UNIT V

CHAPTER 5

Unsupervised Learning

Syllabus

K-Means, K-medoids, Hierarchical, and Density-based Clustering, Spectral Clustering. Outlier analysis: introduction of isolation factor, local outlier factor.

Evaluation metrics and score: elbow method, extrinsic and intrinsic methods

5.1 UNSUPERVISED LEARNING : K MEANS CLUSTERING

GQ. Explain clustering in unsupervised learning.

- In unsupervised learning the most important task is the Clustering. Clustering is used to store data points in to related groups. In clustering advance knowledge is not present about the group definitions.

Definition : "Clustering is a process of partitioning a set of data in a set of meaningful sub-classes, called as clusters".

- In clustering we group the "similar" objects in one cluster and "dissimilar" objects in another cluster.

K- means Clustering

- To solve the well known clustering problem K-means is used, which is one of the simplest unsupervised learning algorithms.
- Given data set is classified assuming some prior number of clusters through a simple and easy procedure. In k- means clustering for each cluster one centroid is defined. Total there are k centroids.
- The centroids should be defined in a tricky way because result differs based on the location of centroids. To get the better results we need to place the centroids far away from each other as much as possible.
- Next, each point from the given data set is stored in a group with closest centroid. This process is repeated for all the points. The first step is finished when all points are grouped. In the next step new k centroids are calculated again from the result of the earlier step.
- After finding these new k centroids, a new grouping is done for the data points and closest new centroids. This process is done iteratively.
- The process is repeated unless and until no data point moves from one group to another.
- The aim of this algorithm is to minimize an objective function such as sum of a squared error function. The objective function is defined as follows :

$$J = \sum_{j=1}^{k} \sum_{i=1}^{x} || x_i^j - C_j ||^2$$

- Here $|| x_i^j - C_j ||^2$ shows the selected distance measure between a data point x_i^j and the cluster centre C_j. It is a representation of the distance of the n data points from their respective cluster centers.

UQ. Describe the essential steps of K-means algorithm for clustering analysis. (Ref. - May 15, 5 Marks)

The algorithm is comprises of the following steps :

1. Identify the K centroids for the given data points that we want to cluster.
2. Store each data point in the group that has the nearest centroid.
3. When all data points have been stored, redefine the K centroids.
4. Repeat Steps 2 and 3 until the no data points move from one group to another. The result of this process is the clusters from which the metric to be minimized can be calculated.

- The k-means algorithm does not guarantee the most optimal solution corresponding to global minimum objective function, although it can be proved that the process will always terminate.
- Initial random selection of cluster centers affects the performance of the algorithm. The k-means algorithm is applied for a number of times to reduce this effect.
- Let's assume that n sample data points $x_1, x_2, ..., x_n$ of the same class are present, and we know that the data points belongs to k clusters, $k < n$.
- Let m_i represents the mean of the data points in cluster i. x can be stored in cluster i, if $|| x - m_i ||$ is the minimum of all the k distances.
- The k-means procedure is shown below:
- Select initial values for the means $m_1, m_2, ..., m_k$

Until no data point moves from one group to another

Use the calculated means to group the data points into clusters

For i from 1 to k

Mean of all of the samples for cluster i is used to replace m_i with the

end_for

end_until

- The K-means algorithm is implemented in three steps.
- Iterate until stable (= no data point move group)

1. Determine the centroid coordinate
2. Determine the distance of each data point to the centroid
3. Group the data points based on minimum distance.

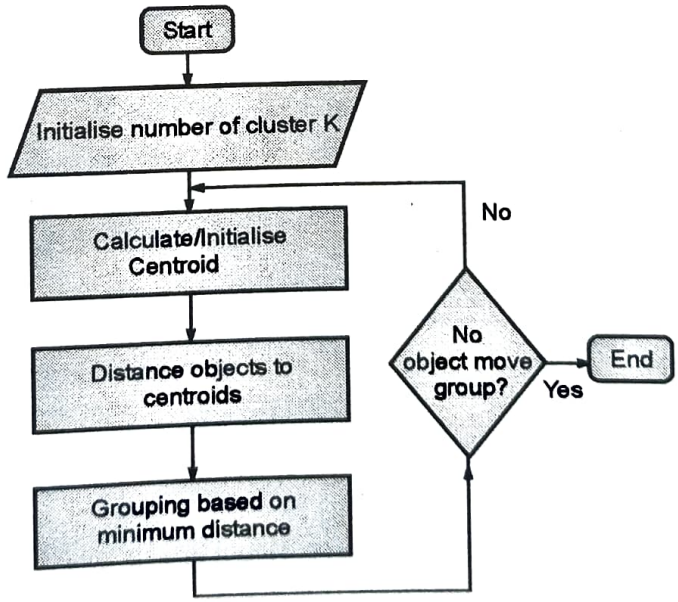


Fig. 5.1.1

5.1.1 Examples on K-means Clustering

Ex. 5.1.1 : Given { 2, 4, 10, 12, 3, 20, 30, 11, 25}. Assume number of clusters i.e. K = 2

☑ **Soln. :**

Randomly assign means : $m_1 = 3$, $m_2 = 4$

The numbers which are close to mean $m_1 = 3$ are grouped into cluster k_1 and others in k_2.

Again calculate new mean for new cluster group.

K_1 = (2, 3}, k_2 = {4, 10, 12, 20, 30, 11, 25} $m_1 = 2.5$, $m_2 = 16$

K_1 = (2, 3, 4}, k_2 = {10, 12, 20, 30, 11, 25} $m_1 = 3$, $m_2 = 18$

K_1 = (2, 3, 4, 10}, k_2 = {12, 20, 30, 11, 25} $m_1 = 4.75$, $m_2 = 19.6$

K_1 = (2, 3, 4, 10, 11, 12}, k_2 = {20, 30, 25} $m_1 = 7$, $m_2 = 25$

Final clusters

K_1 = (2, 3, 4, 10, 11, 12}, k_2 = {20, 30, 25}

Ex. 5.1.2 : Given {10, 4, 2, 12, 3, 20, 30, 11, 25, 31} Assume number of clusters i.e. K = 2

☑ **Soln. :**

Randomly assign alternative values to each cluster

K_1 = (10, 2, 3, 30, 25}, k_2 = {4, 12, 20, 11, 31} $m_1 = 14$, $m_2 = 15.6$

Re assign

K_1 = (2, 3, 4, 10, 11, 12}, k_2 = {20, 25, 30, 31} $m_1 = 7$, $m_2 = 26.5$

Re assign

K_1 = (2, 3, 4, 10, 11, 12}, k_2 = {20, 25, 30, 31} $m_1 = 7$, $m_2 = 26.5$

Final clusters

K_1 = (2, 3, 4, 10, 11, 12}, k_2 = {20, 25, 30, 31}

Ex. 5.1.3 : Let's assume that we have 4 types of items and each item has 2 attributes or features. We need to group these items in to k = 2 groups of items based on the two features.

Object	Attribute 1(x) Number of parts	Attribute 2(y) Colour code
Item 1	1	1
Item 2	2	1
Item 3	4	3
Item 4	5	4

☑ **Soln. :**

Initial value of centroid

Suppose we use item 1 and 2 as the first centroids, $c_1 = (1, 1)$ and $c_2 = (2, 1)$

The distance of item 1 = (1, 1) to $c_1 = (1, 1)$ and with $c_2 = (2, 1)$ is calculated as,

$$D = \sqrt{(1-1)^2 + (1-1)^2} = 0$$

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

The distance of item 2 = (2, 1) to c_1 = (1, 1) and with c_2 = (2, 1) is calculated as,

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

$$D = \sqrt{(2-2)^2 + (1-1)^2} = 0$$

The distance of item 3 = (4, 3) to c_1 = (1, 1) and with c_2 = (2, 1) is calculated as,

$$D = \sqrt{(4-1)^2 + (3-1)^2} = 3.61$$

$$D = \sqrt{(4-2)^2 + (3-1)^2} = 2.83$$

The distance of item 4 = (5, 4) to c_1 = (1, 1) and with c_2 = (2, 1) is calculated as,

$$D = \sqrt{(5-1)^2 + (4-1)^2} = 5$$

$$D = \sqrt{(5-2)^2 + (4-1)^2} = 4.24$$

Objects-centroids distance

$$D^0 = \begin{bmatrix} 0 & 1 & 3.61 & 5 \\ 1 & 0 & 2.83 & 4.24 \end{bmatrix} \begin{matrix} c_1 = (1,1) & \text{group 1} \\ c_2 = (2,1) & \text{group 2} \end{matrix}$$

To find the cluster of each item we consider the minimum Euclidian distance between group1 and group 2.

From the above object centroid distance matrix we can see,

Item 1 has minimum distance for group1, so we cluster item 1 in group 1.

Item 2 has minimum distance for group 2, so we cluster item 2 in group 2.

Item 3 has minimum distance for group 2, so we cluster item 3 in group 2.

Item 4 has minimum distance for group 2, so we cluster item 4 in group 2.

Object Clustering

$$G^0 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 \end{bmatrix}$$

Iteration 1 : Determine centroids

C_1 has only one member thus c_1 = (1, 1) remains same.

$$C_2 = (2 + 4 + 5/3, 1 + 3 + 4/3) = (11/3, 8/3)$$

The distance of item 1 = (1, 1) to c_1 = (1, 1) and with c_2 = (11/3, 8/3) is calculated as,

$$D = \sqrt{(1-1)^2 + (1-1)^2} = 0$$

$$D = \sqrt{(1-11/3)^2 + (1-8/3)^2} = 3.41$$

The distance of item 2 = (2, 1) to c_1 = (1, 1) and with c_2 = (11/3, 8/3) is calculated as,

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

$$D = \sqrt{(2-11/3)^2 + (1-8/3)^2} = 2.36$$

The distance of item 3 = (4, 3) to c_1 = (1, 1) and with c_2 = (2, 1) is calculated as,

$$D = \sqrt{(4-1)^2 + (3-1)^2} = 3.61$$

$$D = \sqrt{(4-11/3)^2 + (3-8/3)^2} = 0.47$$

The distance of item 4 = (5, 4) to c_1 = (1, 1) and with c_2 = (2, 1) is calculated as,

$$D = \sqrt{(5-1)^2 + (4-1)^2} = 5$$

$$D = \sqrt{(5-11/3)^2 + (4-8/3)^2} = 1.89$$

Objects-centroids distance

$$D^2 = \begin{bmatrix} 0 & 1 & 3.61 & 5 \\ 3.41 & 2.36 & 0.47 & 1.89 \end{bmatrix} \begin{matrix} c_1 = (1, 1) & \text{group 1} \\ c_2 = \left(\frac{11}{3}, \frac{8}{3}\right) & \text{group 2} \end{matrix}$$

From the above object centroid distance matrix we can see,

Item 1 has minimum distance for group1, so we cluster item 1 in group 1.

Item 2 has minimum distance for group 1, so we cluster item 2 in group 1.

Item 3 has minimum distance for group 2, so we cluster item 3 in group 2.

Item 4 has minimum distance for group 2, so we cluster item 4 in group 2.

Object Clustering

$$G^1 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

Iteration 2 : Determine centroids

$$C_1 = (1 + 2/2, 1 + 1/2) = (3/2, 1)$$

$$C_2 = (4 + 5/2, 3 + 4/2) = (9/2, 7/2)$$

The distance of item 1 = (1, 1) to c_1 = (3/2, 1) and with c_2 = (9/2, 7/2) is calculated as,

$$D = \sqrt{(1-3/2)^2 + (1-1)^2} = 0.5$$

$$D = \sqrt{(1-9/2)^2 + (1-7/2)^2} = 4.3$$

The distance of item 2 = (2, 1) to c_1 = (3/2, 1) and with c_2 = (9/2, 7/2) is calculated as,

$$D = \sqrt{(2-3/2)^2 + (1-1)^2} = 0.5$$

$$D = \sqrt{(2-9/2)^2 + (1-7/2)^2} = 3.54$$

The distance of item 3 = (4, 3) to c_1 = (3/2, 1) and with c_2 = (9/2, 7/2) is calculated as,

$$D = \sqrt{(4-3/2)^2 + (3-1)^2} = 3.20$$

$$D = \sqrt{(4-9/2)^2 + (3-7/2)^2} = 0.71$$

The distance of item 4 = (5, 4) to c_1 = (3/2, 1) and with c_2 = (9/2, 7/2) is calculated as,

$$D = \sqrt{(5-3/2)^2 + (4-1)^2} = 4.61$$

$$D = \sqrt{(5-9/2)^2 + (4-7/2)^2} = 0.71$$

Objects-centroids distance

$$D^2 = \begin{bmatrix} 0.5 & 0.5 & 3.20 & 4.61 \\ 4.3 & 3.54 & 0.71 & 0.71 \end{bmatrix} \begin{matrix} c_1 = \left(\frac{3}{2}, 1\right) & \text{group 1} \\ c_2 = \left(\frac{9}{2}, \frac{7}{2}\right) & \text{group 2} \end{matrix}$$

From the above object centroid distance matrix we can see,

Item 1 has minimum distance for group1, so we cluster item 1 in group 1.

Item 2 has minimum distance for group 1, so we cluster item 2 in group 1.

Item 3 has minimum distance for group 2, so we cluster item 3 in group 2.

Item 4 has minimum distance for group 2, so we cluster item 4 in group 2.

Object Clustering

$$G^2 = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

$G^2 = G^1$, Objects does not move from group any more. So, the final clusters are as follows:

Item 1 and 2 are clustered in group 1

Item 3 and 4 are clustered in group 2

Ex. 5.1.4 : Suppose we have eight data points and each data point has 2 features. Cluster the data points into 3 clusters using k-means algorithm.

Data points	Attribute 1(x)	Attribute 2(y)
1	2	10
2	2	5
3	8	4
4	5	8
5	7	5
6	6	4
7	1	2
8	4	9

☑ **Soln. :**

Initial value of centroid

Suppose we use data points 1, 4 and 7 as the first centroids, $c_1 = (2, 10)$, $c_2 = (5, 8)$ and $c_3 = (1, 2)$

The distance of data point 1 = (2, 10) to $c_1 = (2, 10)$, $c_2 = (5, 8)$ and with $c_3 = (1, 2)$ is,

$$D = \sqrt{(2-2)^2 + (10-10)^2} = 0$$

$$D = \sqrt{(2-5)^2 + (10-8)^2} = 3.61$$

$$D = \sqrt{(2-1)^2 + (10-2)^2} = 8.06$$

The distance of data point 1 = (2, 5) to $c_1 = (2, 10)$, $c_2 = (5, 8)$ and with $c_3 = (1, 2)$ is,

$$D = \sqrt{(2-2)^2 + (5-10)^2} = 5$$

$$D = \sqrt{(2-5)^2 + (5-8)^2} = 4.24$$

$$D = \sqrt{(2-1)^2 + (5-2)^2} = 3.16$$

The distance of data point 1 = (8, 4) to c_1 = (2, 10), c_2 = (5, 8) and with c_3 = (1, 2) is,

$$D = \sqrt{(8-2)^2 + (4-10)^2} = 8.48$$

$$D = \sqrt{(8-5)^2 + (4-8)^2} = 5$$

$$D = \sqrt{(8-1)^2 + (4-2)^2} = 7.28$$

The distance of data point 1 = (5, 8) to c_1 = (2, 10), c_2 = (5, 8) and with c_3 = (1, 2) is,

$$D = \sqrt{(5-2)^2 + (8-10)^2} = 3.61$$

$$D = \sqrt{(5-5)^2 + (8-8)^2} = 0$$

$$D = \sqrt{(5-1)^2 + (8-2)^2} = 7.21$$

The distance of data point 1 = (7, 5) to c_1 = (2, 10), c_2 = (5, 8) and with c_3 = (1, 2) is,

$$D = \sqrt{(7-2)^2 + (5-10)^2} = 7.07$$

$$D = \sqrt{(7-5)^2 + (5-8)^2} = 3.61$$

$$D = \sqrt{(7-1)^2 + (5-2)^2} = 6.71$$

The distance of data point 1 = (6, 4) to c_1 = (2, 10), c_2 = (5, 8) and with c_3 = (1, 2) is,

$$D = \sqrt{(6-2)^2 + (4-10)^2} = 7.21$$

$$D = \sqrt{(6-5)^2 + (4-8)^2} = 4.12$$

$$D = \sqrt{(6-1)^2 + (4-2)^2} = 5.39$$

The distance of data point 1 = (1, 2) to c_1 = (2, 10), c_2 = (5, 8) and with c_3 = (1, 2) is,

$$D = \sqrt{(1-2)^2 + (2-10)^2} = 8.06$$

$$D = \sqrt{(1-5)^2 + (2-8)^2} = 7.21$$

$$D = \sqrt{(1-1)^2 + (2-2)^2} = 0$$

The distance of data point 1 = (4, 9) to c_1 = (2, 10), c_2 = (5, 8) and with c_3 = (1, 2) is,

$$D = \sqrt{(4-2)^2 + (9-10)^2} = 2.24$$

$$D = \sqrt{(4-5)^2 + (9-8)^2} = 1.4$$

$$D = \sqrt{(4-1)^2 + (9-2)^2} = 7.62$$

Objects-centroids distance

$$D^0 = \begin{bmatrix} 0 & 5 & 8.48 & 3.61 & 7.07 & 7.21 & 8.06 & 2.24 \\ 3.61 & 4.24 & 5 & 0 & 3.61 & 4.12 & 7.21 & 1.4 \\ 8.06 & 3.16 & 7.28 & 7.21 & 6.71 & 5.39 & 0 & 7.62 \end{bmatrix} \begin{matrix} c_1 = (2, 10) & \text{group 1} \\ c_2 = (5, 8) & \text{group 2} \\ c_3 = (1, 2) & \text{group 3} \end{matrix}$$

From the above object centroid distance matrix we can see,

- Data point 1 has minimum distance for group1, so we cluster data point 1 in group 1.
- Data point 2 has minimum distance for group3, so we cluster data point 2 in group 3.
- Data point 3 has minimum distance for group 2, so we cluster data point 3 in group 2.
- Data point 4 has minimum distance for group 2, so we cluster data point 4 in group 2.
- Data point 5 has minimum distance for group 2, so we cluster data point 5 in group 2.
- Data point 6 has minimum distance for group 2, so we cluster data point 6 in group 2.
- Data point 7 has minimum distance for group 3, so we cluster data point 7 in group 3.
- Data point 8 has minimum distance for group 2, so we cluster data point 8 in group 2.

Object Clustering

$$G^0 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

Iteration 1 : Determine centroids

C1has only one member thus $c_1 = (2, 10)$ remains same.

$$C_2 = (8 + 5 + 7 + 6 + 4/5, 4 + 8 + 5 + 4 + 9/5) = (6, 6)$$

$$C_3 = (2 + 1/2, 5 + 2/2) = (1.5, 3.5)$$

Objects-centroids distance

$$D^1 = \begin{bmatrix} 0 & 5 & 8.48 & 3.61 & 7.07 & 7.21 & 8.06 & 2.24 \\ 5.66 & 4.12 & 2.83 & 2.24 & 1.41 & 2 & 6.40 & 3.16 \\ 6.52 & 1.58 & 6.25 & 5.7 & 5.7 & 4.52 & 1.58 & 6.04 \end{bmatrix} \begin{matrix} c_1 = (2, 10) & \text{group 1} \\ c_2 = (6, 6) & \text{group 2} \\ c_3 = (1.5, 3.5) & \text{group 3} \end{matrix}$$

Object Clustering

$$G^1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

Iteration 2 : Determine centroids

$$C_1 = (2 + 4/2, 10 + 9/2) = (3, 9.5)$$

$$C_2 = (8 + 5 + 7 + 6/4, 4 + 8 + 5 + 4/4) = (6.5, 5.25)$$

$$C_3 = (2 + 1/2, 5 + 2/2) = (1.5, 3.5)$$

$$D^2 = \begin{bmatrix} 1.12 & 2.35 & 7.43 & 2.5 & 6.02 & 6.26 & 7.76 & 1.12 \\ 6.54 & 4.51 & 1.95 & 3.13 & 0.56 & 1.35 & 6.38 & 7.68 \\ 6.52 & 1.58 & 6.52 & 5.7 & 5.7 & 4.52 & 1.58 & 6.04 \end{bmatrix} \begin{matrix} c_1 = (3, 9.5) & \text{group 1} \\ c_2 = (6.5, 5.25) & \text{group 2} \\ c_3 = (1.5, 3.5) & \text{group 3} \end{matrix}$$

Object Clustering

$$G^2 = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

Iteration 3 : Determine centroids

$C_1 = (2 + 5 + 4/3, 10 + 9 + 8/3) = (3.67, 9)$

$C_2 = (8 + 7 + 6/3, 4 + 5 + 4/3) = (7, 4.33)$

$C_3 = (2 + 1/2, 5 + 2/2) = (1.5, 3.5)$

$$D^2 = \begin{bmatrix} 1.95 & 4.33 & 6.61 & 1.66 & 5.2 & 5.52 & 7.49 & 0.33 \\ 6.01 & 5.04 & 1.05 & 4.17 & 0.67 & 1.05 & 6.44 & 5.55 \\ 6.52 & 1.58 & 6.52 & 5.7 & 5.7 & 4.52 & 1.58 & 6.04 \end{bmatrix} \begin{matrix} c_1 = (3.67, 9) & \text{group 1} \\ c_2 = (7, 4.33) & \text{group 2} \\ c_3 = (1.5, 3.5) & \text{group 3} \end{matrix}$$

Object Clustering

$$G^3 = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

$G^3 = G^2$, Objects does not move from group any more. So, the final clusters are as follows:

Data points 1, 4 and 8 are clustered in group 1

Data points 3, 5 and 6 are clustered in group 2

Data points 2 and 7 are clustered in group 3

UEx. 5.1.5 **Ref. - May 15, May 16, 10 Marks**

Apply K-means algorithm on given data for k = 3. Use c_1 (2), c_2 (16) and c_3 (38) as initial cluster centres.

Data : 2, 4, 6, 3, 31, 12, 15, 16, 38, 35, 14, 21, 23, 25, 30

☑ **Soln. :**

$c_1 = 2, \quad c_2 = 16, \quad c_3 = 38$

The numbers which are close to mean are grouped into respective clusters.

$k_1 = (2, 4, 6, 3\}, \quad k_2 = \{12, 15, 16, 14, 21, 23, 25\}, \quad k_3 = \{31, 35, 30\}$

Again calculate new mean for new cluster group.

$c_1 = 3.75, \quad c_2 = 18, \quad c_3 = 32$

New clusters

$k_1 = (2, 4, 6, 3\}, \quad k_2 = \{12, 15, 16, 14, 21, 23, 25\}, \quad k_3 = \{31, 35, 30\} \quad c_1 = 3.75, \quad c_2 = 18, \quad c_3 = 32$

Clusters remains unchanged

Final clusters

$K_1 = (2, 4, 6, 3\}, \quad k_2 = \{12, 15, 16, 14, 21, 23, 25\}, \quad k_3 = \{31, 35, 30\}$

5.2 K-MEDOIDS

GQ. Explain the concept of k-medoids.

- The K-medoids problem is a clustering problem similar to K-means.
- The K-medoids algorithms are partitioned. i.e. breaking the dataset up into groups and attempt to minimize the distance between the points in a cluster and a point which is the center of that cluster.
- K-medoids chooses actual data points as centers. The center of the cluster need not be one of the input data points (it is the average between the points in the cluster).

5.2.1 Definiation of Medoid

The medoid of a cluster is defined as the object in the cluster whose average dissimilarity to all the objects in the cluster is minimal, that is, it is a most centrally located point in the cluster.

In general, the k-medoids problem is hard to solve exactly. Hence, many heuristic solutions exist.

5.2.2 Partitioning Around Medoids (PAM)

PAM uses a greedy search which may not find the optimum solution, but it is faster than exhaustive search.

It works as follows :

(1) (BUILD) Initialise : greedily select k of the n data points an the medoids to minimize the cost.

(2) Associate each data point to the closest mediod.

(3) (SWAP) while the cost of the configuration decreases.

(i) For each mediod m, and for each non-medoid data point o :

(a) Consider the swap of m and o, and compute the cost change.

(b) If the cost change is the current best, remember this m and o combination.

(ii) Perform the best swap for m_{best} and o_{best}, if it decreases the cost function.

Otherwise the algorithm terminates.

5.2.3 Run-time of PAM Algorithm

The run-time complexity of the original PAM algorithm per iteration of (3) is $o(k(n-k)^2)$

This run- time can be reduced to $o(n^2)$. This is achieved by splitting the cost change into three parts such that computation can be shared (it is fast PAM).

Other Algorithm

Algorithms other than PAM have also been suggested and is known as "Alternating" heuristic, as it alternates between two optimisation steps.

1. Select initial medoids randomly.

2. Iterate while the cost decreases :
   (i) In each cluster, make the point that minimises the sum of distances within the cluster-medoid.
   (ii) Reassign each point to the cluster defined by the closest medoid determined in the previous step.

5.2.4 Comparison between K-means and k-Medoids Problem

Sr. No.	K mean	k-medoids
1.	k-means is a clustering problem	k-medoids is also clustering problem.
2.	k-mean algorithm partitions the dataset to minisise the distance between the points in the cluster.	k-medoids algorithm also partitions the dataset to minimize the distance between the points in the cluster.
3.	k-means In k-means, centre of a cluster is not necessarily one of the input data points	k-medoids chooses actual data points as centers and allows for accessibility of the input data points.
4.	k-means generally require **Euclidean distance** for efficient solutions	k-medoids can be used with arbitrary dissimilarity measures.
5.	k-means uses Euclidean distance for efficient solution . Hence the effect of noise and outliers cannot be avoided.	k-medoids minimize a sum of pairwise dissimilarities instead of a sum of squared Euclidean distances, it is more robust to noise and outliers.

5.3 UNSUPERVISED LEARNING : HIERARCHICAL CLUSTERING

GQ. Explain the concept of hierarchical clustering.

5.3.1 Hierarchical Clustering

Agglomerative Hierarchical Clustering

- In agglomerative clustering initially each data point is considered as a single cluster. In the next step, pairs of clusters are merged or agglomerated.
- This step is repeated until all clusters have been merged in to a single cluster. At the end a single cluster remains that contains all the data points.
- Hierarchical clustering algorithms works in top-down manner or bottom-up manner. Hierarchical clustering is known as Hierarchical agglomerative clustering.

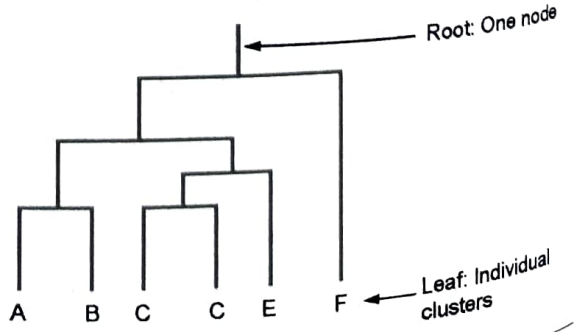


Fig. 5.3.1 : Dendogram

- In agglomerative, clustering is represented as a dendogram as in Fig. 5.3.1 where each merge is represented by a horizontal line.
- A clustering of the data objects is obtained by cutting the dendogram at the desired level, then each connected forms a cluster.
- The basic steps of Agglomerative hierarchical clustering are as follows :

1. Compute the proximity matrix (distance matrix)
2. Assume each data point as a cluster.
3. Repeat
4. Merge the two nearest clusters.
5. Update the proximity matrix
6. Until only a single cluster remains

- In Agglomerative hierarchical clustering proximity matrix is symmetric i.e., the number on lower half will be same as the numbers on top half.
- Different approaches to defining the distance between clusters distinguish the different algorithm's i.e., Single linkage, Complete linkage and Average linkage clusters.
- In single linkage, the distance between two clusters is considered to be equal to shortest distance from any member of one cluster to any member of other cluster.

D (r, s) = Min {d (i, j), object i → cluster r and object j → cluster s

- In complete linkage, the distance between two clusters is considered to be equal to greatest distance from any member of one cluster to any member of other cluster.

D (r, s) = Max {d (i, j), object I → cluster r and object j → cluster s

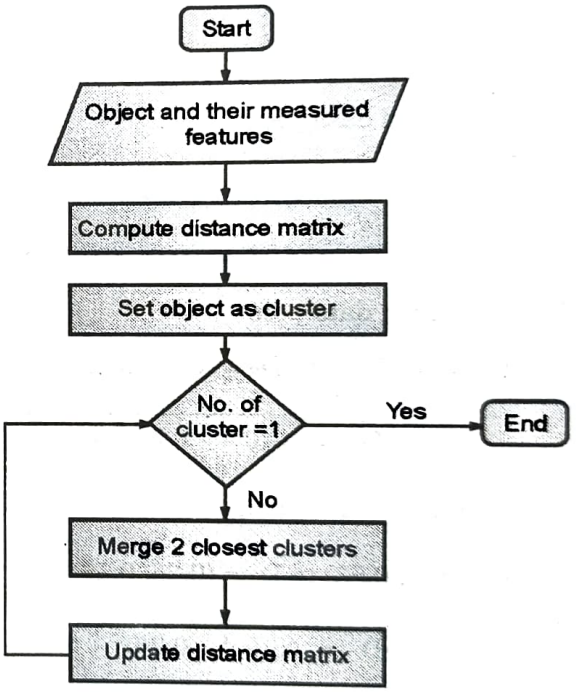


Fig. 5.3.2

In average linkage, we consider the distance between any two clusters A and B is taken to be equal to average of all distances between pairs of object i in A and j in B.ie., mean distance between elements of each other.

D (r, s) = Mean {d (i, j), object i→ cluster r and object j → cluster s

5.3.2 Examples on Hierarchical Clustering

Ex. 5.3.1 : The table shows the six data points. Use all link methods to find clusters. Use Euclidian distance measure.

	X	y
D_1	0.4	0.53
D_2	0.22	0.38
D_3	0.35	0.32
D_4	0.26	0.19
D_5	0.08	0.41
D_6	0.45	0.30

☑ **Soln. :**

First we will solve using single linkage

The distance of data point

$D_1 = (0.4, 0.53)$ to $D_2 = (0.22, 0.38)$ is,

$$D = \sqrt{(0.4 - 0.22)^2 + (0.53 - 0.38)^2} = 0.24$$

The distance of data point

$D_1 = (0.4, 0.53)$ to $D_3 = (0.35, 0.32)$ is,

$$D = \sqrt{(0.4 - 0.35)^2 + (0.53 - 0.32)^2} = 0.22$$

The distance of data point

$D_1 = (0.4, 0.53)$ to $D_4 = (0.26, 0.19)$ is,

$$D = \sqrt{(0.4 - 0.26)^2 + (0.53 - 0.19)^2} = 0.37$$

The distance of data point

$D_1 = (0.4, 0.53)$ to $D_5 = (0.08, 0.41)$ is,

$$D = \sqrt{(0.4 - 0.08)^2 + (0.53 - 0.41)^2} = 0.34$$

The distance of data point

$D_1 = (0.4, 0.53)$ to $D_6 = (0.45, 0.30)$ is,

$$D = \sqrt{(0.4 - 0.45)^2 + (0.53 - 0.30)^2} = 0.23$$

Similarly we will calculate all distances.

Distance matrix

D_1	0					
D_2	0.24	0				
D_3	0.22	0.15	0			
D_4	0.37	0.20	0.15	0		
D_5	0.34	0.14	0.28	0.29	0	
D_6	0.23	0.25	0.11	0.22	0.39	0
	D_1	D_2	D_3	D_4	D_5	D_6

0.11 is smallest. D_3 and D_6 have smallest distance. So, we combine this two in one cluster and recalculate distance matrix.

Distance $((D_3, D_6), D_1)$ = min (distance (D_3, D_1),

distance (D_6, D_1)) = min (0.22, 0.23) = 0.22

Distance $((D_3, D_6), D_2)$ = min (distance (D_3, D_2),

distance (D_6, D_2)) = min (0.15, 0.25) = 0.15

Distance $((D_3, D_6), D_4)$ = min (distance (D_3, D_4),

distance (D_6, D_4)) = min (0.15, 0.22) = 0.15

Distance $((D_3, D_6), D_5)$ = min (distance (D_3, D_5),

distance (D_6, D_5)) = min (0.28, 0.39) = 0.28

Similarly we will calculate all distances.

Distance matrix

D_1	0				
D_2	0.24	0			
(D_3, D_6)	0.22	0.15	0		
D_4	0.37	0.20	0.15	0	
D_5	0.34	0.14	0.28	0.29	0
	D_1	D_2	(D_3, D_6)	D_4	D_5

0.14 is smallest. D_2 and D_5 have smallest distance. So, we combine this two in one cluster and recalculate distance matrix.

Distance $((D_3, D_6), (D_2, D_5))$ = min (distance (D_3, D_2),

distance (D_6, D_2), distance (D_3, D_5), distance (D_6, D_6))

= min (0.15, 0.25, 0.28, 0.29) = 0.15

Similarly, we will calculate all distances.

Distance matrix

D_1	0			
(D_2, D_5)	0.24	0		
(D_3, D_6)	0.22	0.15	0	
D_4	0.37	0.20	0.15	0
	D_1	(D_2, D_5)	(D_3, D_6)	D_4

0.15 is smallest. (D_2, D_5) and (D_3, D_6) as well as D_4 and (D_3, D_6) have smallest distance. We can pick either one.

Distance matrix

D_1	0		
(D_2, D_5, D_3, D_6)	0.22	0	
D_4	0.37	0.15	0
	D_1	(D_2, D_5, D_3, D_6)	D_4

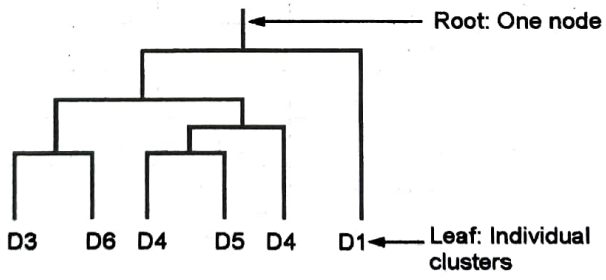
0.15 is smallest. (D_2, D_5, D_3, D_6) and D_4 have smallest distance. So, we combine this two in one cluster and recalculate distance matrix.

Distance matrix

D_1	0	
$(D_2, D_5, D_3, D_6, D_4)$	0.22	**0**
	D_1	$(D_2, D_5, D_3, D_6, D_4)$

Now a single cluster remains $(D_2, D_5, D_3, D_6, D_4, D_1)$

Next, we represent the final dendogram for single linkage as,



Now we will solve using complete linkage

Distance matrix

D_1	0					
D_2	0.24	0				
D_3	0.22	0.15	0			
D_4	0.37	0.20	0.15	0		
D_5	0.34	0.14	0.28	0.29	0	
D_6	0.23	0.25	0.11	0.22	0.39	0
	D_1	D_2	D_3	D_4	D_5	D_6

0.11 is smallest. D_3 and D_6 have smallest distance. So, we combine this two in one cluster and recalculate distance matrix.

Distance $((D_3, D_6), D_1)$ = max (distance (D_3, D_1),

distance (D_6, D_1)) = max (0.22, 0.23) = 0.23

Similarly, we will calculate all distances.

Distance matrix

D_1	0				
D_2	0.24	0			
(D_3, D_6)	0.23	0.25	0		
D_4	0.37	0.20	0.22	0	
D_5	0.34	0.14	0.39	0.29	0
	D_1	D_2	(D_3, D_6)	D_4	D_5

0.14 is smallest. D_2 and D_5 have smallest distance. So, we combine this two in one cluster and recalculate distance matrix.

Distance matrix

D_1	0			
(D_2, D_5)	0.34	0		
(D_3, D_6)	0.23	0.39	0	
D_4	0.37	0.29	0.22	0
	D_1	(D_2, D_5)	(D_3, D_6)	D_4

0.22 is smallest. Here (D_3, D_6) and D_4 have smallest distance. So, we combine these two in one cluster and recalculate distance matrix.

Distance matrix

D_1	0		
(D_2, D_5)	0.34	0	
(D_3, D_6, D_4)	0.37	0.39	0
	D_1	(D_3, D_6, D_4)	(D_3, D_6, D_4)

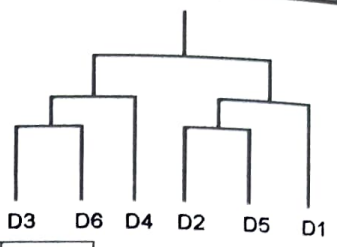
0.34 is smallest. (D_2, D_5) and D_1 have smallest distance so, we combine these two in one cluster and recalculate distance matrix.

Distance matrix

(D_2, D_5, D_1)	0	0
(D_3, D_6, D_4)	0.39	0
	(D_2, D_5, D_1)	(D_3, D_6, D_4)

Now a single cluster remains (D_2, D_5, D_1, D_3, D_6, D_4)

Next, we represent the final dendogram for complete linkage as,



Now we will solve using average linkage

Distance matrix

D_1	0					
D_2	0.24	0				
D_3	0.22	0.15	0			
D_4	0.37	0.20	0.15	0		
D_5	0.34	0.14	0.28	0.29	0	
D_6	0.23	0.25	0.11	0.22	0.39	0
	D_1	D_2	D_3	D_4	D_5	D_6

0.11 is smallest. D_3 and D_6 have smallest distance. So, we combine this two in one cluster and recalculate distance matrix.

Distance ((D_3, D_6), D_1) = 1/2 (distance (D_3, D_1)

+ distance (D_6, D_1)) = 1/2 (0.22 + 0.23) = 0.23

Similarly, we will calculate all distances.

Distance matrix

D_1 .	0				
D_2	0.24	0			
(D_3, D_6)	0.23	0.2	0		
D_4	0.37	0.20	0.19	0	
D_5	0.34	0.14	0.34	0.29	0
	D_1	D_2	(D_3, D_6)	D_4	D_5

0.14 is smallest. D_2 and D_5 have smallest distance. So, we combine this two in one cluster and recalculate distance matrix.

Distance matrix

D_1	0			
(D_2, D_5)	0.29	0		
(D_3, D_6)	0.22	0.27	0	
D_4	0.37	0.22	0.15	0
	D_1	(D_2, D_5)	(D_3, D_6)	D_4

(D_3, D_6) and D_4 have smallest distance. So, we combine this two in one cluster and recalculate distance matrix.

Distance matrix

D_1	0		
(D_2, D_5)	0.24	0	
(D_3, D_6, D_4)	0.27	0.26	0
	D_1	(D_2, D_5)	(D_3, D_6, D_4)

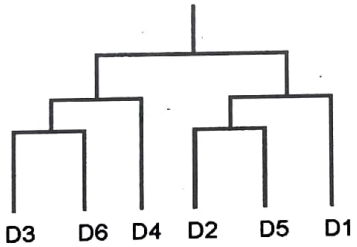
0.24 is smallest. (D_2, D_5) and D_1 have smallest distance. So, we combine this two in one cluster and recalculate distance matrix.

Distance matrix

(D_2, D_5, D_1)	0	0
(D_3, D_6, D_4)	0.26	0
	(D_2, D_5, D_1)	(D_3, D_6, D_4)

Now a single cluster remains $(D_2, D_5, D_1, D_3, D_6, D_4)$

Next, we represent the final dendogram for average linkage as,



Ex. 5.3.2 : Apply single linkage, complete linkage and average linkage on the following distance matrix and draw dendogram.

☑ **Soln. :**

First we will solve using single linkage

Distance matrix

P_1	0				
P_2	2	0			
P_3	6	3	0		
P_4	10	9	7	0	
P_5	9	8	5	4	0
	P_1	P_1	P_3	P_4	P_5

2 is smallest. P_1 and P_2 have smallest distance. So, we combine this two in one cluster and recalculate distance matrix.

Distance $((P_1, P_2), P_3)$ = min (distance (P, P_3),

distance (P_2, P_3)) = min (6, 3) = 3

Similarly, we will calculate all distances.

☞ **Distance matrix**

(P_1, P_2)	0			
P_3	3	0		
P_4	9	7	0	
P_5	8	5	4	0
	(P_1, P_2)	P_3	P_4	P_5

3 is smallest. (P_1, P_2) and P_3 have smallest distance. So, we combine this two in one cluster and recalculate distance matrix.

Distance $((P_1, P_2, P_3), P_4)$ = min (distance (P_1, P_4),

distance (P_2, P_4), distance (P_3, P_4)) = min (9, 7) = 7

Similarly, we will calculate all distances.

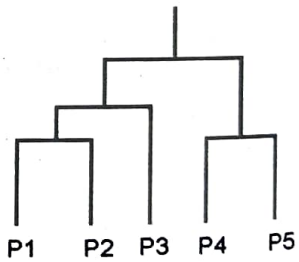
Distance matrix

(P_1, P_2, P_3)	0		
P_4	7	0	
P_5	5	4	0
	(P_1, P_2, P_3)	P_4	P_5

4 is smallest. P_4 and P_5 have smallest distance.

Distance matrix

(P_1, P_2, P_3)	0	
(P_4, P_5)	5	0
	(P_1, P_2, P_3)	(P_4, P_5)



Now a single cluster remains $(P_1, P_2, P_3, P_4, P_5)$

Next, we represent the final dendogram for single linkage as,

Now we will solve using complete linkage

Distance matrix

P_1	0				
P_2	2	0			
P_3	6	3	0		
P_4	10	9	7	0	
P_5	9	8	5	4	0
	P_1	P_2	P_3	P_4	P_5

2 is smallest. P_1 and P_2 have smallest distance. So, we combine this two in one cluster and recalculate distance matrix.

Distance $((P_1, P_2), P_3)$ = max (distance (P_1, P_3),

distance (P_2, P_3)) = max (6, 3) = 6

Similarly, we will calculate all distances.

Distance matrix

(P_1, P_2)	0			
P_3	6	0		
P_4	10	7	0	
P_5	9	5	4	0
	(P_1, P_2)	P_3	P_4	P_5

4 is smallest. P_4 and P_5 have smallest distance. So, we combine this two in one cluster and recalculate distance matrix.

Distance matrix

(P_1, P_2)	0		
P_3	6	0	
(P_4, P_5)	10	7	0
	(P_1, P_2)	P_3	(P_4, P_5)

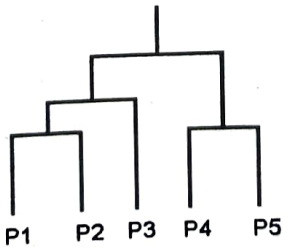
6 is smallest. (P_1, P_2) and P_3 have smallest distance. So, we combine this two in one cluster and recalculate distance matrix.

Distance matrix

(P_1, P_2, P_3)	0	
(P_4, P_5)	10	0
	(P_1, P_2, P_3)	(P_4, P_5)

Now a single cluster remains $(P_1, P_2, P_3, P_4, P_5)$

Next, we represent the final dendogram for complete linkage as,



Now we will solve using average linkage

Distance matrix

P_1	0				
P_2	2	0			
P_3	6	3	0		
P_4	10	9	7	0	
P_5	9	8	5	4	0
	P_1	P_2	P_3	P_4	P_5

2 is smallest. P_1 and P_2 have smallest distance. So, we combine this two in one cluster and recalculate distance matrix.

Distance $((P_1, P_2), P_3)$ = 1/2 (distance (P_1, P_3),

distance (P_2, P_3)) = 1/2 (6, 3) = 4.5

Similarly, we will calculate all distances.

Distance matrix

(P_1, P_2)	0			
P_3	4.5	0		
P_4	9.5	7	0	
P_5	8.5	5	4	0
	(P_1, P_2)	P_3	P_4	P_5

4 is smallest. P_4 and P_5 have smallest distance. So, we combine this two in one cluster and recalculate distance matrix.

Distance matrix

(P_1, P_2)	0		
P_3	4.5	0	
(P_4, P_5)	9	6	0
	(P_1, P_2)	P_3	(P_4, P_5)

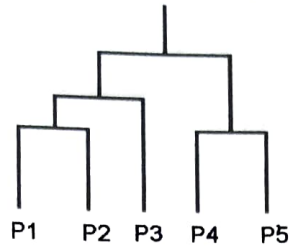
4.5 is smallest. (P_1, P_2) and P_3 have smallest distance. So, we combine this two in one cluster and recalculate distance matrix.

Distance matrix

(P_1, P_2, P_3)	0	
(P_4, P_5)	8	0
	(P_1, P_2, P_3)	(P_4, P_5)

Now a single cluster remains (P_1, P_2, P_3, P_4, P_5)

Next, we represent the final dendogram for average linkage as,



UEx. 5.3.3 **Ref - May 16, 10 Marks**

Apply Agglomerative clustering algorithm on given data and draw dendrogram. Show three clusters with its allocated points. Use single link method.

	A	B	C	D	E	F
A	0	$\sqrt{2}$	$\sqrt{10}$	$\sqrt{17}$	$\sqrt{5}$	$\sqrt{20}$
B	$\sqrt{2}$	0	$\sqrt{8}$	3	1	$\sqrt{18}$
C	$\sqrt{10}$	$\sqrt{8}$	0	$\sqrt{5}$	$\sqrt{5}$	2
D	$\sqrt{17}$	1	$\sqrt{5}$	0	2	3
E	$\sqrt{5}$	1	$\sqrt{5}$	2	0	$\sqrt{13}$
F	$\sqrt{20}$	$\sqrt{18}$	2	3	$\sqrt{13}$	0

☑ **Soln. :**

Distance matrix

A	0					
B	1.414	0				
C	3.162	2.828	0			
D	4.123	1	2.236	0		
E	2.236	1	2.236	2	0	
F	4.472	4.242	2	3	3.6	0
	A	B	C	D	E	F

1 is smallest. B, D and B, E have smallest distance. We can select anyone. So, we combine B, D in one cluster and recalculate distance matrix using single linkage.

Distance matrix

A	0				
B,D	1.414	0			
C	3.162	2.236	0		
E	2.26	1	2.236	0	
F	4.472	3	2	3.6	0
	A	B,D	C	E	F

1 is smallest. B, D and E have smallest distance. So, we combine this two in one cluster and recalculate distance matrix.

Distance matrix

A	0			
B,D,E	1.414	0		
C	3.162	1	0	
F	4.472	3	2	0
	A	B,D,E	C	F

1 is smallest. B, D, E and C are combined together.

Distance matrix

A	0		
B,D,E,C	1.414	0	
F	4.472	2	0
	A	B,D,E,C	F

In the questions three clusters are asked with their allocated points. Three clusters are A, (B, D, E, C) and F.

UEx. 5.3.4 Ref - May 17, 10 Marks

For the given set of points identify clusters using complete link and average link using Agglomerative clustering.

	A	B
P_1	1	1
P_2	1.5	1.5
P_3	5	5
P_4	3	4
P_5	4	4
P_6	3	3.5

☑ **Soln. :**

First we will solve using complete linkage

Distance matrix

p1	0					
P2	0.707	0				
P3	5.656	4.949	0			
P4	3.605	2.915	2.236	0		
P5	4.242	3.535	1.414	1	0	
P6	5.201	2.5	1.802	0.5	1.118	0
	P1	P2	P3	P4	P5	P6

0.5 is smallest. P4 and P6 have smallest distance. We can select anyone. So, we combine this in one cluster and recalculate distance matrix using complete linkage.

Distance matrix

P1	0				
P2	0.707	0			
P3	5.656	4.949	0		
P4,P6	5.201	2.5	1.802	0	
P5	4.242	3.535	1.414	1.118	0
	P1	P2	P3	P4,P6	P5

0.707 is smallest. P1 and P2 have smallest distance. So, we combine this two in one cluster and recalculate distance matrix.

Distance matrix

P1,P2	0			
P3	5.656	0		
P4,P6	5.201	2.236	0	
P5	4.242	1.414	1.118	0
	P1,P2	P3	P4,P6	P5

1.118 is smallest. P4, P6 and P5 are combined together.

Distance matrix

P1,P2	0		
P3	5.656	0	
P4,P5,P6	5.201	2.236	0
	P1,P2	P3	P4,P5,P6

2.236 is smallest. P4, P5, P6 and P3 are combined together.

P1,P2	0	
P3,P4,P5,P6	5.656	0
	P1,P2	P3,P4,P5,P6

Next we will combine all clusters in a single cluster.

Now we will solve using average linkage

Distance matrix

P1	0					
P2	0.707	0				
P3	5.656	4.949	0			
P4	3.605	2.915	2.236	0		
P5	4.242	3.535	1.414	1	0	
P6	5.201	2.5	1.802	0.5	1.118	0
	P1	P2	P3	P4	P5	P6

0.5 is smallest. P4 and P6 have smallest distance. We can select anyone .So, we combine this in one cluster and recalculate distance matrix using complete linkage.

Distance matrix

P1	0				
P2	0.707	0			
P3	5.656	4.949	0		
P4,P6	4.403	2.707	2.019	0	
P5	4.242	3.535	1.414	1.059	0
	P1	P2	P3	P4,P6	P5

0.707 is smallest. P1 and P2 have smallest distance. So, we combine this two in one cluster and recalculate distance matrix.

Distance matrix

P1,P2	0			
P3	5.302	0		
P4,P6	3.55	2.019	0	
P5	3.888	1.414	1.059	0
	P1, P2	P3	P4, P6	P5

1.059 is smallest. P4, P6 and P5 are combined together.

Distance matrix

P1,P2	0		
P3	5.302	0	
P4,P5,P6	3.66	1.817	0
	P1,P2	P3	P4,P5,P6

1.817 is smallest. P4,P5,P6 and P3 are combined together.

P1,P2	0	
P3,P4,P5,P6	4.07	0
	P1,P2	P3,P4,P5,P6

Next we will combine all clusters in a single cluster.

5.4 DENSITY-BASED CLUSTERING

GQ. What is D.B.C ? Explain its working.

- Density-based clustering refers to a method that is based on local cluster criterion such as density connected points. Density-based clustering is an unsupervised learning methology used in model building and machine learning algorithms
- The data points in the region separated by two clusters of low density are considered as noise.

- The surroundings of a radius ϵ of a given object are known as ϵ-neighbourhood of the object.
- If the ϵ-neighbourhood of the object contains at least at least a minimum numbers, Minpts of objects, then it is called a **core object.**.

5.4.1 Background of D-B Clustering

- There are two different parameters to calculate the density-based clustering.

Eps : It is considered as the maximum radius of the neighbourhood

Minpts (i) : Minpts refer to the minimum number of points in an examples neighborhood of that point.

NEps (i) : {K belongs to D and dist (i, k) ≤ Eps}.

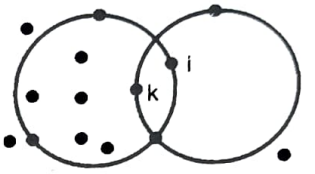
Directly density reachable :

A point i is considered as the directly density reachable from a point k with respect to Eps, Minpts if

i belongs to NEps(k).

Core point condition :

NEps (k) ≥ minimum points



Min pts = 5
Eps = 1 cm

Fig. 5.4.1

5.4.2 Density Reachable

A point denoted by i is a density reachable from a point j with respect to Eps, Minpts if there is a sequence chain of a point i_1, ..., i_n, p_n = i such that i_1 + 1 is directly density reachable from i_1.

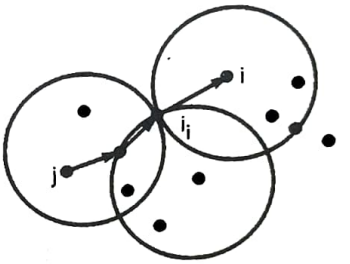


Fig. 5.4.1

5.4.3 Working of Density-Based Clustering

1. Let a set of objects be denoted by D', we can say that an object i is directly density reachable from the object j only if it is located within the ϵ-neighborhood of j, and j is a core object.
2. An object i is density reachable from the object j with respect O ϵ and Minpts in a given set of objects, D' only, if there is a sequence of object chains point i_1, ..., i_n, i_1 = j, p_n = i such that i_i + 1 is directly density reachable from i_i with respect to ϵ and Minpts.
3. An object i is density connected object j with respect to ϵ and Minpts in a give set of objects, D', only if there is an object O belonging to D such that both points i and j are density reachable from O with respect to ϵ and Minpts.

Major Features of Density-Based Clustering

The primary features of density-based clustering are :

(i) It is a scan method,

(ii) It requires density parameters as a termination condition,

(iii) It is used to manage noise in data clusters.

(iv) Density-based clustering is used to identify clusters of arbitrary size.

5.4.4 Density-Based Clustering Methods

(1) DBSCAN

- DBSCAN stands for density-based spatial clustering of applications with noise.
- It depends on a density-based notion of cluster. It also identifies clusters of arbitrary size in the spatial database with outliers.

(2) OPTICS

- OPTICS stands for ordering points to identify the clustering structure. It gives a significant order of database with respect to its density-based clustering structure.
- The order of the cluster contains information equivalent to the density-based clustering related to a long range of parameter settings.
- OPTICS methods are beneficial for both automatic and interactive cluster analysis, including determining an intrinsic clustering structure.

(3) DENCLUE

It enables a compact mathematical description of arbitrarily shaped cluster in high dimension state of data, and it is good for data sets with a huge amount of noise.

5.4.5 Comparison between K-Means and DBSCAN

GQ. Compare kimean of DBSCAN.

Sr. No.	K-Means	DBSCAN
1.	k-means generally cluster all the objects.	DBSCAN discards objects that it defines as nose.
2.	k-means needs a prototype-based concept of a cluster.	DBSCAN needs a density-based concept.
3.	k-means has difficulty with non-globular clusters and clusters of multiple sizes.	DBSCAN is used to handle clusters of multiple sizes and structures and is not powerfully influenced by noise or outliers.
4.	k-means can be used for data that has a definite centroid, including a mean or median.	DBSCAN needs its definition of density, which further depends on the traditional Euclidean concept of density, which is significant for data.
5.	k-means can be used to sparse, high dimensional data, including file data.	DBSCAN generally implements poorly for such information. It is because Euclidean definition of density does not operate well for high dimensional data.
6.	The basic k-means algorithm is similar to a statistical clustering approach. It considers all clusters which come from spherical Gaussian distributions with several means but the equal covariance matrix.	DBSCAN creates no assumption about the distribution of the record.

Use of density-based clustering

The density-based clustering tool works by detecting areas where they are separated by areas that are empty or sparse.

Points that are not part of a cluster are labeled as **Noise**.

5.5 APPLICATIONS OF MACHINE LEARNING

GQ. Mention various applications of machine learning.

UQ. Write short note on : Machine learning applications. (Ref. - May 16, May 17, 10 Marks)

(1) Learning Associations

- A supermarket chain-one an example of retail application of machine learning is basket analysis, which is finding associations between products bought by customers :
- If people who buy P typically also buy Q and if there is a customer who buys Q and does not buy P, he or she is a potential P customer. Once we identify such customers, we can target them for cross-selling.
- In finding an association rule, we are interested in learning a conditional probability of the form P (Q/P) where Q is the product we would like to condition on P, which are the product / products which we know that customer has already purchased.

  P (Milk / Bread) = 0.7
- It implies that 70% of customers who buy bread also buy milk

(2) Classification

- A credit is an amount of money loaned by a financial institution.
- It is important for the bank to be able to predict in advance the risk associated with a loan. Which is the probability that the customer will default and not pay the whole amount back?
- In credit scoring, the bank calculates the risk given the amount of credit and the information about the customer. (Income, savings, collaterals, profession, age, past financial history). The aim is to infer a general rule from this data, coding the association between a customer's attributes and his risk.
- Machine Learning system fits a model to the past data to be able to calculate the risk for a new application and then decides to accept or refuse it accordingly.

  If income > Q_1 and savings > Q_2

  Then low – risk ELES high – risk
- Other classification examples are Optical character recognition, face recognition, medical diagnosis, speech recognition and biometric.

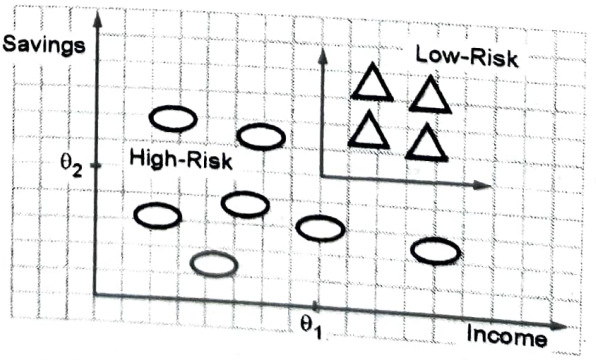


Fig. 5.5.1 : Classification for credit scoring

(3) Regression

- Suppose we want to design a system that can predict the price of a flat.
- Let's take the inputs as the area of the flat, location and purchase year and other information that affects the rate of flat.
- The output is the price of the flat. The applications where output is numeric are regression problems.
- Let X represents flat features and Y is the price of flat. We can collect training data by surveying past purchased transactions and the Machine Learning algorithm fits a function to this data to learn Y as a function of X for the suitable values of W and W_0.

$$Y = w^*x + w_0$$

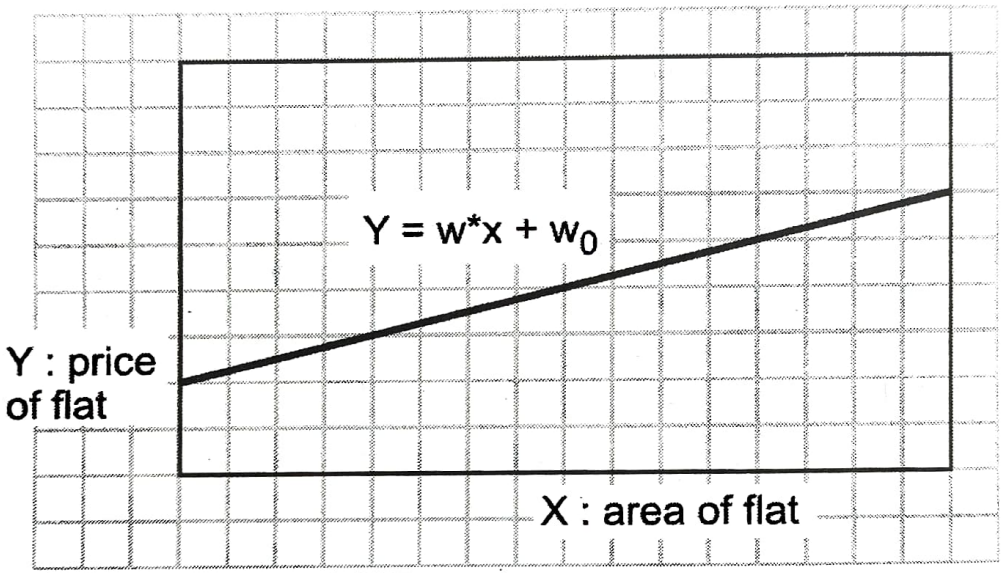


Fig. 5.5.2 : Regression for prediction of price of flat

(4) Unsupervised Learning

- One of the important unsupervised learning problem is clustering. In clustering dataset is partitioned in to meaningful sub classes known as clusters. For example, suppose you want to decorate your home using given items.
- Now you will classify them using unsupervised learning (no prior knowledge) and this classification can be on the basis of color of items, shape of items, material used for items, type of items or whatever way you would like.

(5) Reinforcement Learning

- There are some of the applications where output of system is a sequence of actions. In such applications the sequence of correct actions instead of single action is important in order to reach goal.
- An action is said to be good if it is part of good policy. Machine learning program generates a policy by learning previous good action sequences. Such methods are called reinforcement methods.

- A good example of reinforcement learning is chess playing. In artificial intelligence and machine learning, one of the most important research area is game playing.
- Games can be are easily described but at the same time, they are quite difficult to play well.
- Let's take a example of chess that has limited number of rules, but the game is very difficult because for each state there can be large number of possible moves.
- Another application of reinforcement learning is robot navigation. The robot can move in all possible directions at any point of time.
- The algorithm should reach goal state from an initial state by learning the correct sequence of actions after conducting number of trial runs.
- When the system has unreliable and partial sensory information, it makes reinforcement learning complex. Let's take an example of robot with incomplete camera information. Here robot does not know its exact location.

5.6 GRAPH BASED CLUSTERING

GQ. Explain graph based clustering and its algorithm.

- Graph clustering is an important subject, and deals with clustering with graphs.
- The data of a clustering problem can be represented as a graph where each element to be clustered is represented as a node and the distance between two elements is modeled by a certain weight on the edge linking the nodes.
- Thus in graph clustering, elements within a cluster are connected to each other but have no connection to elements outside that cluster.

5.6.1 Graph Clustering Algorithm

- The HCS (Highly Connected Subgraphs) clustering algorithm (also known as the HCS algorithm, and other names such as highly connected clusters /components /Kernels) is an algorithm based on graph connectivity for cluster analysis.
- It works by representing the similarity data in a **similarity graph**, and then finding all the highly connected subgraphs.
- It does not make any prior assumptions on the number of clusters.
- The HCS algorithm gives a clustering solution, which is inherently meaningful in the application domain.

5.6.2 Method of Graph-based Clustering

(1) **Transform the data into a graph representation :**

(i) Vertices are the data points to be clustered

(ii) Edges are weighted and based on similarity between data points.

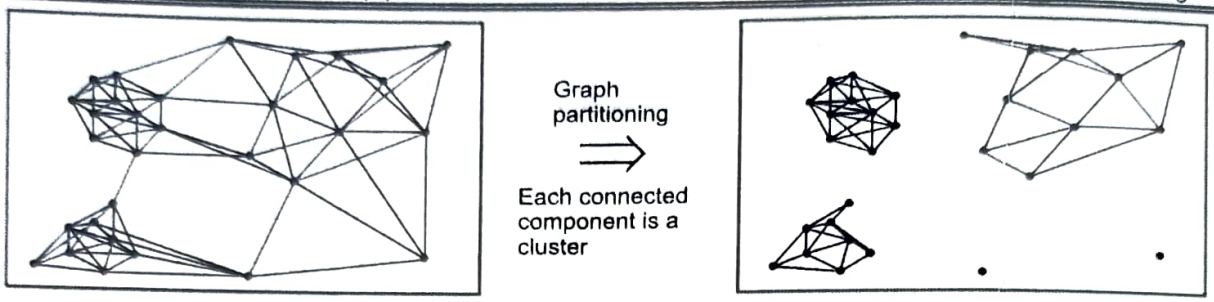


Fig. 5.6.1

(2) Clustering as graph partitioning

Two things are required :

(i) An objective function to determine what would be the best way to 'cut' the edges of a graph.

(ii) An algorithm to find the optimal partition (optimal according to objective function).

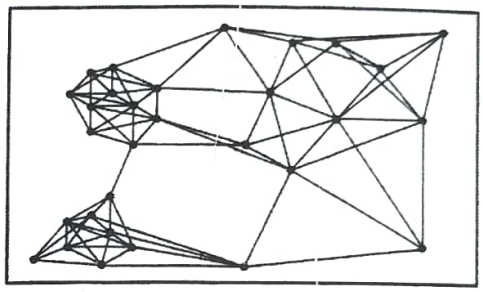


Fig. 5.6.2

(3) Objective function for partitioning

Suppose we want to partition the set of vertices V into two sets : V_1 and V_2. One possible objective function is to minimise graph cut :

Cut $(V_1, V_2) \sum_{i \in V_1, j \in V_2} W_{ij}$; W_{ij} is weight of the edge between nodes i and j

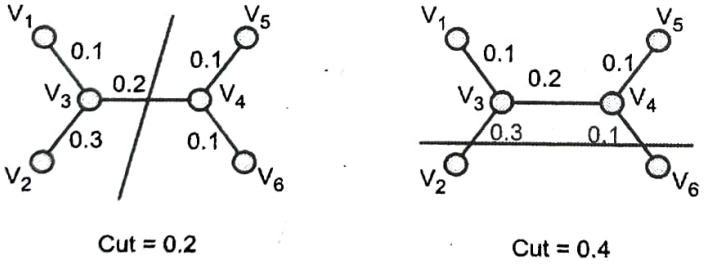


Fig. 5.6.3

(4) **Objective function for partitioning limitation of** minimising graph cut

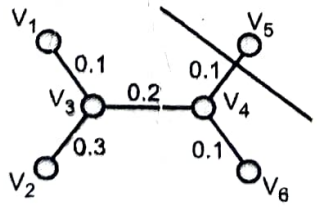


Fig. 5.6.4

The optimal solution might be to split up a single node from the rest of the graph!

Not a desirable solution,

(5) Objective function for partitioning

Our aim is not to only to have 'minimise the graph', but also look for "balanced" clusters.

$$\text{Ratio Cut } (V_1, V_2) = \frac{\text{Cut } (V_1, V_2)}{|\ V_1\ |} + \frac{\text{Cut } (V_1, V_2)}{|\ V_2\ |}$$

$$\text{Normalised Cut } (V_1, V_2) = \frac{\text{Cut } (V_1, V_2)}{\sum_{i \in V_1} d_i} + \frac{\text{Cut } (V_1, V_2)}{\sum_{j \in V_2} d_j}$$

Where $d_i = \sum_j W_{ij}$

V_1 and V_2 are the set of nodes in partitions 1 and 2;

$|\ V_1\ |$ is the number of nodes in partition V_1

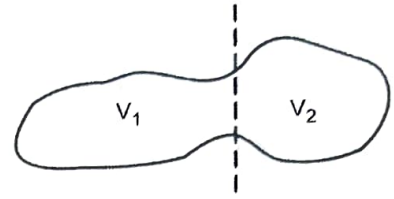
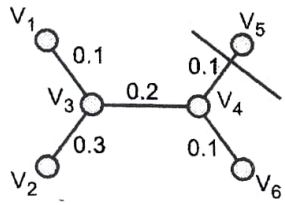


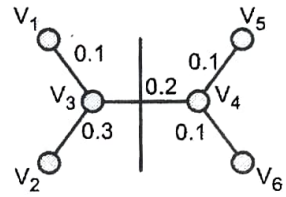
Fig. 5.6.5

(a) Example



Cut = 0.1

Fig. 5.6.6



Cut = 0.2

Fig. 5.6.7

Ratio cut $= \frac{0.1}{1} + \frac{0.1}{5} = 0.12$

Normalised cut

$= \frac{0.1}{0.1} + \frac{0.1}{1.5} = 1.07$

Ratio cut $= \frac{0.2}{3} + \frac{0.2}{3} = 0.13$

Normalised cut

$\frac{0.2}{1} + \frac{0.2}{0.6} = 0.53$

(b) Example : If graph is unweighted (or has to same edge weight).

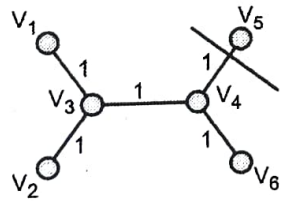


Fig. 5.6.8

Cut = 1

Ratio cut $= \frac{1}{1} + \frac{1}{5} = 1.2$

Normalised cut

$= \frac{1}{1} + \frac{1}{9} = 1.11$

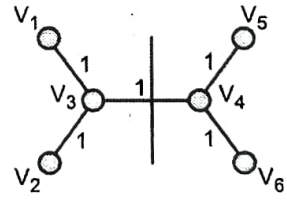


Fig. 5.6.9

Cut = 2

Ratio cut $= \frac{1}{3} + \frac{1}{3} = 0.67$

Normalised cut

$\frac{1}{5} + \frac{1}{5} = 0.2$

(6) Algorithm for graph partitioning

Method of minimising the objective function :

(i) We can use a heuristic (greedy) approach to do this.

(ii) An elegant way to optimise the function is by using ideas from spectral graph theory. This leads to a class of algorithms known as **spectral clustering**.

(7) Spectral clustering

Spectral properties of a graph :

(i) We find eigenvalues and eigenvectors of the adjacency matrix. They can be used to represent a graph.

(ii) There exists a relationship between spectral properties of a graph and the graph partitioning problem.

Method

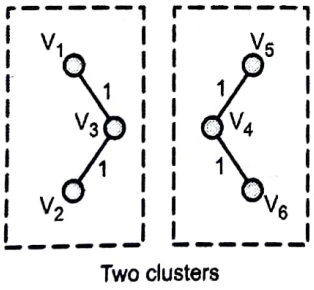
(i) Start with a similarity/adjacency matrix, W, of a graph :

(ii) Define a diagonal matrix D

$$D_{ij} = \begin{cases} \sum_{k=1}^{n} W_{ik} & , \text{ if } i = j \\ 0 & , \text{ otherwise} \end{cases}$$

If W is a binary 0 – 1 matrix, then D_{ii} represents the degree of node i.

☞ **Preliminaries**



$$W = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

Two block-diagonal matrices

Fig. 5.6.10

(8) Graph Laplacian matrix

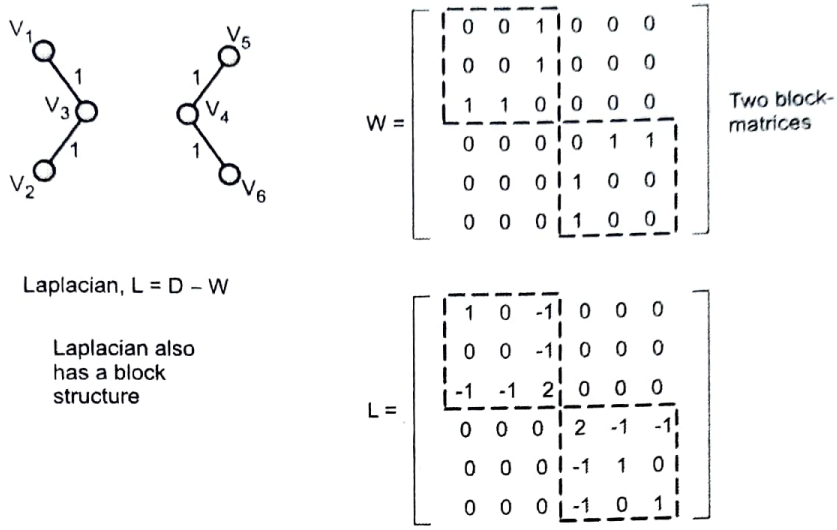


Fig. 5.6.11

(9) Properties of graph Laplacian

(i) $L = (D - W)$ is a symmetric matrix

(ii) L is a positive semi-definite matrix

It means all eigenvalues are non-negative i.e. ≥ 0.

(10) Spectral clustering

Consider a data-set with N data points :

(i) Construct an $N \times N$ similarity matrix W.

(ii) Compute the $N \times N$ Laplacian matrix, $L = D - W$

(iii) Compute the K "Smallest" eigenvectors of L :

(a) Each eigenvector V_1 is an $N \times 1$ Column vector.

(b) Create a matrix V containing eigen vectors $V_1, V_2, \ldots, V_k$ as column (one may exclude the first eigenvector)

(iv) Cluster the rows of V using K-means or other clustering algorithms into K-clusters.

☞ **Example**

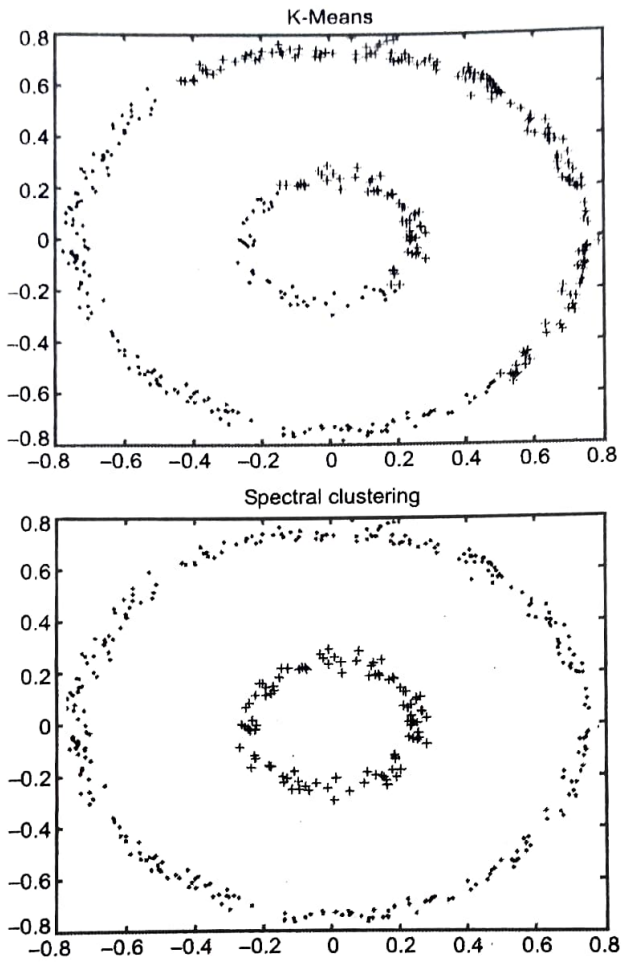


Fig. 5.6.12

☞ **Remark**

Spectral properties of a graph (i.e. eigen-values and eigenvectors) contain information about clustering structure.

5.7 OUTLIER ANALYSIS

GQ. What is outlier analysis.

- Outlier analysis is a fundamental issue in data mining, it is used to detect and remove anomalous objects from data-mining.
- The approach to detect outlier includes three methods. They are (i) clustering, (ii) pruning and (iii) computing outlier score.

(i) For clustering k-means algorithm is used. It partitions the dataset into given number of clusters.

(ii) In pruning points which are closed to centroid of each cluster and pruned. Pruning is based on distance measure.

(iii) For unpruned points, local distance based outlier factor (LDOF) measure is calculated.

- A measure called LDOF, tells how much a point is deviating from its neoighbours.
- The high LDOF value of a point indicates that the point is deviating more from its neighbours and probably it may be an outlier.
- The outlier detection problem in some cases is similar to the classification problem.
- For example, the main concern of clustering based outlier detection algorithms is to find clusters and outliers is to be removed. Because this make more reliable clustering. Outliers are generally regarded as noise.
- Some noisy points may be far away from the data points, whereas the others may be close.
- The far away noisy points affect the result considerably because they are more different from the data points.
- Hence it is desirable to remove the outliers, which are far away from all other points ion cluster.
- The identification of an outlier is affected by various factors, many of them are practical application.
- For example, fraud or criminal deception, will always be a costly problem for many profit organizations. Data mining can minimize some of these losses by making use of massive collections of customer data.
- Using web log files, it becomes possible to recognise fraudulent behaviour, changes in behaviour of customers or faults in systems.
- The typical fault detection can discover exceptions in the amount of money spent, type of item purchased, time and location.
- Another example is a computer security intrusion detection system. It finds outlier patterns as a possible intrusion attempts.
- Intrusion detection corresponds to a suite of techniques that are used to identify attacks against computers and network infrastructures.

5.8 ISOLATION FACATORS

GQ. What are isolation factors ?

- Any data point/observation that deviates significantly from other observations is called an Anomaly/outlier.
- Anomaly detection finds its application in various domains like network intrusion detection, sudden rise/drop in sales etc.

- Isolation factors (IF), similar to Random Forests (or Random Factors) are built on decision trees. It is an unsupervised model, since there are no pre-defined labels.
- In an isolation, randomly sub-sampled data is processed in a tree-structure based on randomly selected features.
- The samples which end up in shorter distances indicate anomalies, as it is easier to separate them from other observations.

How do Isolation factor / Forests work ?

The algorithm starts with the training of the data.

1. When given a dataset, a random subsample of the data is selected.
2. Branching starts by selecting a random feature (from the set of all N features) first. And then branching is done on a random threshold; (i.e. any value in the range of minimum and maximum values of the selected feature).
3. If the value of a data point is less than the selected threshold, it goes to the left branch otherwise to the right. And thus a node is split into left and right branches.
4. This process from step 2 is continued recursively till each data point is completely isolated.

5.8.1 Limitations of Isolation Factor

Isolations are computationally efficient and are very effective in Anomaly detection.

Despite its advantages, there are a few limitations as mentioned below :

1. The final anomaly score depends on the contamination parameter, provided while training the model, contamination has occurred.

   It indicates that we have an idea of what percentage of the data is anomalous before hand to get a better prediction.

2. Also the model suffers from a bias due to the way the branching takes place.

5.9 LOCAL OUTLIER FACTOR

GQ. Explain local outlier factor. Mention its advantages and disadvantages.

Local Outlier Factor (LOF) is an algorithm used for unsupervised outlier detection. It produces an anomaly score that represents data points which are outliers in the data set.

It does this by measuring the local density deviation of a given data point with respect to the data points near it.

Working of LOF

- Local density is determined by estimating distances between data points that are neighbours (k-nearest neighbours).
- So for each data point, local density can be calculated.
- By comparing this we can check which data points have similar densities and which have a lesser density than its neighbours.
- The ones with lesser densities are considered as outliers.
- Firstly, k-distances are distances between points that are calculated for each point to determine their k-nearest neighbours.
- The second closest point is said to be the second nearest neighbour to the point.
- The distance is used to calculate the reachability distance. It is defined as the maximum of the distance between two points and the k-distance of that point.
- Here is an image which represents reachability distance of a point to various neighbours.

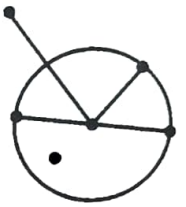


Fig. 5.9.1

- For points inside the circle the k-distance is considered and for points outside the cluster, the distance between points is considered.
- The reachability distance to all of the k-nearest neighbours of a point are calculated to determine the Local Reachability Density (LRD) of that point.
- The local reachability density is calculated by taking the inverse of the sum of all the reachability distances of all the k-nearest neighbouring points.
- So, if the density of the neighbours and the points are almost equal, we say they are quite similar.
- If the density of the neighbours is lesser than the density of the point, the point is inlier, i.e. inside the cluster; and if the density is of the neighbours is move, then the point is outlier.

5.9.1 Advantages of Local Outlier Factor

(i) Sometimes it is difficult to determine outliers. A point that it is at a small distance from a very dence cluster might be considered as an outlier but a point that is at a farthest distance from a wider spread cluster may be considered as an inlier.

With LOR, outliers in local areas are determined, so this does not persist.

(iii) The method in LOF can be applied in many other fields to solve problems of detecting outliers like geographic data, video streams etc.

(iii) The LOF can be used to implement a different dissimilarity function as well. And it is found to outperform many other algorithms of anomaly detection.

5.9.2 Disadvantages of Local Outlier Factor

(i) It is not always the same LOF score that determines whether a point is an outlier or not. It might vary for different data sets.

(ii) In higher dimensions, the LOF algorithm detection accuracy gets effected.

(iii) As LOF score can be any number, it might be a little inconvenient to understand the distinguishing of inliers and outliers based on it.

5.10 EVALUATION METRICS AND SCORE

GQ. Explain different evaluation metrics and score.

To evaluate the models, we consider different kinds of merits. The choice of metric depends on the type of model and the implementation plan of the model.

The following metrics will tell us in evaluating the model's accuracy

Metrics contents

1.	Confusion matrix	2.	F1 Score
3.	Gain and Lift charts	4.	Kolmogorov Smirnov charts
5.	AUC – RUC	6.	Log – Loss
7.	Gini coefficient	8.	Concordant – Discordant ratio
9.	Root mean squared Error	10.	Cross – Validation

1. Confusion matrix

A confusion matrix is an N × N matrix, where N is the number of predicted classes.

We note that the following points

(i) **Accuracy :** The proportion of the total number of predictions that were correct.

(ii) **Positive predictive value or precision :** The proportion of positive cases that were correctly identified.

(iii) **Negative predictive value :** The proportion of negative cases that were correctly identified.

(iv) **Sensitivity or recall :** The proportion of actual positive cases which are correctly identifies.

(v) **Specificity :** The properties of actual negative cases which are correctly identified.

In general, we are concerned with one of the above defined metric. For example, in a pharmaceutical company, concern will be of minimal wrong positive diagnosis. Here it is a concern of high specificities.

2. F1 Score

F1 score is the harmonic mean of precision and recall values for a classification problem.

Formula is : $F1 = 2\left[\frac{(\text{Precison}) \cdot (\text{recall})}{(\text{Precison}) + (\text{recall})}\right]$

We have already discussed F1 score in detail.

3. Gain and Lift charts

Gain and lift chart are concerned to check the rank ordering of the probabilities. We mention the steps to build a Lift/gain chart.

- **Step 1 :** Calculate probability for each observations
- **Step 2 :** Rank these probabilities in decreasing order.
- **Step 3 :** Build deciles with each group having almost 10% of the observations.
- **Step 4 :** Calculate the response rate at each deciles for good (Responders), Bad (Non-responders) and total.

Lift/Gain charts are widely used in campaign targeting problem.

4. Kolmogorov Smirnov Chart :

K-S chart measures performance of classification models. K-S is a measure of the degree of separation between the positive and negative distribution.

5. Area under the ROC curve (AUC-ROC).

- The biggest advantage of using ROC-curve is that it is independent of change in proportion of responders.
- The ROC curve is the plot between sensitivity and (1-specificity). (1- specificity) is know as false positive rate and sensitivity is also known as True positive rate.
- For a model which gives class as output, will be represented as a single point in ROC plot.

6. Log Loss

- AUC – ROC considers only the order of probabilities and it does not take into account the model's capability to predict higher probabilities for samples more likely to be positive.

  So, we calculate log loss.
- Log loss is the negative average of the log of corrected predicted probabilities for each instance.

7. Gini coefficient

Gini is nothing but ratio between area between the ROC curve and the diagonal line and the area of the above triangle.

Gini = 2 (AUC) – 1

Gini above 60% is a good model

8. Concordant – Discordant Ratio

It is primarily used to access the model's predictive power. This metric is useful for any classification predictions.

9. Root mean squared error (RMSE)

It is used in regression problems. It assumes that errors are unbiased and follow a normal distribution.

RMSE metric is given by

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{N} (\text{predicted}_i - \text{Actual}_i)^2}{N}}$$

Where N is total number of observation.

10. Root mean squared Logarithmic Error

Here, we take log of the predictions and actual values.

RMSLE is used when we don't want to penalise huge differences in the predicted and the actual valves when both predicted and true valves are huge numbers.

11. R-squared/Adjusted R-Squared

In RMSE metric, we do not have a benchmark to compare.

The formula for R-squared is :

$$R^2 = 1 - \frac{\text{MSE (model)}}{\text{MSE (base line)}}$$

12. Cross-Validation

It is one of the most important concepts in any type of data modelling. It says : Try to leave a sample on which you do not train the model and test the model on this sample before finalising the model.

5.11 ELBOW METHOD

- In cluster analysis, the **elbow method** is a **heuristic** used in determining the number of clusters in a data set.
- In this method, explained variation as a function of the number of clusters is plotted and picking the **elbow of the curve** as the number of clusters to use.

5.11.1 Intuition Works

- Using the "elbow" as a cutoff point is a common heuristic in mathematical optimisation. It chooses a point where diminishing returns are no longer worth the additional cost.

- The intuition is that increasing the number of clusters will improve the **fit** since there are more parameters to use.
- In practice, this may not be a sharp elbow and as a heuristic method, such an 'elbow' cannot be correctly identified.
- To overcome this the quantity called 'elbow strength' was introduced.

5.11.2 Measures of Variation

- There are various measures of "explained variation", used in the elbow method.
- Commonly, variation is quantified by **variance**, and the ratio used is the ratio of between-group variance to the total variance.
- One can also use the ratio of between-group variance to within-group -variance.

5.11.3 Elbow – Method Calculation

1. Compute clustering algorithm (e.g k-means clustering) for different values of k.
2. For each k, calculate the total within – cluster sum of square (WSS).
3. Plot the curve of WSS according to the number of clusters k.

5.11.4 Working of Elbow – Method

- Elbow method does not always work well, if data is not very much clustered. Dataset need not have a clear elbow.
- We can have a fairly smooth curve, and it is unclear what is the beast valve of k to choose.

5.12 EXTRINSIC AND INTRINSIC METHOD

GQ. Explain and compare extrinsic and intrinsic method.

Extrinsic motivation

Extrinsic motivation refers to the behaviour of individuals to perform tasks and learn new skills because of external **rewards or avoidance of punishment.**

Examples of extrinsic motivation could include :

(i) Reading a book to prepare for a test

(ii) Exercising to lose weight

(iii) Cleaning your home to prepare for visitors coming over.

5.12.1 Intrinsic Motivation

When one is intrinsically motivated, the behaviour is motivated by the internal desire to do something for its own sake for example, one's personal enjoyment of an activity, or one's desire to learn a skill because one is eager to learn.

Examples of intrinsic motivation include :

(i) Reading a book because you enjoy the storytelling.

(ii) Exercising because you want to relieve stress.

(iii) Cleaning your study- room because it helps you feel will-organized

5.12.2 The difference between Intrinsic and Extrinsic Motivation

Intrinsic motivation comes from within, while extrinsic motivation arises from external factors.

When one is intrinsically motivated, one engages in an activity because one enjoys it and gets personal satisfaction from doing it.

Table 5.12.1

Sr. No.	Intrinsic	Extrinsic
1.	Participating in a sport because it is fun and you enjoy it.	Participating in a sport in order to win a reward or get physically fit.
2.	Learning a new language because you like experiencing new things	Learning a new language because your job requires it.
3.	Spending time with some one because you enjoy their company.	Spending time with some one because they can further your social standing .
4.	Cleaning because you enjoy a tidy space.	Cleaning to avoid making your partner angry.
5.	Playing cards because you enjoy the challenge	Playing cards to win money
6.	Exercising because you enjoy physically challenging your body	Exercising because you want to lose weight or fit into an adult.
7.	Volunteering because it makes you feel content and fulfilled	Volunteering in order to meet a school or work requirement.
8.	Going for a run because you find it relaxing or are trying to beat a personal record	Going for a run to increase your chances at winning a competition.
9.	Painting because it makes you feel calm and happy	Painting so you can sell your art to make money.
10.	Taking on more responsibility at work because you enjoy being challenged and feeling accomplishment	Taking on more responsibility at work in order to receive a raise or promotion.

5.12.3 Which is better : Extrinsic or Intrinsic Motivation

- Actually both motivations are effective. Most people agree with the idea that extrinsic rewards should be used less in order to minimise the **over justification effects**.
- It does not mean that extrinsic motivation always presents negative outcomes. In fact, it can be extremely beneficial in some situations, the situation where someone needs to complete a task that they find unpleasant.
- Excessive rewards may create problems, but when used appropriately, extrinsic motivating factors can be a useful tool.
- There are several factors that can help to promote intrinsic motivation.
- These factors are :
  - (i) **Curiosity :** Fostered curiosity pushed people to explore and learn for the sole pleasure of learning and mastering.
  - (ii) **Challenge :** Being challenged helps people to work at optimal levels continuously, while staying consistent in work towards meaningful goals.
  - (iii) **Recognition :** People have an innate desire to be appreciated, so when efforts are recognized and appreciated by others, satisfaction becomes a reward in and of itself.
  - (iv) **Cooperation :** Cooperating with others satisfies the need to belong. It also present the reward of satisfaction, because cooperation involves helping others and working towards a shared goal.
- While intrinsic motivation is often seen as ideal due to its sustainability and the inherent nature of its rewards , both extrinsic and intrinsic motivation are influential in driving behavior .
- In order to understand how these can be best utilized, it is important to understand their key differences and the optimal times to employ each method.

Chapter Ends ...

❑❑❑