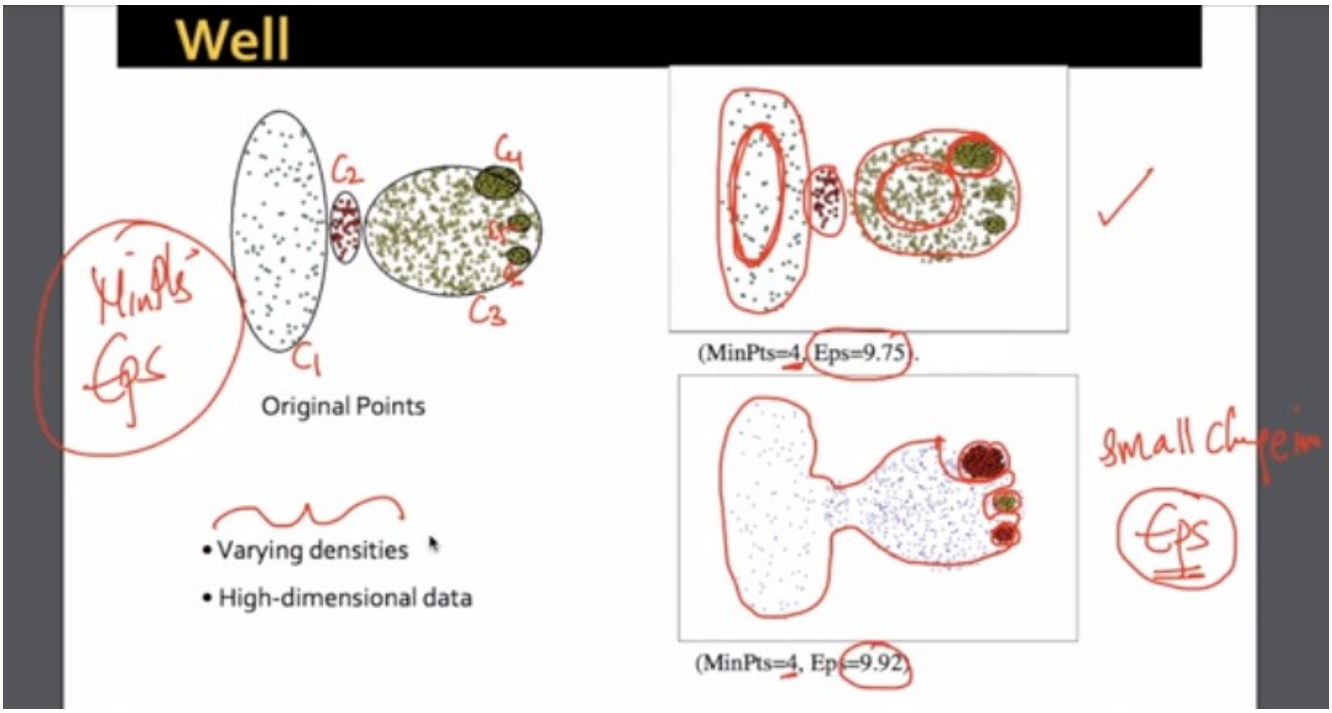
Epsilon value:



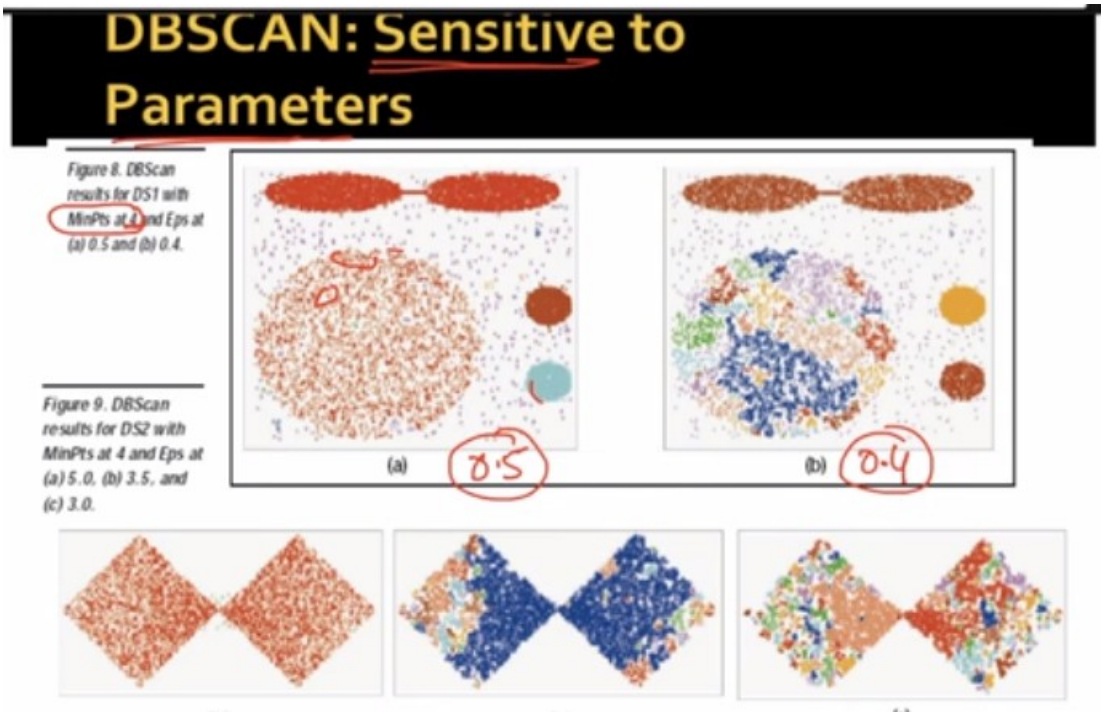
Elbow / Knee method:



Advantages and Limitations of DBSCAN:

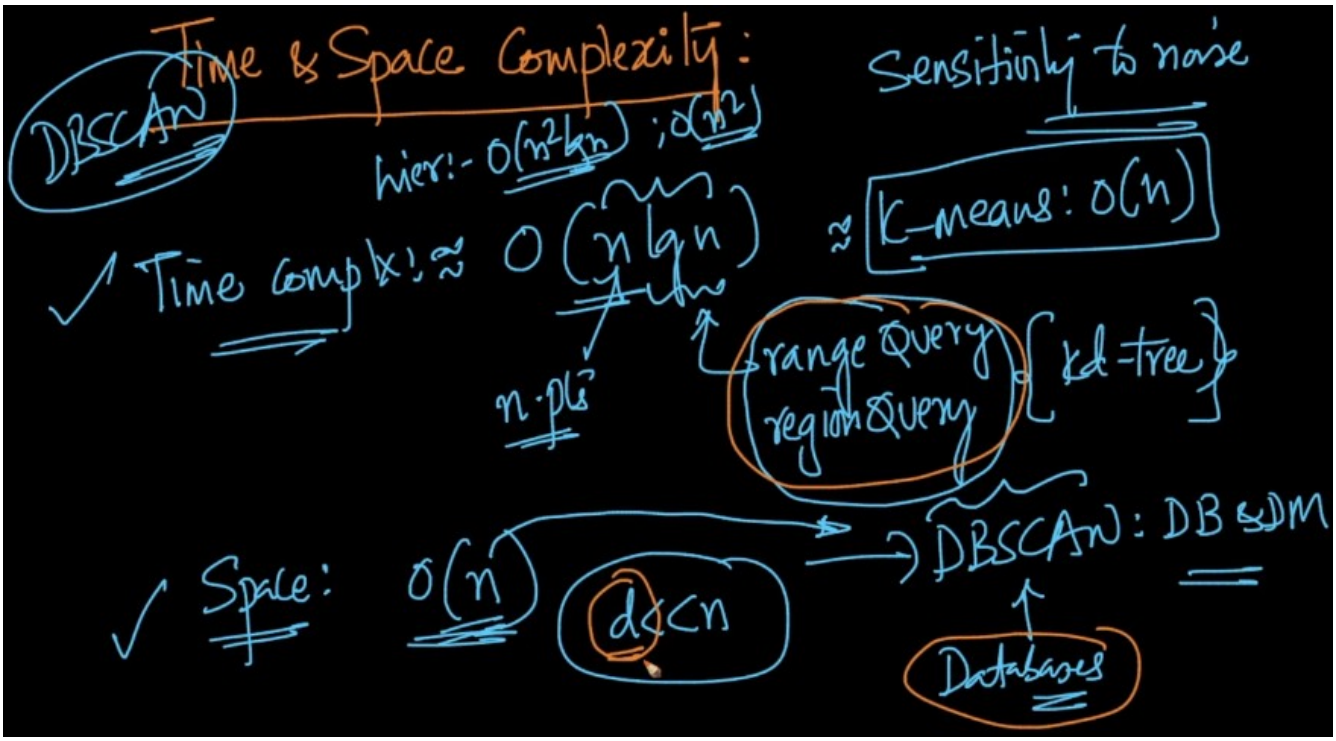


The DBS is extremely sensitive to EPS.



Time and Space Complexity:

The average time complexity for DBSCAN is O(n * log(n)) .



Code samples: