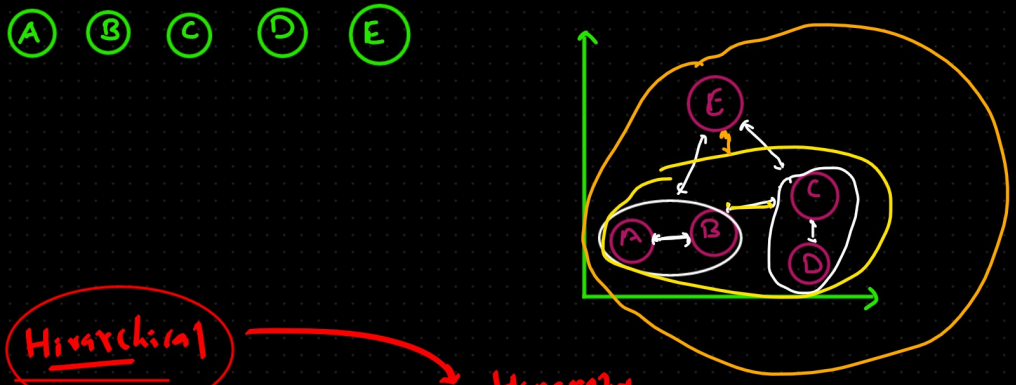
1. Hirarchical clustering
2. DBSCAN
3. Practical
4. Few remaining topic

Scikit (10)

Clustering => Group => group1, group2, group3

- K-Means & Kmeans++
- Hirarchical
- DBScan

1. Agglomeritive => Bottom to up
2. Divisive => top to Bottom



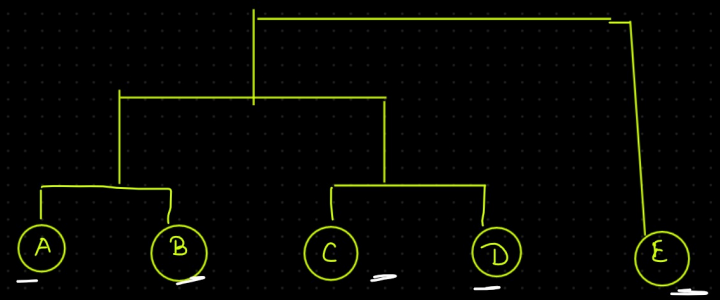
Hirarchical → Hirarchy

top to bottom

Divisive (Divide)

top => All the Data point is being treated as a single cluster

Bottom → all the Data is being treated as a separate or individual cluster

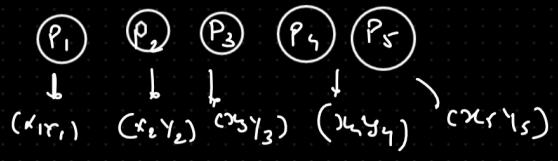


Agglomerative

Bottom to top

Agglomerative ⇒
1. each Data Point is a cluster
2. Bottom to top approach
3. Combine all the Data Point into single cluster

Point matrix ⇒ 5 Points
or
Distance matrix



calculating a Dist b/w each Point

Distance ⇒ B/w the Points

Ecuildian Dist = $\sqrt{(y_2-y_1)^2+(x_2-x_1)^2}$

Note Imp

i am going to fill this matrix with any value

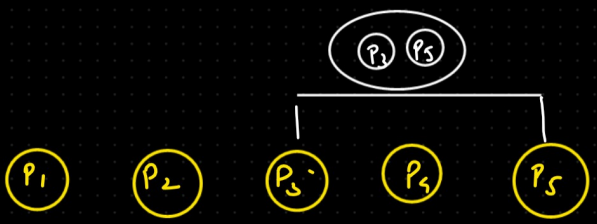
	P1	P2	P3	P4	P5
P1	0				
P2	9	0			
P3	3	7	0		
P4	6	5	9	0	
P5	11	10	2	8	0

Single linkage method ⇐ Minimum Dist B/w the Points

min → Dist (P5, P3)

⇓

Single cluster



min [(P3, P5) , P2]

min [(P3, P5) , P1]

min [D(P3, P2) D(P5, P2)]

min [D(P3, P1) D(P5, P1)]

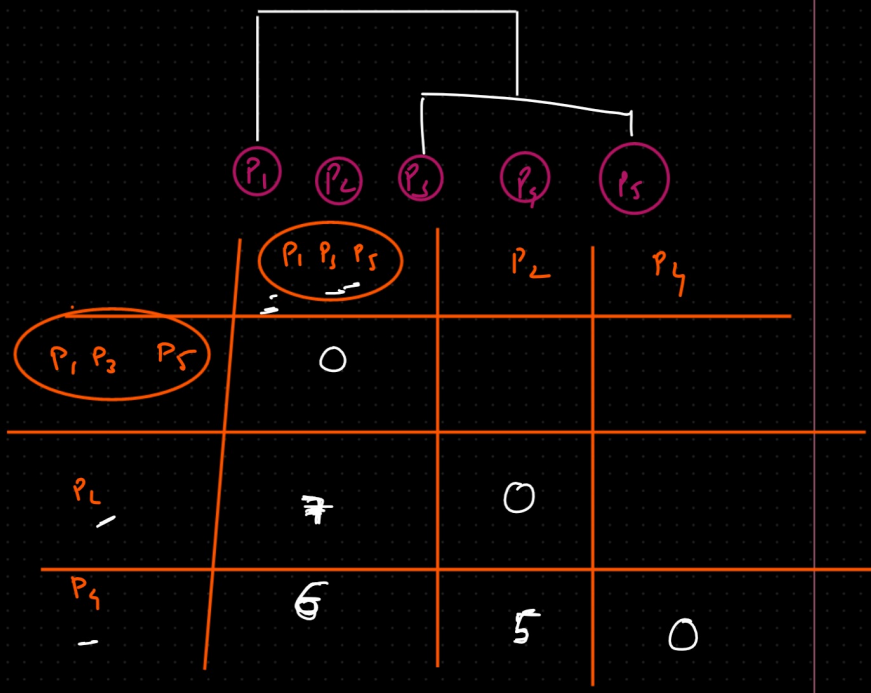
P1	0			
P2	9	0		
P3, P5	3	7	0	
P4	6	5		0

min [7 , 10]

min (3 , 11)

D (P3, P1)

7 (P3, P2) / D(P3, P2)



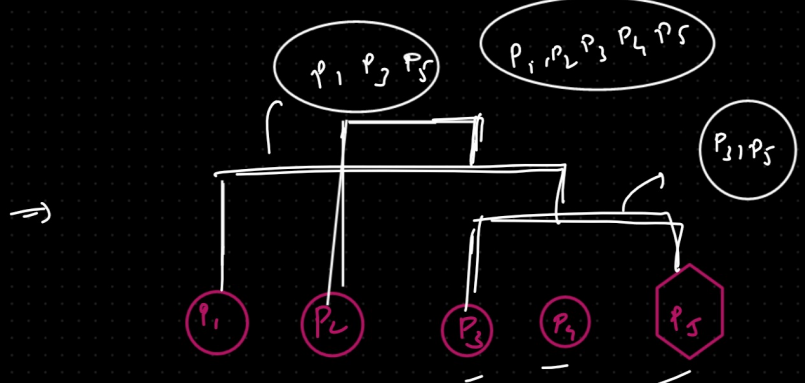
min

min [D ((P1, P3, P5) , P2)]

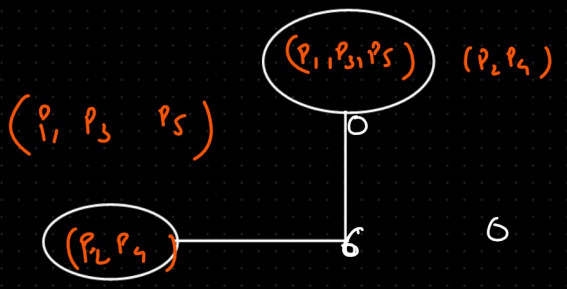
min [D ((P1, P3, P5) , P4)]

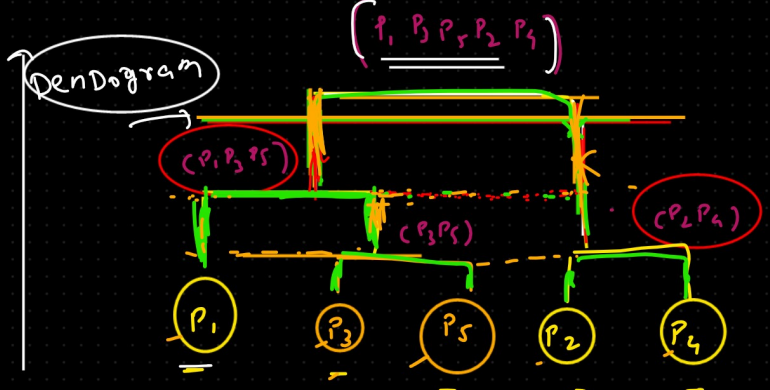
min [D(P1, P2) D(P3, P2) D(P5, P2)] = [9 , 7 , 10] = (7)

min [D(P1, P4) D(P3, P4) D(P5, P4)] = [6 , 9 , 8] = (6)



	P1	P2	P3	P4	P5
P1	0				
P2	9	0			
P3	3	7	0		
P4	6	5	9	0	
P5	11	10	2	8	0



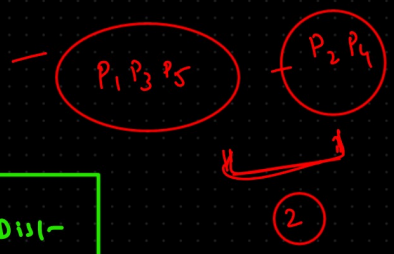


$$\min\left[(P_1 P_3 P_5)\ (P_2 P_4)\right]$$

$$\min\left[(P_1,P_2)\ (P_1,P_4]\ (P_3\ P_2)\ (P_3\ P_4)\ (P_5\ P_2)\ (P_5\ P_4)\right]$$

$$\min(9, 7, 10, 6, 5, 8)$$

(6)



Line ⇒ inter cluster Dist-

Density Based Method (DBSCAN)

Density Based Spatial Clustering with Application with Noise ⇒ Outlier

1. Eps Distance
2. Minimum Point
3. Core Point
4. Border Point
5. Noise Point

1. Density (cluster, group)
2. Noise (outlier)

EPS Distance ⇒ 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 ... ∞ ⇒ hyperparameter

{ Based on the data }

1. MinPoints ⇒ 1, 2, 3, 4, 5 ... ∞ { hyperparameter }
2. Core Point
3. Border Point
4. Noise Point

border point

Noise point

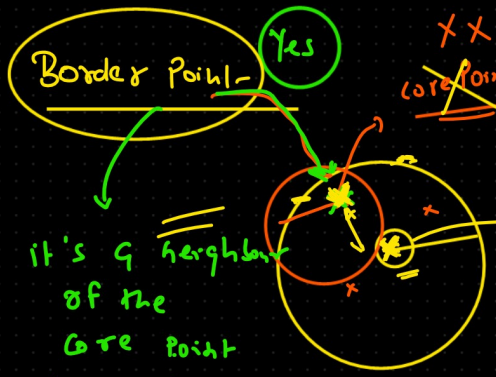
NO ⇒ why/not satisfying the Cond.

EPS = 5

minPoints = 3

Create a circle by using this EPS value and make sure that you have min Points (3) in that circle

⇒ than only the center will be called Core Point

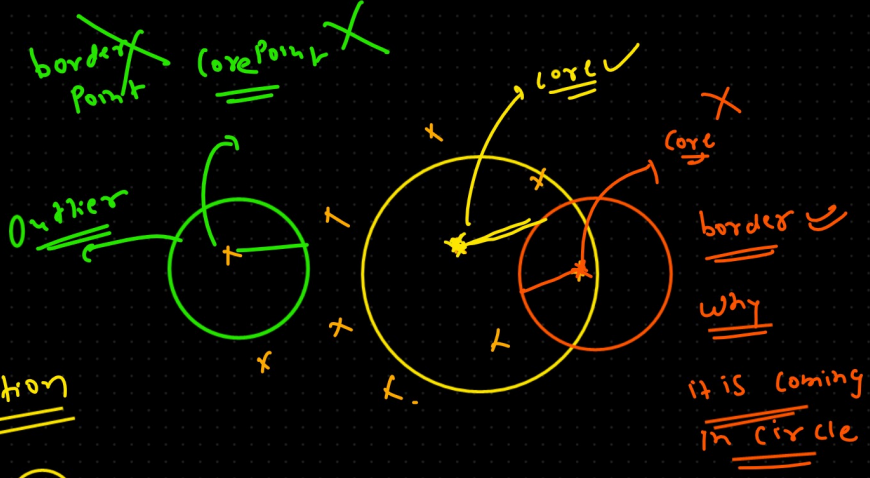


XX Why min Point condition not satisfying Core Point ?

Core Point

Noise Point

outlier



1. Core Point => Satisfying the min Point condition.

2. Border Point => Not satisfying the condition but Neighbour of the core Point

3. Noise Point => Neighter satisfying the cond. Nor a Neighbour of a core point

PCA

1 session

NCP with ML