Unsupervised learning → Only input data (X) and no corresponding output variable.

Clustering – Clustering problem is where we want to discover the inherent grouping in the data, such as grouping customers by purchase.

- Finding subgroups / clusters in a dataset. Cluster the observations in a dataset / into distinct groups so that observations within each group are quite similar to each other.

Practical issue in clustering → i) Observations should be in same scale.

ii) Validating the clusters obtained. Clusters we found represent true subgroup/noise.

iii) Robustness of the clusters iv) Clusters may be distorted due to outliers.

v) Highly dependent on number of K

Kmeans → i) K represents number of clusters to be found in the data.

ii) It is also known as hard clustering because every data point does not present in multiple clusters, making clusters unique.

iii)

Steps in Kmeans – i) Suppose we choose number of clusters = 2. K = 2. algorithm chooses two random points as centroids

ii) Now to form two groups from set of data, algorithm chooses two random points as centroids and computes euclidean distances from centroid to all other datapoints.

iii) Algorithm after measuring the distance of all data points, Kmeans algo works in iterations as now it updates the centroid by making mean.

iv) Will repeat above steps until, no data points changes the cluster upon updating the centroids

Examples of steps – Lets consider 6 datapoints.

	D_1	D_2	P_3	D_4	D_5	D_6
X	1	2	3	7	8	9
Y	1	2	3	7	8	9

i) Select 2 random datapoints → D_2, D_5.

D_2 as Cluster 1 and P_5 as Cluster 2

$= \sqrt{(2-1)^2 + (2-1)^2} = \sqrt{2} = 1.41$

ii) Euclidean distance $= \sqrt{(x_2-x_1)^2 + (y_2-y_1)^2}$

Data points			Centroid 1			Centroid 2.			Assign clusters
D	X	Y	X	Y	Distance from C_1	X	Y	Distance from C_2	Clusters
D_1	1	1	2	2	1.41	8	8	9.89	1
$C_1 \rightarrow D_2$	2	2	2	2	0	8	8	8.48	1
P_3	3	3	2	2	1.41	8	8	7.07	1
D_4	7	7	2	2	7.07	8	8	1.41	2
$C_2 \rightarrow D_5$	8	8	2	2	8.48	8	8	0	2
D_6	9	9	2	2	9.89	8	8	1.41	2

First sample D_1, assigned to cluster 1 as distance of C1 is less than C2.

iii) Update the new centroid by taking mean of data points assigned to each cluster. Cluster 1 = mean of all data point assigned to cluster 1. Cluster 2 also same.

	X	Y
Cluster 1	$\frac{1+2+3}{3} = 2$	$\frac{1+2+3}{3} = 2$
Cluster 2	$\frac{7+8+9}{3} = 8$	$\frac{7+8+9}{3} = 8$.

New Centroids, $C_1 = (2, 2)$
$C_2 = (8, 8)$

iv) Same ~~clus~~ centroids came. If different comes, then again calculate euclidean distance and keep on reiterating until no clusters labels are reassigned on updating the centroid. stop the process.

Numbers of clusters K →

i) Profiling approach → Identify characteristics of each segment and define 'K.
K takes multiple value, then analyze each clusters & the cluster which give meaningful result is choosen as final.

ii) Elbow method → i) Compute average distance of data points from centroid.
ii) Increase number of centroids, average distance decreases.
iii) use multiple K and plot them of graph. Where there is a elbow, choose the value of K.

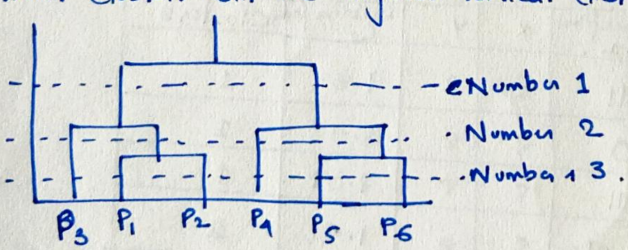
Preprocessing for K means Clustering →

i) Outlier treatment (because distance based technique) ii) Missing value treatment
iii) Rescaling data (scale should be same as it is distance based).
iv) Dimensionality Reduction (higher number of useless dimension make clustering less meaningful).

Hierarchical Clustering →

Dendogram → show hierarchical Relationship between objects

Working - i) Suppose we have 6 data points, so we will have 6 clusters.
ii) Calculate euclidean distance from each clusters (6). Merge smallest 2 euclidean distance. suppose (P_1, P_2) & (P_5, P_6).
iii) Again calculate distance and measure & merge (P_3 (P_1, P_2) & (P_4 (P_5, P_6))
iv) No of clusters = Vertical line crossing the threshold.
Optimal clusters will be highest vertical distance on the dendogram.



Number 1 have 2 clusters.
Number 2 have 4 clusters.
Number 3 have 6 cluster.
Number 1 have highest vertical distance so choose cluster = 2.

Hierarchical clustering → It is also a hard clustering.

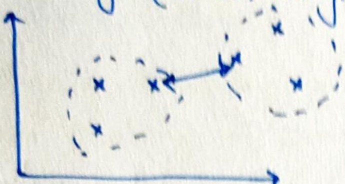
2 type of hierarchical clustering - i) Aglomerative - Bottom up approach. Initially all assigned to different cluster & based on similarity, they merge.

ii) Divisive → Top down approach. Initially all data points are based on one clusters and based on ~~si~~ dissimilarity we divide the clusters into small clusters.

Linkage → In both clustering similarity (agglomerative) or dissimilarity (divisive), we require distance between clusters.

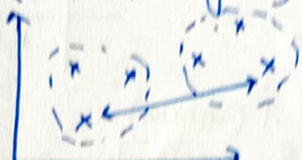
3 types of linkage → i) Single linkage (Nearest Neighbours)
ii) Complete linkage (Farthest Neighbours)
iii) Average linkage.

Single linkage (Nearest Neighbor)



Between two clusters, find the shortest distance between them.

Complete linkage (Farthest Neighbour)



Between two clusters, find the maximum distance between them.

Average linkage
Considers average distance, for this we calculated the average distance from each data point of a cluster to all datapoint of other clusters.

Example of linkage → Suppose use single linkage for hierarchical clustering.

Datapoints,

Data point	D_1	D_2	D_3	D_4	D_5	D_6
x	1	2	3	7	8	9
y	1	2	3	7	8	9

1) So six clusters, & six datapoints
2) Create distance matrix based on euclidean distance $= \sqrt{(x_2-x_1)^2+(y_2-y_1)^2}$

	D_1	D_2	D_3	...	D_7
D_1	0				
D_2		0			
D_3			0	....	
⋮					
D_7					0

3) Merge minimum distance points (D_5, D_6).
4) After merge (D_5, D_6), introduce linkage method.
Suppose we want to calculate distance from D_1. So find distance from D_1 to D_5 and D_1 to D_5 and if we select single linkage, choose the minimum distance and recalculate for others also.
5) Get the optimize clusters through dendogram.

Advantage and disadvantage of hierarchical clustering →
i) Sensitive to noise/outliers   ii) Require standardisation (distance based algo)
iii) Difficult to identify numbers of clusters

Elbow method → i) Total error   ii) Variance / Total squared error
iii) Within cluster sum of square (WSS).

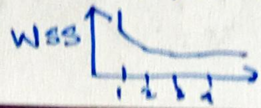
Eg -

Length	(mean (length) - length)	$(\text{error})^2$
1	$(3-1) = 2$	4
2	$(3-2) = 1$	1
3	0	0
4	1	1
5	2	4
mean $= \frac{15}{5} = 3$	Total error = 0	Total square error = 10

Mean variance $= \frac{10}{n-1} = \frac{10}{4} = 5.$

Mean Variance (length) = 5.

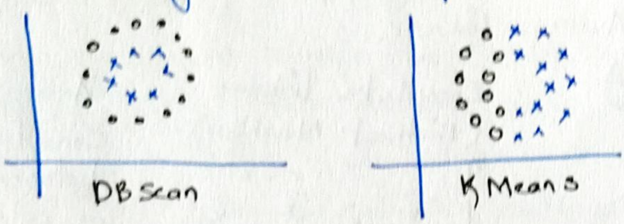
Total Variance is WSS.



Total variance in each cluster is WCSS
Within cluster sum of square

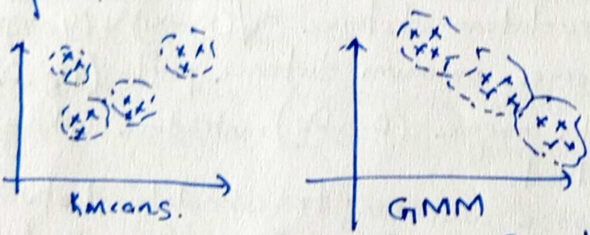
DB Scan → Density based Clustering - Good with Outliers.

- K means and hierarchical clustering work good for compact and well separated clusters. And severly affected by presence of outliers & noise.
- DBScan works good with compact like clusters (spherical clustering)



Gausian mixture model (GMM) -

- Probablistic model for representing normal distribution subpopulation with an overall population.
- GMM assume there are certain number of Gaussian distribution and each of these distribution represent a cluster.
- It assume parameters follows normal distribution.
- GMM advantage is K Means weakness. K means will do well when data is quite separated but if data is overlapping, GMM is a good option.



- K means places a circle of each clusters, and it act as a hard cut off. for cluster assignment. Any point outside the circle is not consider a member of clusters.

- GMM address this issue, since it is probablistic model.

Expectation - Minimization (EM) algorithim →

- EM is a statistical algorithim for finding the right model parameters
- We use EM when data has missing value / data is incomplete.

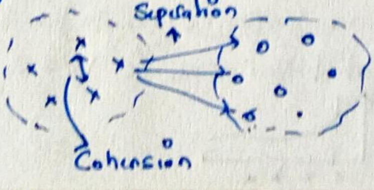
EM has two steps -

i) E-step → In this step, available data is used to estimate (guess) the value of missing variable.

ii) M-step → Based on estimated values, generated in Estep the data is used to update parameters.

Cluster Validation → i) Cluster cohension (Compactness / ~~big~~ tightness)
(Check randomness) ii) Cluster separation (isolation, how well data points are seprated from each other)

K Means have silhoutee values.



i) Silhoutee cofecient value ranges from [-1,1]

ii) -1 is clustering is wrong. 0 both cluster are same. +1 clusters are different.

Steps in Silhoutte – i) Create distance matrix, euclidean distances.

ii) For each point x, calculate a) Cohension, Intra cluster dist
b) Seperation, Inter cluster dist

iii) Silhoutte coefficient = Seperation – Cohension

Many time we get –ve value, normalize = $\frac{\text{Seperation} - \text{Cohension}}{\text{Max(Seperation/Cohension)}}$

iv) Value of Silhoutte coefficient close to 1 indicates objects are well clustered.
Value of close to –1 suggest objects is poorly clustered.

Disadvantage of Clustering →

i) K Means Clustering – i) Choose K Manually ii) Work only good with well seperated clusters.
iii) Distance based model iv) Outliers, different scale.
v) Lack of probablistic cluster management.

ii) Hierarchical clustering – i) If we have large dataset, become difficult to determine correct number of clusters by dendogram.
ii) Sensitive to noise.

iii) DB Scan – i) Work well with seperating high density clusters with low density clusters.
ii) Suffer badly with high dimension data.

iv) Gaussian (EM) clustering – i) Does not work if data do not follow normal distribution.

When to use which clustering –

i) Hierarchical Clustering → When the data is small. Easy to visualize.

ii) K Means Clustering → Well seperated data, not so sphenical data.

iii) DB Scan → works well with sphenical data, have outliers in the data, data are in orbitrary shape but extremely accurate. It determine numbers of clusters automatically.

iv) Gaussian clustering – If data follow normal distribution & data overlap.

~~Assumption in clustering~~

Euclidean distance → $\sqrt{(x_2-x_1)^2+(y_2-y_1)^2}$

Mahatten distance → $|x_2-x_1|+|y_2-y_1|$