05/03/21

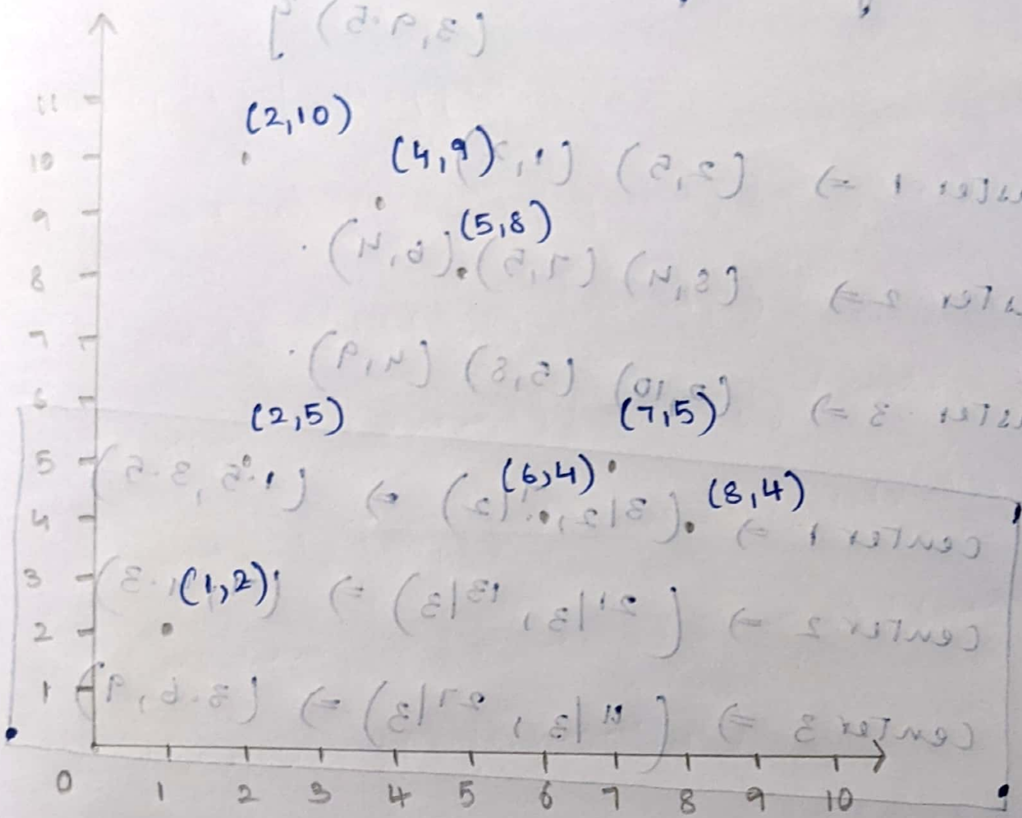
Homework-4

PART-A

1) K-Means Algorithm:

{ (2,10) (2,5) (8,4) (5,8) (7,5) (6,4) (1,2) (4,9) }

a)



b) 3 clusters

Cluster 1 ⇒ (1,2) (2,5)

Cluster 2 ⇒ (2,10) (4,9) (5,8)

Cluster 3 ⇒ (6,4) (7,5) (8,4)

c) initial centers {(2,5) (5,8) (4,9)}

Cluster 1 ⇒ (2,5) (1,2)

Cluster 2 ⇒ (5,8) (8,4) (7,5) (6,4)

Cluster 3 ⇒ (4,9) (2,10)

d) center 1 ⇒ $\left(\frac{2+1}{2}, \frac{5+2}{2}\right) = (1.5, 3.5)$

center 2 ⇒ $\left(\frac{5+8+7+6}{4}, \frac{8+4+5+4}{4}\right) = (6.5, 5.25)$

center 3 $\Rightarrow \left(\frac{4+2}{2}, \frac{9+10}{2}\right) = (3, 9.5)$.

e) Center's after second iteration:

clustering using {(1.5, 3.5) (6.5, 5.25) (3, 9.5)}

cluster 1 $\Rightarrow$ (2,5) (1,2)

cluster 2 $\Rightarrow$ (8,4) (7,5) (6,4).

cluster 3 $\Rightarrow$ (2,10) (5,8) (4,9).

$$\boxed{\begin{aligned} &\text{center 1} \Rightarrow (3/2, 7/2) \Rightarrow (1.5, 3.5) \\ &\text{center 2} \Rightarrow (21/3, 13/3) \Rightarrow (7, 4.3) \\ &\text{center 3} \Rightarrow (11/3, 27/3) \Rightarrow (3.6, 9). \end{aligned}}$$

f) centers after third iteration:

clustering using {(1.5, 3.5), (7, 4.3) (3.6, 9)}

cluster 1 $\Rightarrow$ (2,5) (1,2)

cluster 2 $\Rightarrow$ (8,4) (7,5) (6,4)

cluster 3 $\Rightarrow$ (2,10) (5,8) (4,9)

$$\boxed{\begin{aligned} &\text{center 1} \Rightarrow (1.5, 3.5). \\ &\text{center 2} \Rightarrow (7, 4.3). \\ &\text{center 3} \Rightarrow (3.6, 9). \end{aligned}}$$

g) cluster 1 ⇒ [(1,2)(2,5)] [(1,2)(2,5)]

cluster 2 ⇒ [(2,10)(4,9)(5,8)] [(2,10) (4,9)(5,8)]

cluster 3 ⇒ [(8,4)(7,5)(6,4) [(8,4)(7,5) (6,4)]

same.

h) No of iterations for the clusters to converge is 2. The clusters did not change after that.

i) Resulting centers : {(1.5,3.5) (7,4.3) (3.6,9)}

Resulting clusters :

cluster 1 ⇒ (2,5) (1,2).

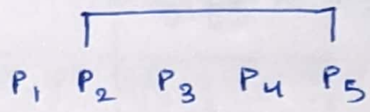
cluster 2 ⇒ (8,4) (7,5) (6,4).

cluster 3 ⇒ (2,10) (5,8) (4,9).

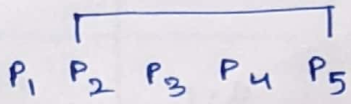
2) Hierarchial algorithm

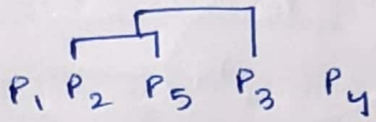
1) Single link:

$(P_2, P_5) - 0.98$

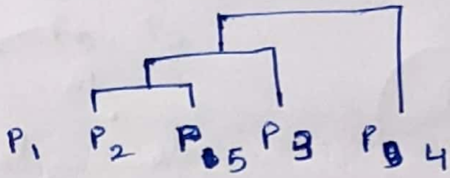


$(P_3, P_5) - 0.85$



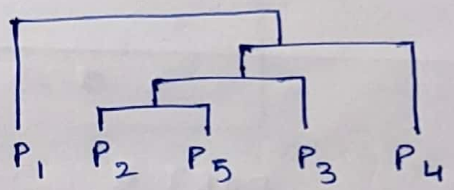


$(P_4, P_5) - 0.75$



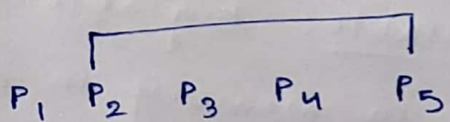
$(P_2, P_3) - 0.64$ (already exists)

$(P_1, P_4) - 0.55$

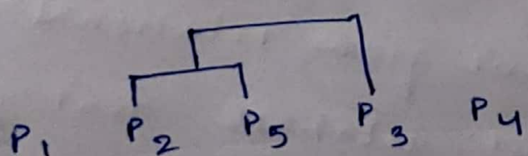


2) Complete Link:

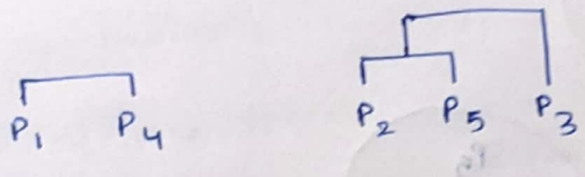
$(P_2, P_5) - 0.98$



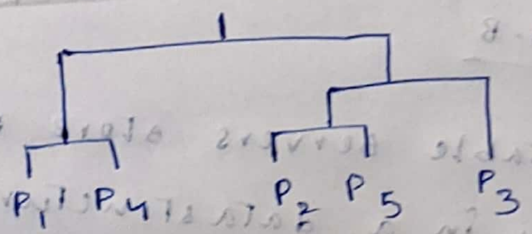
	P_1	$P_2 \cup P_5$	P_3	P_4
P_1	1.0	0.1	0.41	0.55
$P_2 \cup P_5$	0.10	1.0	0.64	0.47
P_3	0.41	0.64	1.0	0.44
P_4	0.55	0.47	0.44	1.0.



	P_1	$P_2 \cup P_3 \cup P_5$	P_4
P_1	1.0	0.1	0.55
$P_2 \cup P_3 \cup P_5$	0.1	1.0	0.44
P_4	0.55	0.44	1.0.



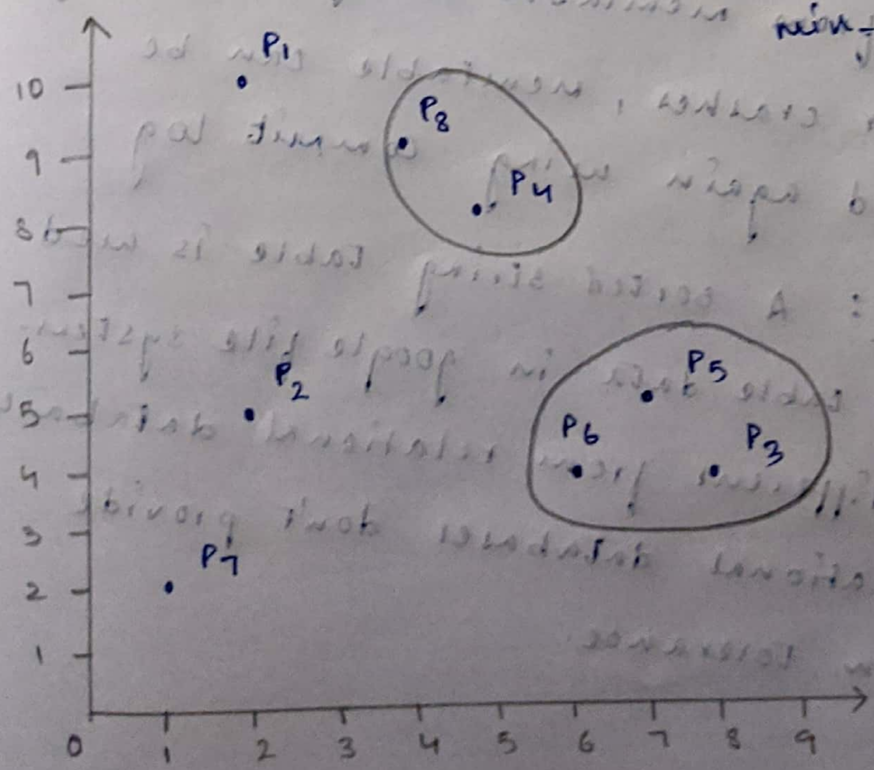
	$P_1 \cup P_4$	$P_2 \cup P_3 \cup P_5$
$P_1 \cup P_4$	1.0	0.1
$P_2 \cup P_3 \cup P_5$	0.1	1.0.



3) **DBSCAN Algorithm:**

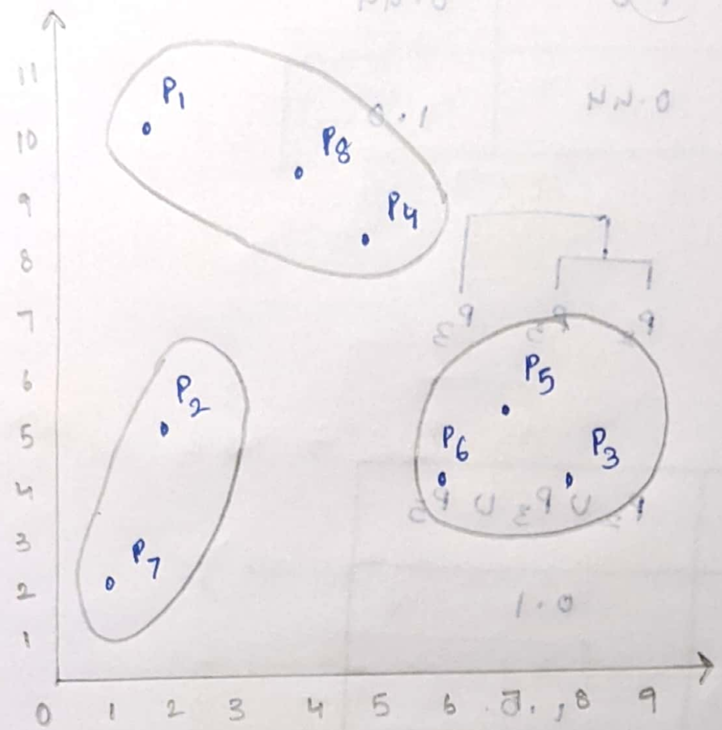
epsilon = 2

min_sample = 2



epsilon = $\sqrt{10}$.

min_sample = 2.



PART-B

4) Memtable: Bigtable servers store recent updates in memory in a datastructure known as memtable.

Reading from memtable is quick and if the server crashes, memtable can be constructed again using commit log.

5) SStable: A sorted string table is used to store table data in google file system. It is different from relational databases as relational databases don't provide partition tolerance.

6) CAP Theorems:

For any system that shares data it is impossible to guarentee simultaneously the following properties:

1) consistency.
2) Availability.
3) Partition-tolerance.

Very large scale system will partition at some point.

1) it is essential to decide between consistency and availability.
2) Traditional DBMS prefer consistency and partition tolerance over availability and partition tolerance.
over availability.
3) Most web applications choose availability.

7) A tablet is a set of consecutive rows of a table and is a unit of distribution and load balancing within BigTable.

A tablet server stores and serves tablets to clients. For a given tablet, a tablet server acts as a leader and other servers follow. replicas of the tablet.