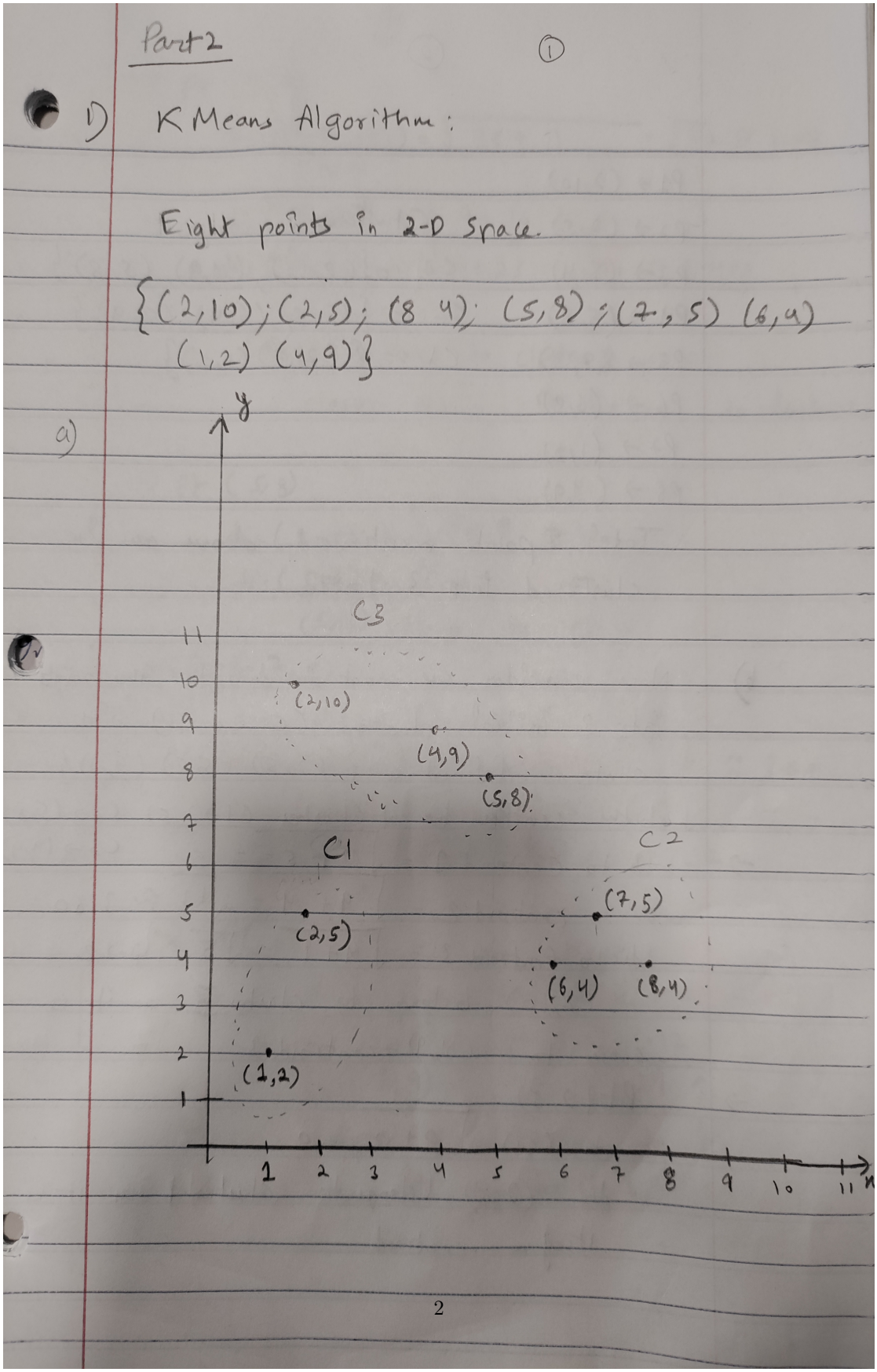
Part 2

Clustering and Recommendation Systems

Puru Jaiswal
PXJ200018

May 3, 2022

1 K-Means Algorithm



P1 → (2,10)
P2 → (2,5)
P3 → (8,4)
P4 → (5,8)
P5 → (7,5)
P6 → (6,4)
P7 → (1,2)
P8 → (4,9)

Clusters :
C3 := {(2,10), (4,9), (5,8)}
C2 = {(7,5), (6,4), (8,4)}
C1 := {(2,5) (1,2)}

Total 8 points mentioned above can be clustered into 3 clusters.

b) Now consider we want to divide the points into 3 initial clusters (C1, C2, C3) with centers defined as {(2,5) (5,8) (4,9)}

Let the centres be at Clusters C1 → (2,5) C2 → (5,8) C3 → (4,9)

→ dis to centroid 1 = $\sqrt{0+5^2} = 5$

dis to centroid 2 = $\sqrt{9+4} = \sqrt{13} \approx 3.60$

dis to centroid 3 = $\sqrt{4+1} = \sqrt{5} \approx 2.23$

(2,10) belongs to cluster 3 as it is closer to (4,9) → centroid 3.

→ Pt (2,5)

dis (c1) = $\sqrt{0+0} = 0$

(2,5) belongs to cluster 1 as it is itself a centroid

③

→ Pt (8,4)

dis (centroid c1) = $\sqrt{36+1} = \sqrt{37} \approx 6.08$
(2,5)

dis ((5,8)) = $\sqrt{9+16} = 5$

dis ((4,9)) = $\sqrt{16+25} = \sqrt{41} = 6.403$

(8,4) belongs to cluster 2 as it is closer c2 ∴ it belongs to cluster 2

→ Pt (5,8)

dis (2,5) = $\sqrt{9+9} \approx 4.24$

dis (5,8) = 0

∴ (5,8) belongs to c2

→ Pt (7,5)

dis (2,5) = $\sqrt{5^2} = 5$

dis (5,8) = $\sqrt{4+9} = \sqrt{13} \approx 3.60$

dis (4,9) = $\sqrt{9+16} = 5$

∴ 7,5 belongs to c2

→ Pt (6,4)

dis (2,5) = $\sqrt{16+1} = \sqrt{17} = 4.12$

dis (5,8) = $\sqrt{1+16} \approx 4.12$

dis (4,9) = $\sqrt{4+25} = \sqrt{29} \approx 5....$

Here (6,4) can belong to either c1 or c2, let's say ~~it~~ it belongs to c2

→ Pt (1,2)

dis (2,5) = $\sqrt{1+9}$ = $\sqrt{10}$ = 3.16

dis (5,8) = $\sqrt{16+36}$ = $\sqrt{52}$ ≈ 7.2

dis (4,9) = $\sqrt{9+49}$ = $\sqrt{58}$

(1,2) to belongs to C1

→ Pt (4,9)

dis (4,9) = 0

4,9 belongs to C3

C1 → {(1,2), (2,5)}

C2 → {(8,4), (5,8), (7,5) (6,4)}

C3 → {(2,10), (4,9)}

c) After 1 Iteration

Centre for C1

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

centre for C2

$\left(\frac{8+5+7+6}{4}, \frac{4+8+5+4}{4}\right)$

= (6.5, 5.25)

Centre for C3

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

New centres C1 → (1.5, 3.5) C2 → (6.5, 5.25)
C3 → (3, 9.5)

⑤

→ Pt (2, 10)

dis (1.5, 3.5) = $\sqrt{(0.5)^2+(6.5)^2} \approx 6.519$

dis (6.5, 5.25) = $\sqrt{(4.5)^2+(4.75)^2}$

≈ 6.543

dis (3, 9.5) = $\sqrt{1+(0.5)^2} \approx 1.118$

∴ (2, 10) belongs to C2

→ Pt (2, 5)

dis (1.5, 3.5) = $\sqrt{(0.5)^2+(1.5)^2} \approx 1.5$

dis (6.5, 5.25) = $\sqrt{(4.5)^2+(0.25)^2} \approx 4.507$

dis (3, 9.5) = $\sqrt{1+(4.5)^2} \approx 4.60$

∴ (2, 5) is close to C1

→ Pt (8, 4)

dis (1.5, 3.5) = $\sqrt{(6.5)^2+(0.5)^2} = 6.519$

dis (6.5, 5.25) = $\sqrt{(1.5)^2+(1.25)^2} = 1.952$

dis (3, 9.5) = $\sqrt{25+(5.5)^2} = 7.433$

(8, 4) belongs to C2

→ Pt (5, 8)

dis (1.5, 3.5) = $\sqrt{(3.5)^2+(4.5)^2} = 5.70$

dis (6.5, 5.25) = $\sqrt{(1.5)^2+(2.75)^2} = 3.132$

dis (3, 9.5) = $\sqrt{4+(1.5)^2} = 2.5$

∴ (5, 8) belongs to C3

→ Pt (7,5) (6)

dis (1.5, 3.5) = $\sqrt{(5.5)^2+(1.5)^2} \approx 5.7$

dis (6.5, 5.25) = $\sqrt{(0.5)^2+(0.25)^2} \approx 0.56$

dis (3, 9.5) = $\sqrt{16+(4.5)^2} \approx 6.02$

(7,5) is more closer to C2.

→ Pt (6,4)

dis (1.5, 3.5) = $\sqrt{(4.5)^2+(0.5)^2} \approx 4.527$

dis (6.5, 5.25) = $\sqrt{(0.5)^2+(1.25)^2} \approx 1.346$

dis (3, 9.5) = $\sqrt{9+(5.5)^2} \approx 6.265$

∴ (6,4) belongs to C2.

→ Pt (1,2)

dis (1.5, 3.5) = $\sqrt{(0.5)^2+(1.5)^2} \approx 1.581$

dis (6.5, 5.25) = $\sqrt{(5.5)^2+(3.25)^2} \approx 6.388$

dis (3, 9.5) = $\sqrt{4+(7.5)^2} \approx 7.762$

(1,2) belongs to C1

→ Pt (4,9) =

dis (1.5, 3.5) = $\sqrt{(2.5)^2+(5.5)^2} \approx 6.041$

dis (6.5, 5.25) = $\sqrt{(2.5)^2+(3.75)^2} \approx 4.507$

dis (3, 9.5) = $\sqrt{1+(0.5)^2} \approx 1.118$

(4,9) belongs to C3

C1 → { (2,5) (1,2) }

C2 → { (8,4), (7,5), (6,4) }

C3 → { (5,8), (4,9) (2,10) }

d) What's the center of cluster after 2nd iteration?

(7)

Center for c1

$\left(\frac{3}{2}, \frac{7}{2}\right) \rightarrow (1.5, 3.5)$

center for c2: $\left(\frac{8+7+6}{3}, \frac{4+5+4}{3}\right)$

$= (7, 4.3)$

center for c3: $\left(\frac{5+4+2}{3}, \frac{8+9+10}{3}\right)$

$(3.6, 9)$

→ (2,10) $dis(1.5, 3.5) = \sqrt{(0.5)^2 + (6.5)^2} \approx 6.519$

$dis(7, 4.3) = \sqrt{25 + (5.7)^2} \approx 7.582$

$dis(3.6, 9) = \sqrt{(1.6)^2 + 1} \approx 1.882$

(2,10) belongs to c3

→ (2, 5)

$dis(1.5, 3.5) = \sqrt{(0.5)^2 + (1.5)^2} \approx 1.58$

$dis(7, 4.3) = \sqrt{25 + (0.7)^2} \approx 5.05$

$dis(3.6, 9) = \sqrt{(1.6)^2 + 16} \approx 4.31$

(2, 5) belongs to c1

→ (8, 4) $dis(1.5, 3.5) = \sqrt{(6.5)^2 + (0.5)^2} \approx 16.519$

$dis(7, 4.3) = \sqrt{1 + (0.3)^2} \approx 1.044$

$dis(3.6, 9) = \sqrt{(4.4)^2 + 5^2} \approx 6.66$

(8, 4) belongs to c2

→ (5, 8) (8)

dis (1.5, 3.5) = $\sqrt{(3.5)^2 + (4.5)^2} \approx 5.70$

dis (7, 4.3) = $\sqrt{2^2 + (3.7)^2} \approx 4.206$

dis (3.6, 9) = $\sqrt{(1.4)^2 + 1} \approx 1.72$

(5, 8) belongs to C3

→ (7, 5)

dis (1.5, 3.5) = $\sqrt{(5.5)^2 + (1.5)^2} \approx 5.7$

dis (7, 4.3) = $\sqrt{0 + (0.7)^2} \approx 0.7$

dis (3.6, 9) = $\sqrt{(3.4)^2 + 16} \approx 5.25$

(7, 5) belongs to C2

→ (6, 4)

dis (1.5, 3.5) = $\sqrt{(4.5)^2 + (0.5)^2} \approx 4.527$

dis (7, 4.3) = $\sqrt{1 + (0.3)^2} \approx 1.044$

dis (3.6, 9) = $\sqrt{(2.4)^2 + 25} \approx 5.548$

(6, 4) belongs C2

→ (1, 2)

dis (1.5, 3.5) = $\sqrt{(0.5)^2 + (1.5)^2} \approx 1.581$

dis (7, 4.3) = $\sqrt{6^2 + (2.3)^2} \approx 6.426$

dis (3.6, 9) = $\sqrt{(2.6)^2 + 49} \approx 7.467$

(1, 2) belongs to C1

→ (4, 9)

dis (1.5, 3.5) = $\sqrt{(2.5)^2 + (5.5)^2} \approx 6.041$

dis (7, 4.3) = $\sqrt{(9) + (4.7)^2} \approx 5.576$

dis (3.6, 9) = $\sqrt{(0.4)^2 + 0} \approx 0.4$

(4, 9) belongs to C3

(9)

$C_1 \rightarrow \{(2,5), (1,2)\}$

$C_2 \rightarrow \{(8,4), (7,5), (6,4)\}$

$C_3 \rightarrow \{(2,10), (5,8), (4,9)\}$

e) What is the center of clusters after on 3rd iteration?

Centroid 1 $\rightarrow \left(\frac{2+1}{2}, \frac{5+2}{2}\right) \rightarrow (1.5, 3.5)$

Centroid 2 $\rightarrow (7, 4.3)$

Centroid 3 $\rightarrow (3.6, 9)$

f) Iteration 1

$C_1 \rightarrow \{(1,2), (2,5)\}$

$C_2 \rightarrow \{(8,4), (5,8), (7,5), (6,4)\}$

$C_3 \rightarrow \{(2,10), (4,9)\}$

Part (a)

$C_1 \rightarrow (2,5)(1,2)$

$C_2 \rightarrow (7,5)(6,4)(8,4)$

$C_3 \rightarrow (2,10)(4,9)(5,8)$

Iteration 2:

$C_1 \rightarrow \{(2,5)(1,2)\}$

$C_2 \rightarrow \{(8,4)(7,5), (6,4)\}$

$C_3 \rightarrow \{(5,8), (4,9), (2,10)\}$

Iteration 3:

$C_1 \rightarrow \{(2,5)(1,2)\}$

$C_2 \rightarrow \{(8,4)(7,5), (6,4)\}$

$C_3 \rightarrow \{(5,8), (4,9), (2,10)\}$

(10) If we compare iteration 1 with part A we see that in iteration 1 cluster 2 consists of 4 points, after iteration 2 it has same points as part a, and in iteration 3 the centroid becomes constant, therefore the algorithm stops as there is no reassignment happening.

g) How many iterations are required for the clusters to converge?

3 clusters are required.

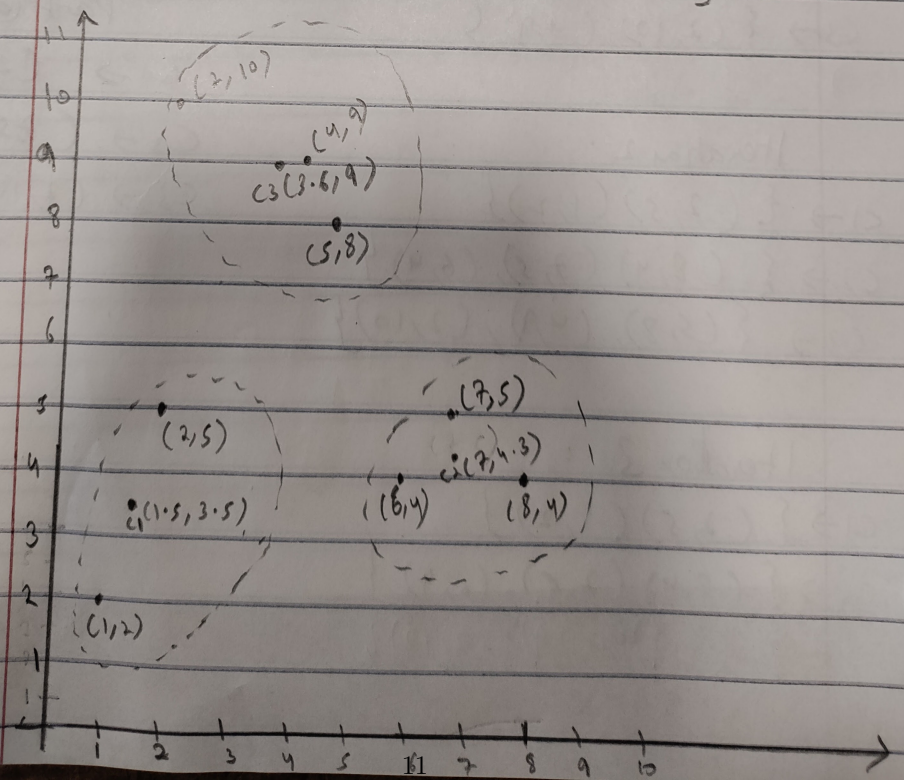
h) Resulting centers $\{(1.5, 3.5), (7, 4.3), (3.6, 9)\}$ (C1, C2, C3)

Resulting clusters:

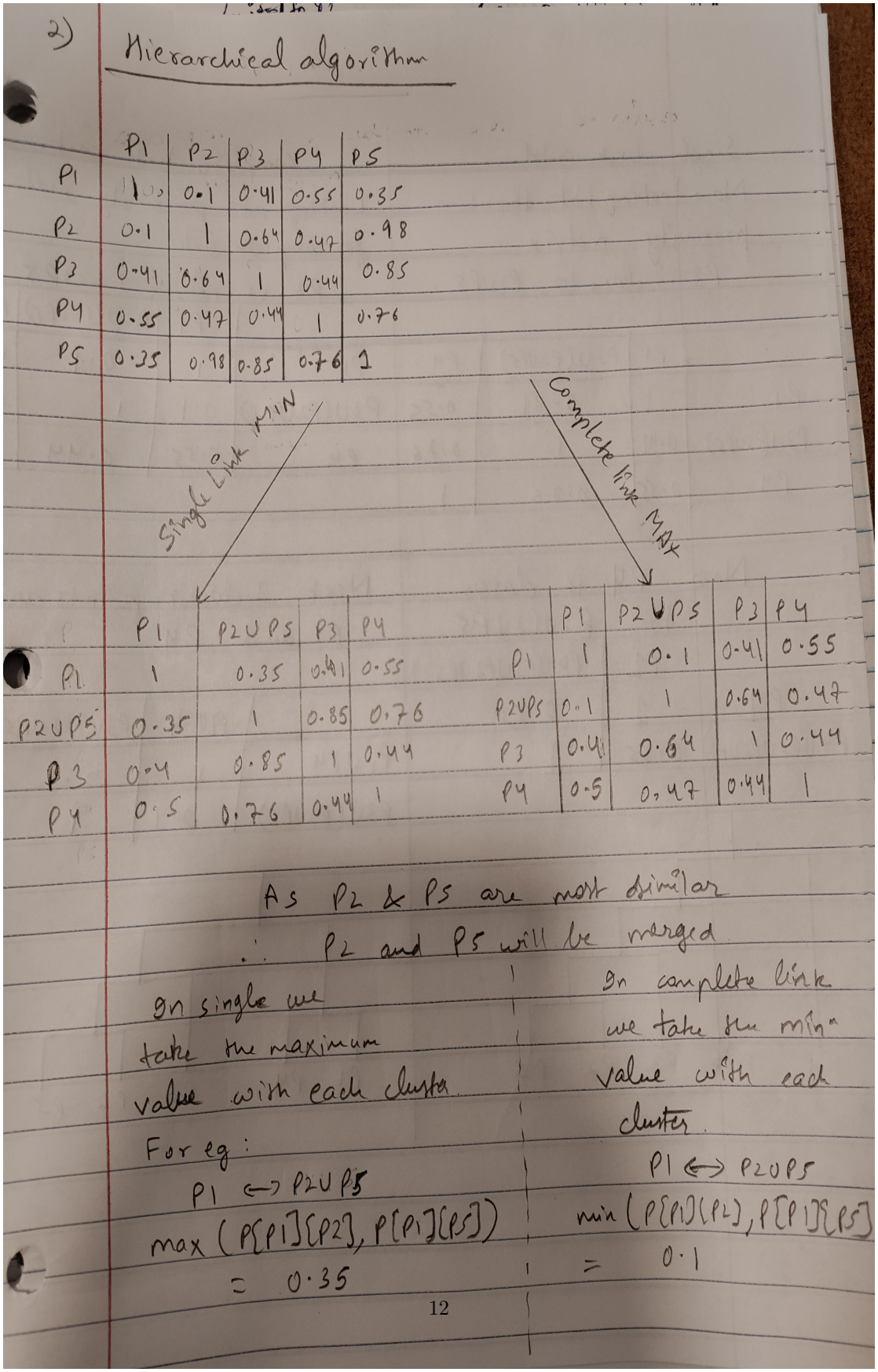
C1: {(2,5), (1,2)}

C2: {(8,4), (7,5), (6,4)}

C3: {(5,8), (4,9), (2,10)}



2 Hierarchical Algorithm



Single Link contd...

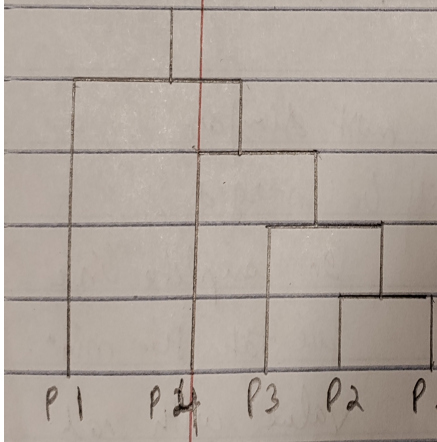
Now looking at the proximity matrix

P3 is close to P2UP5

	P1	P3U(P2UP5)	P4
P1	1	0.41	0.55
P3U(P2UP5)	0.41	1	0.76
P4	0.55	0.76	1

Now P4 is closer to P3UP2UP5

	P1	P4U(P3UP2UP5))
P1	1	0.55
P4U(P3UP2UP5)	0.55	1



complete link cont...

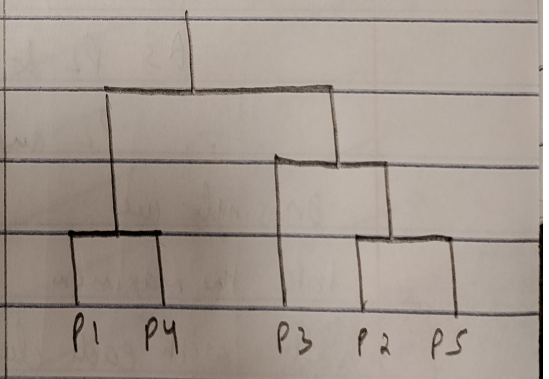
Looking at proximity matrix for complete link

P3 is closer P2UP5

	P1	P3U(P2UP5)	P4
P1	1	0.1	0.55
P3U(P2UP5)	0.1	1	0.44
P4	0.55	0.44	1

Next 2 closest points are P1 and P4

	P1UP4	P3U(P2UP5)
P1UP4	1	0.1
P3U(P2UP5)	0.1	1



3 DBSCAN Algorithm

3) DBSCAN Algorithm.

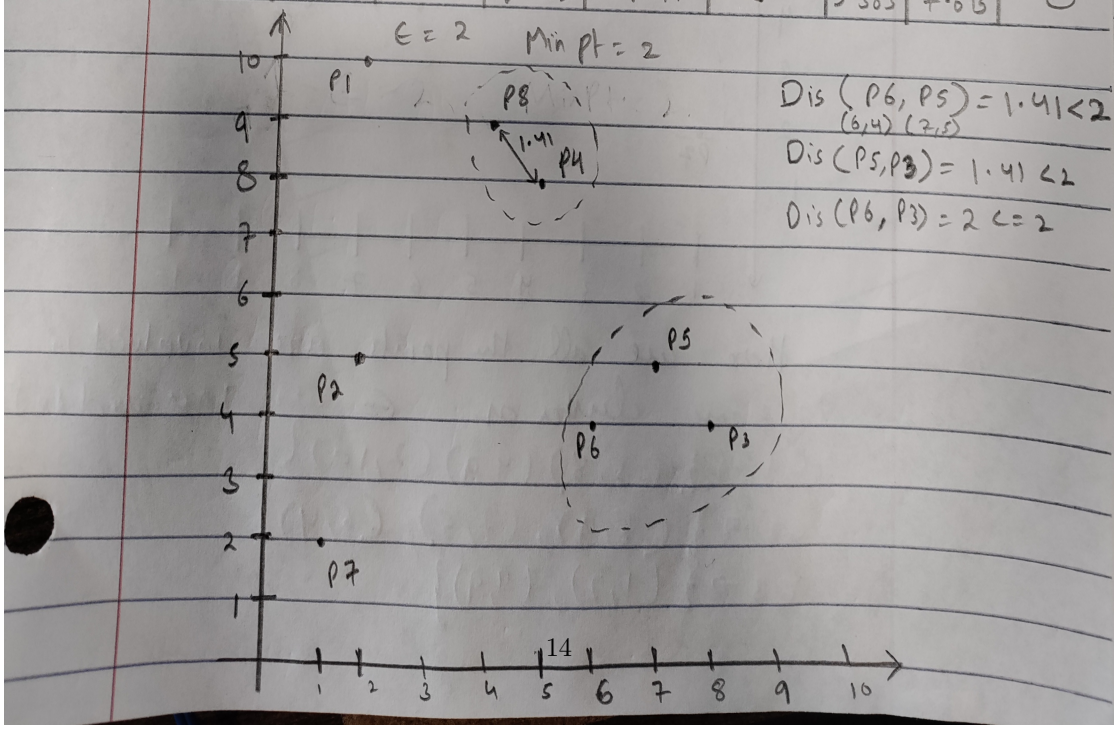
{(2,10); (2,5); (8,4); (5,8); (7,5); (6,4); (1,2); (4,9)}

Euclidean distance metric

a) If Epsilon is 2 and min_samples is 2, what are the clusters that DBSCAN would discover. Plot the discovered clusters.

Below we calculate the distance from every pt to every other pt.

		(2,10)	(2,5)	(8,4)	(5,8)	(7,5)	(6,4)	(1,2)	(4,9)
P1	(2,10)	0	5	8.485	3.60	7.07	7.21	8.06	2.24
P2	(2,5)	5	0	6.08	4.24	5	4.12	3.16	4.47
P3	(8,4)	8.485	6.08	0	5	1.41	2	7.28	6.403
P4	(5,8)	3.60	4.24	5	0	3.60	4.12	7.21	1.41
P5	(7,5)	7.07	5	1.41	3.60	0	1.41	6.71	5
P6	(6,4)	7.21	4.12	2	4.21	1.41	0	5.385	5.385
P7	(1,2)	8.06	3.16	7.28	7.21	6.71	5.385	0	7.615
P8	(4,9)	2.24	4.47	6.403	1.41	5	5.385	7.615	0

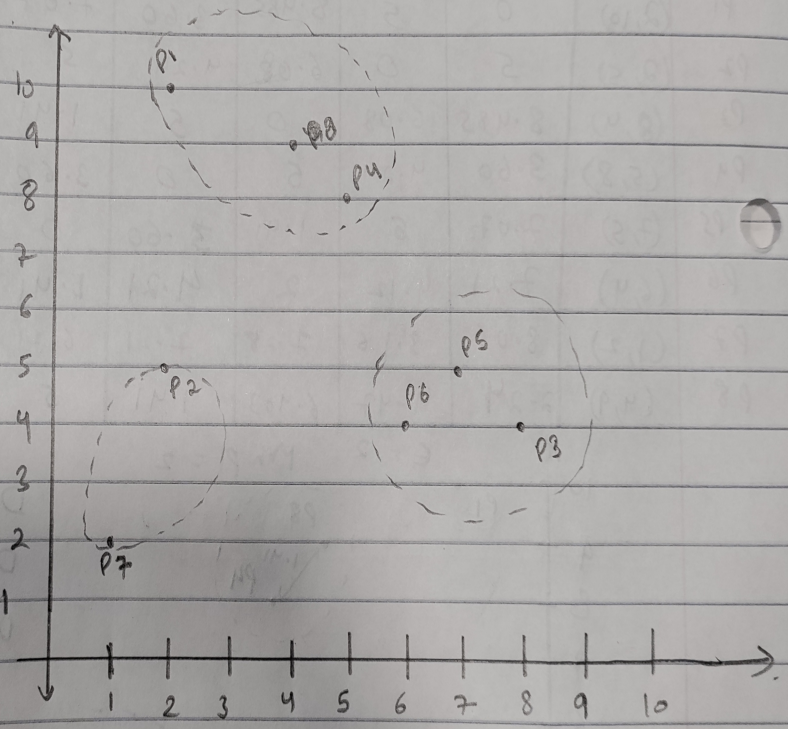


C1 → {(5,8), (4,9)}

C2 → {(8,4), (7,5) (6,4)}

C1 & C2 are the discovered clusters which form a core and rest of the points are noise.

b) What if Epsilon is ↑ to $\sqrt{10} \approx 3.162$



Here all the points are involved in some cluster as ∈ was increased to $\sqrt{10}$

C1 → {(5,8) (4,9) (2,10)}

C2 → {(8,4), (7,5), (6,4)}

C3 → {(1,2), (4,9)}

4 Explain the shortcomings of BFR algorithm and describe how CURE algorithm overcomes the shortcomings.

Problems with BFR Algorithm :

1. Assumes clusters are normally distributed in each dimension.
2. And axes are fixed - ellipses at an angle are not OK.

Advantages of CURE Algorithm

1. Assumes Euclidean distance.
2. Allows clusters to assume any shape.

5 Big Table and Cassandra

1. **There are many ways one might organize the distributed storage of data structured as rows and columns. What design decisions are shared by both Bigtable and Cassandra?**
   In terms of storage : Wide column storage, i.e they stores data in records with an ability to hold large number of dynamic columns.
   Data scheme : Both are schema free.
   Partioning methods : Sharding, i.e related to horizontal partitioning - the practice of separating one's table rows into multiple tables, known as partitions, each table have sames columns and schema but entirely different rows.
   Map Reduce : Supports map reduce.
   They both support immediate consistency i.e for each write operation.

2. **Bigtable uses a "master server" and "tablet servers" to manage the reading and writing of data; Cassandra does not have a distinguished master node. What are the pros and cons of each architecture?**
   Cassandra Pros:
   - Distributed system logic. Multiple data centers and other common network configurations like heterogeneous nodes are handled and exploited well.
   - Strong community with users and project contributors worldwide. The open-source and commercial software people work well together with sharing of lessons learned and improvements based on feedback
   - Multiple datacenters with no or little data loss.
   - Continuous data availability is extremely powerful feature of Cassandra.

   Cassandra Cons:
   - Database event logging not handled properly.
   - Moving data from Cassandra to another relational table is cost heavy.
   - As there is no master so the table doesn't have auto repair feature or rebuild feature.

BigTable Pros:

- Simple Administration.
- It is more scalable because of the partitioning and the master controls the different partitions and once the master is down chubby appoints other partition as the master, and it also ensures that at a time there is only one master.
- Empty cells take no spaces.

BigTable Cons:

- Costly in terms of querying data.
- There is secutiry risks in big table as it deployed on cloud.
- Lack of relational database capabilities.

3. **How is replication handled in each of Bigtable and Cassandra?**
   Cassandra: It stores replicas on multiple nodes to ensure reliability and fault tolerance. A replication strategy determines the nodes where replicas are placed. The total number of replicas across the cluster is referred to as a replication factor. A replication factor of 1 means there is only one copy of each row in the cluster.
   Two replication strategies are available for :

   - **SimpleStrategy**: Use only for a single datacenter and one rack.
   - **NetworkTopologyStrategy**: Highly recommended for most deployments because it is much easier to expand to multiple datacenters when required by future expansion.

   BigTable: It uses chubby for replication management. This service is live when a majority of replicas are running and can communicate with each other. Chubby uses paxos algorithm to keep replicas consistent in the face of failure. The objective of paxos algorithm is to maintain same ordering of commands among multiple replicas so that all replicas eventually converge to the same value. Consensus is achieved through a 2 phase commit protocol which is as follows:

   - **Prepare Phase** : The replica sends majority of replicas, about the change and its associated number.
   - **Commit Phase** : If all replicas accept the message then the proposer sends an accept request to all the replicas and the change is then propogated to all of of them.

4. **What are the main steps in reading and writing data in Bigtable?**
   A virtual complete resource is responsible for reading and wrting data that is associated with series of row key ranges. The data is stored in colossus, GFS. Nodes are given temporarily responsibility for serving various ranges of data based on the operation.

5. **What are the main steps in reading and writing data in Cassandra?**
   Read
   Cassandra processes data at several stages on the read path to discover where data is stored, starting with the data in the memtable and finishing with SSTables.
   Different stages of read processes :

   (a) Check the memTable
   (b) Check if rowCache is enabled.
   (c) Checks Bloom filter.
   (d) Check partition key cache,if enabled.

(e) Goes directly to the compression offset map if a partition key is found in the cache, or checks the partition summary if not. If partition summary is checked, then the partition index is accessed.

- Locates the data on disk using the compression offset map.
- Fetches the data from the SSTable on disk.

Write
Different stages to write process in Cassandra:

- Logging data to commit log.
- Writing data to memTable.
- Flushing data from memTable.
- Storing data on disk in SSTable.

6. **What is the use of Bloom Filter in Cassandra?**
Cassandra uses bloom filters for to determine whether the SSTable has data for particular partition. Bloom filters are unused for range scans, but are used for index scans. Bloom filters are probabilitic sets that allow you to trade memory for accuracy.

7. **How does Cassandra delete data?**
Cassandra treats delete as insert or upsert. The data being deleted is added to the partition in the DELETE command is a deletion marker called a tombstone. The tombstones go through Cassandra's write path, and are written to SSTables on one or more nodes. The key feature of a tombstone: it has a built in expiration time. At the end of expiration time the tombstone is deleted as a part of Cassandra normal compaction process.

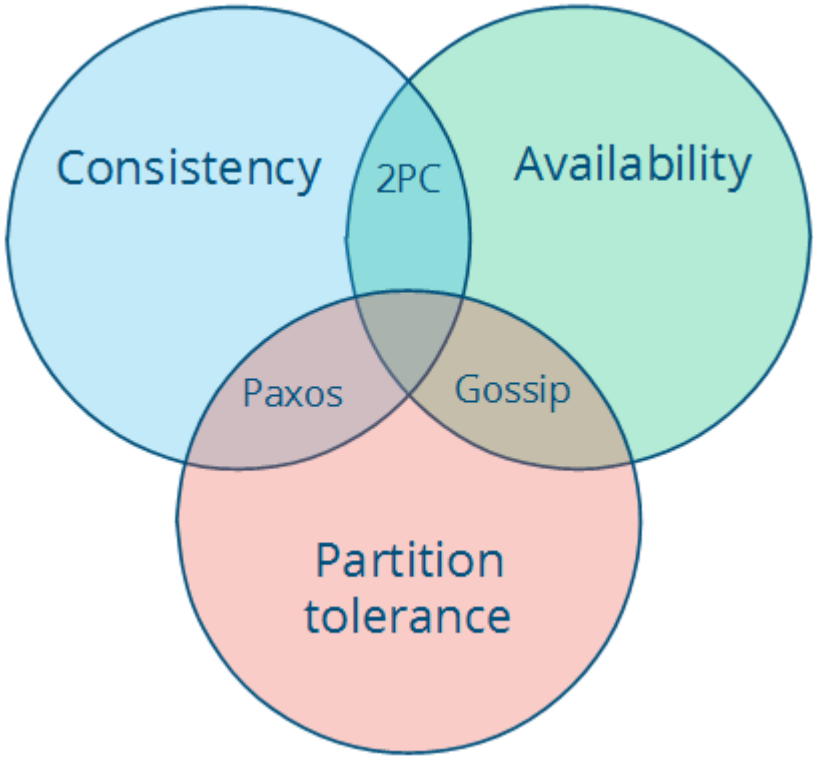
8. **What is SSTable? How is it different from other relational tables?**
Sorted String Tables(SSTable) is a persistent file format which is used by many NoSQl databases to take in memory data stored in memtables, order it for fast access, and stored it on disk in a persistent, ordered, immutable set of files on disk. They are created by memtable flush and deleted by compaction.

9. **Explain CAP theorem.**

CAP theorem a conjecture made by Eric Brewer. The theorem states that of these three properties :

- **Consistency** : All nodes see the same data at the same time.
- **Availability** : Node failure do not prevent survivors from continuing to operate.
- **Partition Tolerance** : The system continues to work despite node failures or network failure.

Only two of three properties can be satisfied simultaneously.



Having all the three properties is non achievable.
The CA and CP system designs both offer the same consistency model: strong consistency. The only difference is that CA system cannot tolerate any node failures, and CP system can tolerate up to f faults given 2f+1 nodes in a Byzantine failure model.
A more modern interpretation of the theorem is during a network partition, a distributed system must choose either Consistency and Availability.