PAGE No.
DATE / /

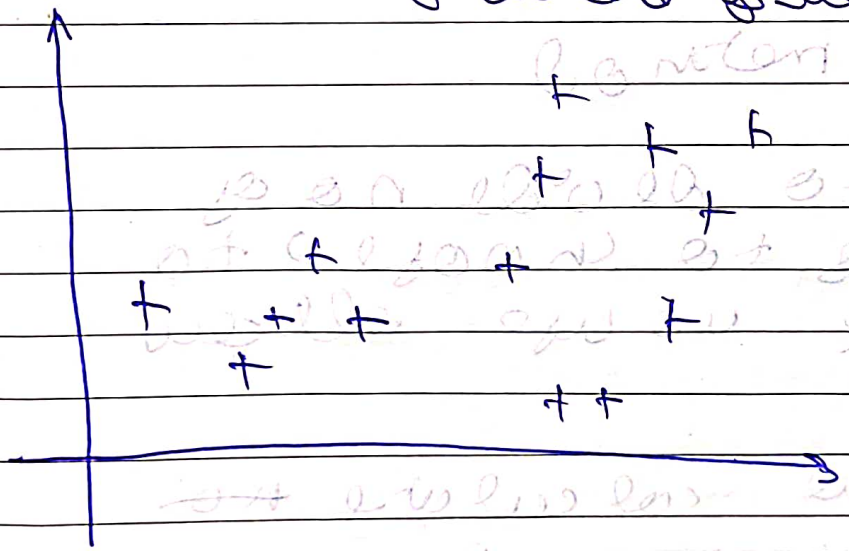
→ Clustering models

* This is part of unsupervised learning in this model is not given labelled data but it is given unlabelled data and model has to group it.

(Model doesn't have category)

→ k-means clustering

(Let say we are given this)



(we will decide how many clusters we want)

PAGE No.
DATE / /

Here we will first

Randomly place 2 centroids (number depend on no of clusters)

Then we assign data to groups based on the centroid

then we calculate center of mass of data points (including centroid) and then we move the centroid there and repeat the process

until this doesn't change anymore or we do this

→ Elbow method

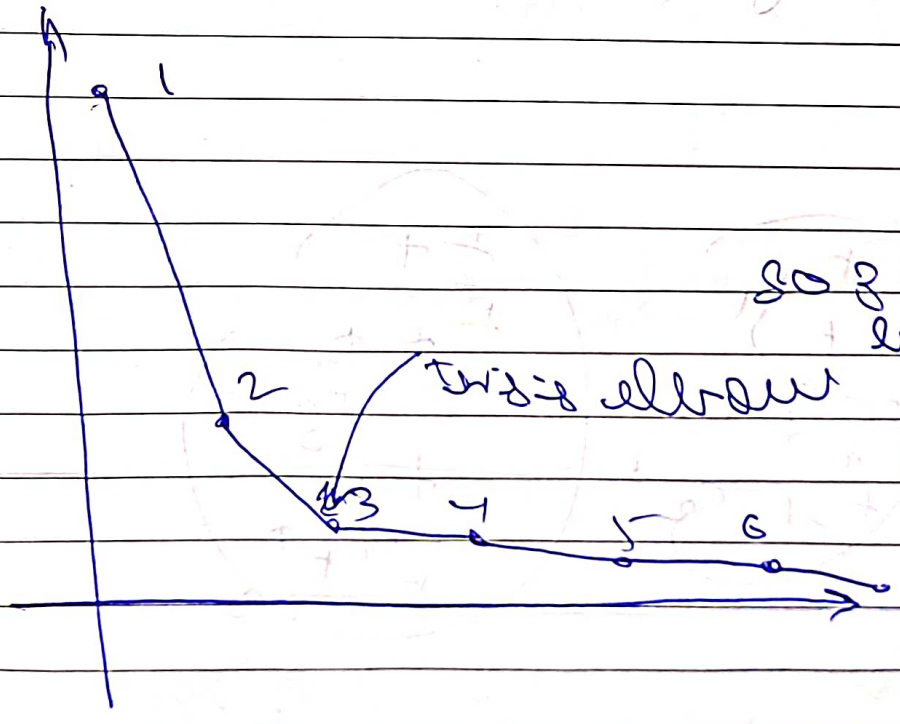
How to decide no of clusters to choose? to do this we use elbow method

Ex: we calculate no of clusters by WCSS score by this formula

PAGE No.
DATE / /

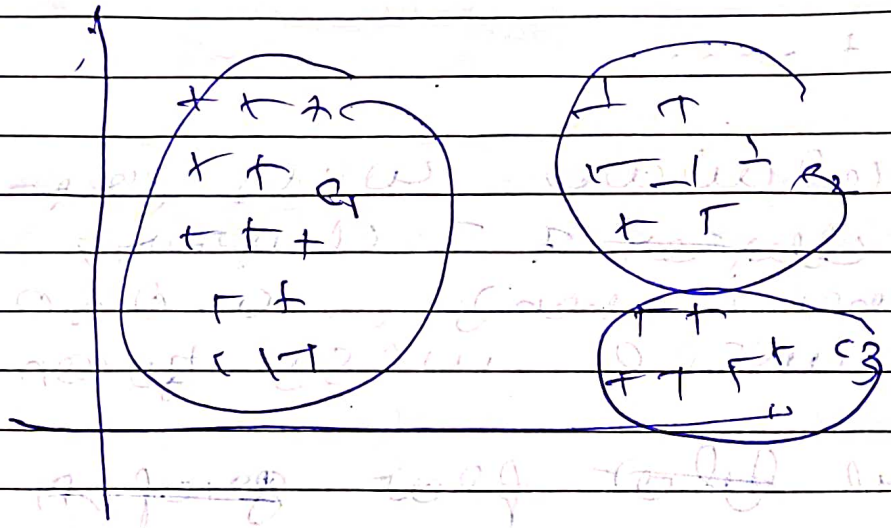
$$WCSS = \sum_{P_i \text{ in cluster 1}} dist(P_i, C_1)^2 + \sum_{P_i \text{ in cluster 2}} dist(P_i, C_2)^2 + \dots$$

and calculate WCSS for-
for ~~cluster 1~~ 1 cluster,
then 2 then 3 ... and so
on until WCSS becomes
then we plot ~~for~~ ~~graph~~
graph

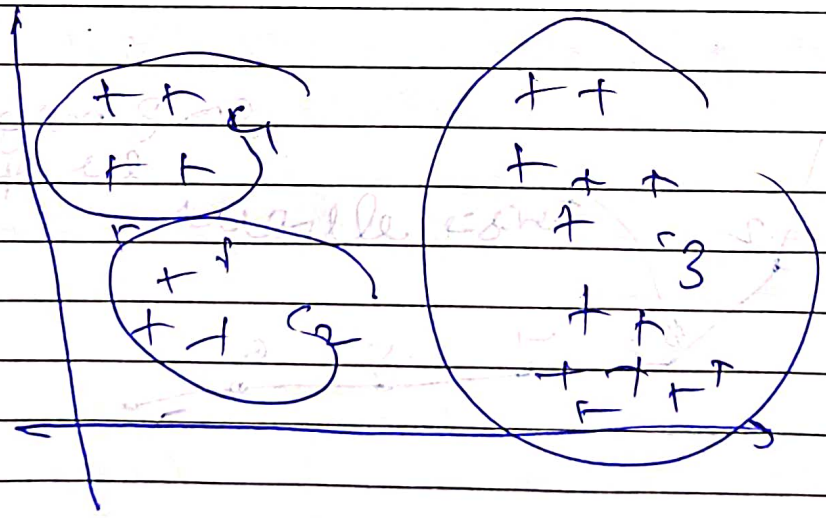


→ k → means ++

if we have data as



But if centroids are initialized as



we have different results

PAGE No.
DATE / /

Random initialization trap

In k means ++

1) first we choose centroid at Random

2) we compute difference (D) to the nearest one of already selected centroids

3) we choose the next centroid using weighted Random selection (weighted by D2)

4) Repeat 2 and 3 until all the centroids are selected

5) apply k-means

PAGE No.
DATE / /

→ Hierarchical clustering

There are 2 types

1] agglomerative (Bottom up)

2] Divisive (covered in ~~class~~ class)

1] first make each datapoint a single cluster (N clusters)

2] take two closest data points and make them one cluster (N-1 clusters)

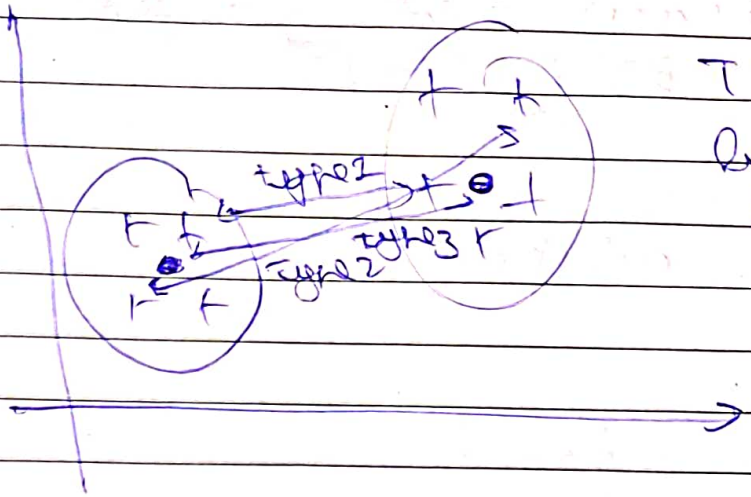
3] take two closest clusters and make them one (N-2 clusters)

Repeat steps until only one cluster

FIN

[we can use any distance but proper explanation]

How to measure distance between 2 centroids



T1 = dist between 2 corresponding points

T2 distance between furthest points

T3:- distance between centroids

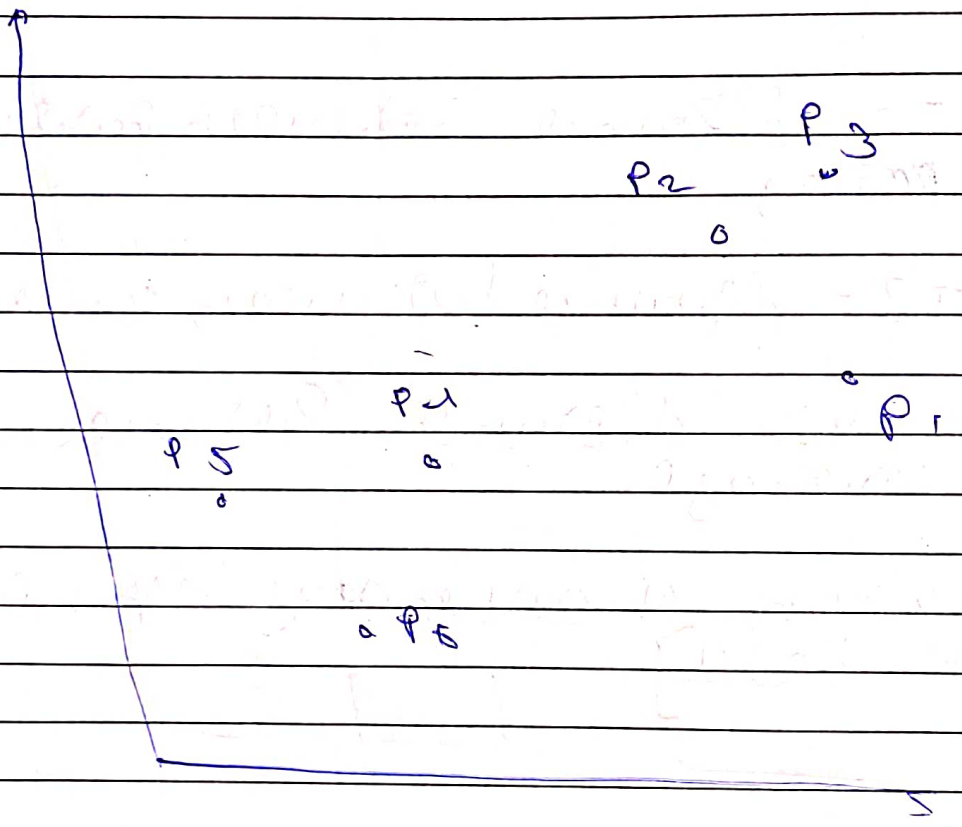
or T4:- distance between angle

[We can choose any depending on data]

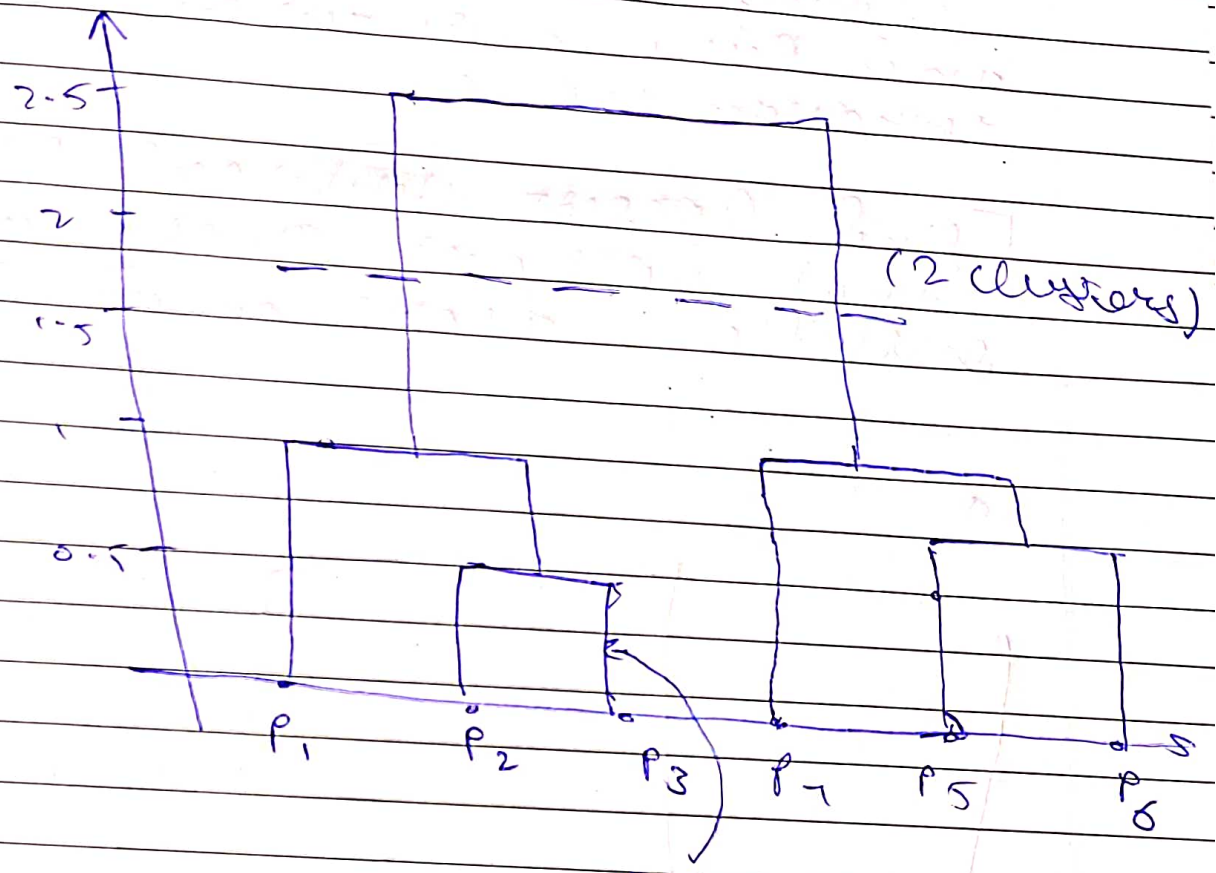
→ Banker's algorithm

In 7-IC the computer maintains the memory flow the algorithm was performed

Suppose we have



Dendogram



Represents the dissimilarity / distance between points

we decide a threshold of dissimilarity

(No of lines threshold crosses are the no of clusters)

PAGE No.
DATE / /

we find the largest distance (largest dist similarity) and keep our threshold through there

[find longest vertical line that doesn't cross any extended horizontal line]

⇒ Q