Rainy	Yes
Sunny	Yes
Overcast	Yes
Rainy	No
Sunday	No
Sunny	Yes
Rainy	No
Overcast	Yes
Overcast	Yes

Step 1: Frequency Table

Weather	Yes	No
Rainy	2	2
Sunny	3	2
Overcast	5	0
Total	10	4

No of Yes = 10
No of Nos = 4
Total instance = 10+4 = 14

Step 2: Likelihood Table

Weather	Yes	No	
Rainy	2	2	4/14 = 0.29
Sunny	3	2	5/14 : 0.36
Overcast	5	0	5/14 : 0.36
Total	10/14 (0.71)	4/14 (0.29)	

Steps Applying Baye's Theorem:

$$P(Yes|Sunny) = \frac{P(Sunny|Yes) \times P(Yes)}{P(Sunny)}$$

$$= \frac{3}{10} \times \frac{0.71}{0.36} = 0.59$$

$$P(no|Sunny) = \frac{P(Sunny|No) \times P(no)}{P(Sunny)}$$

$$= \frac{2}{4} \times \frac{0.29}{0.36}$$

$$P(Yes|Sunny) > P(no|Sunny)$$

Hence for the instance sunny, the class label is Yes.

Q outlook: rainy

$$P\left(\frac{yes}{rainy}\right) = P\left(\frac{rainy}{yes}\right) \times \frac{P(yes)}{P(rainy)}$$

$$: \frac{2}{10} \times \frac{0.71}{0.29} \quad :$$

$$: 0.489 : 0.49$$

$$P\left(\frac{no}{rainy}\right) = \frac{P\left(\frac{rainy}{no}\right) \times P(no)}{P(rainy)}$$

$$: \frac{2}{4} \times \frac{0.29}{0.29}$$

$$: \frac{1}{2} : 0.5$$

$$P(no/rainy) > P(yes/rainy)$$

Hence for instance rainy the class label is No

Decision Tree

* Consider the dataset. Identify the root using decision tree (induction).

RID	Age	Income	Student	Credit_rating	Buys_computer
1	youth	high	no	fair	no
2	youth	high	no	excellent	no
3	middle_aged	high	no	fair	yes
4	senior	medium	no	fair	yes
5	senior	low	yes	fair	yes
6	senior	low	yes	excellent	no
7	middle_aged	low	yes	excellent	yes
8	youth	medium	no	fair	no
9	youth	low	yes	fair	yes
10	senior	medium	yes	fair	yes
11	youth	medium	yes	excellent	yes
12	middle_aged	medium	no	excellent	yes
13	middle_aged	high	yes	fair	yes
14	senior	medium	no	excellent	no

Total no of Transaction = 14

No of attribute = 4 - excluding class label

Attribute = Age, Income, Student, Credit_rating

Op Variable = Buys_computer

class label = yes / no

Total No of yes = 9
Total No of no = 5

Step 1 Entropy (Buys_Computer)

$$= -\sum_{i=1}^{2} P_i \log_2(P_i)$$

Converting $\log_2$ to $\log_{10}$ $= -\left[\frac{9}{14}\log_2\left(\frac{9}{14}\right) + \frac{5}{14}\log_2\left(\frac{5}{14}\right)\right]$

$P(yes)$ $P(no)$

$\log_2 \rightarrow \frac{\log_{10}(0.64)}{\log_{10} 2}$

$= -\left[0.64 \times \log_2(0.64) + 0.35 \log_2(0.35)\right]$

$= -[-0.94216857]$

0.942

Consider the dataset
Identify the root using Decision Tree

Step 2 Information gain (Age)

Age < youth, middle_aged, senior → Three values

Youth < Yes - 2, No - 3 — 2 yes and 3 no

Entropy (Youth) = $-\sum_{i=1}^{n} P_i \log_2 (P_i)$

yes
yes+no

$= -\left[\frac{2}{5}\log_2\left(\frac{2}{5}\right) + \frac{3}{5}\log_2\left(\frac{3}{5}\right)\right]$

no
yes+no

$= -(-0.97095)$

$= 0.97095$

Middle Aged $\begin{cases} Y_4 - 4 \\ No - 0 \end{cases}$

Entropy (Middle-Aged) $= -\left[\frac{4}{4}\log_2\left(\frac{4}{4}\right) + \frac{0}{4}\log_2\left[\frac{0}{4}\right]\right]$

$= \log_2 1$

$= 0$

Senior $\begin{cases} Y_3 - 3 \\ No - 2 \end{cases}$

Entropy (Senior) $= -\left[\frac{3}{5}\log_2\left(\frac{3}{5}\right) + \frac{2}{5}\log_2\left(\frac{2}{5}\right)\right]$

$= 0.97095$

Information gain (age)

Total entropy (age) = weighted average of entropy

$$\left[\frac{\text{youth total}}{\text{total}} * \text{entropy(youth)}\right] \longleftarrow = \frac{5}{14} \times 0.970 + \frac{4}{14} \times 0 + \frac{5}{14} \times 0.970$$

$$= 0.69285$$

$$= 0.693$$

Info gain (age) = Overall Entropy - Total entropy (age)

$$= 0.94 - 0.693$$

$$= 0.247$$

Info gain (Income)

Income < Low — , Medium , High

Low < Yes — 4 , No — 1

High < Yes — 2 , No — 2

Medium < Yes — 4 , No — 2

Entropy (high) = $-\left[\frac{2}{4}\log_2\left(\frac{2}{4}\right)+\frac{2}{4}\log_2\left(\frac{2}{4}\right)\right]$

$= -(-1)$

$= 1$

Entropy (Medium) = $-\left[\frac{4}{6}\log_2\frac{4}{6}+\frac{2}{6}\log_2\left(\frac{2}{4}\right)\right]$

$= 0.916$

Entropy (low) = $-\left[\frac{3}{4}\log_2\frac{3}{4}+\frac{1}{4}\log_2\left(\frac{1}{4}\right)\right]$

$= 0.811$

Total entropy (income) = weighted average of entropy

$= \frac{4}{14}\times 0.811 + \frac{6}{14}\times 0.916 + \frac{4}{14}\times 1$

$= 0.91$

Infogain (income) = Overal entropy - Total entropy (age)

$= 0.94 - 091$

$= 0.029 \cong 0.03$

0.429
0.86 0.571
0.14

Date
Page

Infogain (Student)

Student < Yes / No

Yes — Yes - 6 / No - 1 : 3.906

No — Yes - 3 / No - 4

Entropy (Yes) = $-\left(\frac{6}{7}\log_2\left(\frac{6}{7}\right) + \frac{1}{7}\log\left(\frac{1}{7}\right)\right)$

= 0.58

Entropy (No) = $-\left[\frac{3}{7}\log_2\left(\frac{3}{7}\right) + \frac{4}{7}\log_2\left(\frac{4}{7}\right)\right]$

= 0.985

Total entropy (Student) = Weighted average of entropy

$= \frac{7}{14} \times 0.59 + \frac{7}{14} \times 0.985$

= 0.7875

Infogain (student) = Overall Entropy - Total entropy (Stud)

= 0.94 - 0.7875

= 0.1525

0.75
0.25

Info gain (credit-rating)

fair <Yes — 6
      <No — 2

excellent <Yes — 3
          <No — 3

Entropy (fair) = $-\left(\frac{6}{8}\log_2\frac{6}{8} + \frac{2}{8}\log_2\frac{2}{8}\right)$

= 0.811

Entropy (excellent) = $-\left(\frac{3}{6}\log_2\frac{3}{6} + \frac{3}{6}\log_2\frac{3}{6}\right)$

= 1

Total entropy = Weighted average of entropy

$= \frac{8}{14} \times 0.811 + \frac{6}{14} \times 1$

= 0.892

Info gain (credit rating) = Overall - total (credit)

= 0.94 - 0.892

= 0.048

The info gain with the maximum value is the root

∴ ~~root = age~~ IG(Age) = 0.247
IG(Income) = 0.029
IG(Student) = 0.152
IG(credit) = 0.048

IG(Age) is the highest ∴ root = age.

HW
a Consider the dataset

Income	Student	Credit rating	Class
medium	no	fair	yes
low	yes	fair	yes
low	yes	excellent	no
medium	yes	fair	yes
medium	no	excellent	no

Identify the root using the decision tree

Total transactions:
No of attributes → (Income, student, credit_rating)
o/p variable: class
class labels: yes, no
Total yes: 3 Total No: 2

Entropy of data : $\sum_{i=1}^{n} P_i \log_2(P_i)$

$= -\left[\frac{3}{5}\log_2(3/5) + 2/5 \log_2(2/5)\right] = 0.971$

Info gain (Income) :

Income → medium → Yes = 3, No = 1
Income → low → Yes = 1, No = 1

Entropy (medium) : $-\left[2/3 \log(2/3) + 1/3 \log_2(1/3)\right]$

$= 0.918$

Entropy (low) : $-\left[1/2 \log_2(1/2) + 1/2 \log_2(1/2)\right]$

$= 1$

Total entropy (Income) = Total entropy − entropy of subset

$= 0.971 - 0.951 = 0.02$

Info gain (student)

Student → Yes → Yes = 2, No = 1
Student → No → Yes = 1, No = 1

entropy (yes) $= -\left[2/3 \log(2/3) + 1/3 \log_2(1/3)\right]$

$= 0.918$

Entropy (No) $= -\left[1/2 \log_2(1/2) + 1/2 \log_2(1/2)\right]$

$= 1$

Total entropy (Student) : $3/5 \times 0.98 + 2/5 \times 1 = 0.951$

Info-gain (Student) : $0.971 - 0.951$
$= 0.02$

Info gain of credit_rating

credit → fair → yes - 3, no - 0
credit → excellent → yes - 0, no - 2

Entropy (fair) : $-\left[\frac{3}{3}\log_2(2/3) + \frac{0}{3}\log_2(0/3)\right] = 0$

entropy (excellent) : $-\left[\frac{0}{2}\log_2(0/2) + \frac{2}{2}\log_2(2/2)\right]$
$= 0$

Total entropy : $3/5 \times 0 + 2/5 \times 0 = 0$

Info gain (credit_rating) : $0.971 - 0 = 0.971$

IG (Income) : 0.02
IG (Student) : 0.02
IG (credit_rating) : 0.971

IG with maximum : IG (credit_rating)
∴ Root : credit_rating

KNN Algorithm (K-Nearest Neighbour Algorithm)

⇒ It is a classification algorithm which comes under Supervised Learning

→ It is a lazy learner algorithm. It does not learn the training dataset.

→ It works based on similarity of data. It assumes similarity between the new data and available cases and put

KNN works best when the dataset is small. When dataset is big → Decision Tree.

KNN does not learn a discriminative function from the training set. ∴ it is a lazy learner.

KNN can identify the category to which an input belongs to.

The most preferred value for K is 5.

a. Consider the given dataset:

Maths	CS	Result
4	3	F
6	7	P
7	8	P
5	5	F
8	8	P

Find the result for the query maths = 6 cs = 8 and value of k = 3. using KNN algorithm.

distance – Euclidean Distance

	Maths	CS	Result	d	Neighbour Rank
d_1 –	4	3	F	5.38	5
d_2 –	6	7	P	1	1 ✓
d_3 –	7	8	P	1	2 ✓
d_4 –	5	5	F	3.16	4
d_5 –	8	8	P	2	3 ✓

↳ 2 Smallest values are ranked first

$$d_1 = \sqrt{(x_1 - x_2)^2 + (x_2 - y_2)^2}$$

$$= \sqrt{(6-4)^2 + (8-3)^2}$$

$$= \sqrt{4 + 5^2}$$

$$= 5.38$$

$$d_2 = \sqrt{(6-6)^2 + (8-7)^2}$$

$$= \sqrt{1^2}$$

$$= 1$$

$d_3 = \sqrt{(6-7)^2 + (8-8)^2}$

1

$d_4 = \sqrt{(6-5)^2 + (8-5)^2}$

$= \sqrt{1^2 + 3^2}$

$= \sqrt{10}$

$= 3.16$

$d_5 = \sqrt{(6-8)^2 + (8-8)^2}$

$= \sqrt{2^2 + 0^2}$

$= 2$

$\Rightarrow$ k=3 means we have to select the first 3 ranks
i.e 1, 2, 3
Analyze their results - 1-P 2-P 3-P
∴ 6 and 8 would yield P as output

The result of the instance Math = 6 and CS = 8 will be P ∴ the first 3 neighbour ranks will give the result as P. (majority)

a Consider the dataset

Brightness	Saturation	Class
40	20	Red
50	50	Blue
60	90	Blue
10	25	Red
70	70	Blue
60	10	Red
25	80	Blue

Find the class of the given data brightness: 20 saturation: 35 k=5

	Brightness	Saturation	Class	d	Neighbour rank
d_1	40	20	Red	25	2
d_2	50	50	Blue	33.54	3
d_3	60	90	Blue	68.007	7
d_4	10	25	Red	14.14	1
d_5	70	70	Blue	61.03	6
d_6	60	10	Red	47.16	5
d_7	25	80	Blue	45.27	4

$$d_1 = \sqrt{(60-40)^2 + (35-20)^2}$$
$$= \sqrt{20^2 + (-15)^2}$$
$$= \sqrt{400 + 225} = \sqrt{625} = 25$$

$$d_1 = \sqrt{(10-50)^2 + (20-90)^2}$$

$$= \sqrt{900 + 2025}$$

$$= 33.54$$

$$d_2 = \sqrt{(20-60)^2 + (35-90)^2}$$

$$= \sqrt{1600 + 3025}$$

$$= 68.007$$

$$d_3 = 14.14$$

$$d_5 = 61.03$$

$$d_6 = 47.16$$

$$d_7 = 45.27$$

k = 3

1 - Red
2 - Red
3 - Blue
4 - Blue
5 - Red

So the car is Red for the query brightness: 20 saturation: 35

Note: All Classification comes under supervised learning.
All Clustering comes under unsupervised learning.

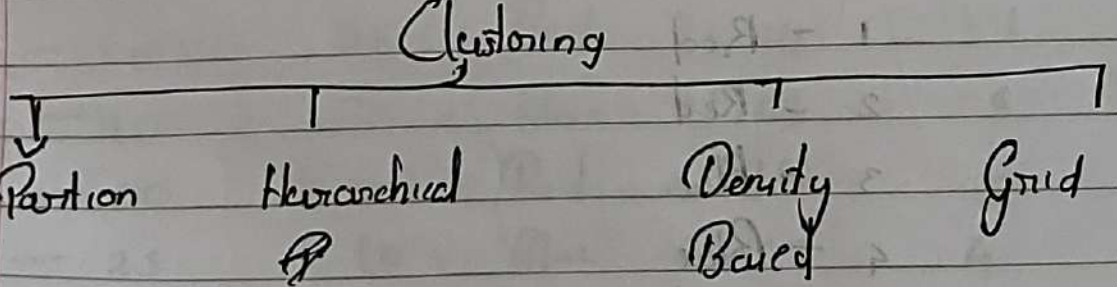
Clustering

way of
→ Grouping the data points into different clusters, consisting of similar data points.

→ Unlabelled datasets are grouped here
→ Unsupervised

Applications

> Market Segmentation
> Recommendation System
> Movie Recommendation System.



The only method where we can specify the no of partitions is the partition method.

Partition Clustering

→ It is used to make partitions on the data in order to form clusters

→ If n partitions are done on p objects of the database then each partition is represented by a cluster and n≤p

Two condition to be satisfied
↳ One objective should only belong to 1 group
↳ There should be no group without its purpose

Method → Iterative Relocation
↳ We can move data from one cluster to another

→ Centroid-based method.
↳ K-Means Clustering Algorithm.

Hierarchical Clustering :-

No of Clusters need not be specified before hand.

Data → Cluster → Tree-like structure (dendogram)

Approach
- Agglomerative Approach - (Bottom-up approach)
- Divisive Approach - (Top to bottom)

Divisive
- Top Down approach

Density Based Clustering :-

The method mainly focuses on density

Grid Based Clustering :-

→ Divided data in the technique into Finite no of cells (grids) to for clustering

→ Faster processing time.

K-Mean Algorithm

a) Consider the data points A1(2,10) A2(2,5) A3(8,4) B1(5,8) B2(7,5) B3(6,4) C1(1,2) C4(4,9). The initial clusters, (centroids) are C1(2,10) C2(5,8) C3(1,2). Find the clusters using K mean algorithm

Euclidean Distance

Data Points			C1 (2, 10)	C2 (5, 8)	C3 (1, 2)	Cluster
A1	2	10	$\sqrt{(2-2)^2+(10-10)^2}$ = 0	3.61	8.06	C1
A2	2	5	$\sqrt{(2-2)^2+(10-5)^2}$ = 5	4.24	3.16	C3
A3	8	4	8.49	5	7.280	C2
B1	5	8	3.61	0	7.211	C2
B2	7	5	7.07	2.236	6.708	C2
B3	6	4	7.21	4.123	5.385	C2
C1	1	2	8.06	7.211	0	C3
C2	4	9	2.24	1.414	7.615	C2

New Centroids

C1 : (2, 10)

C2 : $\left(\frac{8+5+7+6+4}{5}, \frac{4+8+5+4+9}{5}\right)$

: (6, 6)

C3 : $\left(\frac{2+1}{2}, \frac{5+2}{2}\right)$

: $\frac{3}{2}, \frac{7}{2}$

: (1.5, 3.5)

			C1 (2, 10)	C2 (6, 6)	C3 (1.5, 3.5)	cluster	new cluster
A1	2	10	0	5.66	6.52	C1	C1
A2	2	5	5	4.12	1.58	C3	C3
A3	8	4	8.49	2.83	6.52	C2	C2
B1	5	8	3.61	2.24	5.7	C2	C2
B2	7	5	7.07	1.41	5.7	C2	C2
B3	6	4	7.21	2	4.53	C2	C2
C1	1	2	8.06	6.40	1.58	C3	C3
C2	4	9	2.24	3.61	6.04	C2	C1

C2 C1 -> not matching, so need iteration

New Centroids:-

A1 $c_1 = \left(\frac{2+4}{2}, \frac{10+9}{2}\right)$

$: (3, 9.5)$

$c_2 = \left(\frac{8+5+7+6}{4}, \frac{4+8+5+4}{4}\right)$

$: (6.5, 5.25)$

$c_3 = \left(\frac{2+1}{2}, \frac{5+2}{2}\right)$

$: (1.5, 3.5)$

			c1 3, 9.5	c2 6.8 2.5	c3 4.5 8.5	cluster	new cluster
A_1	2	10	1.2	6.54	6.52	c1	c1
A_2	2	5	4.61	4.5	1.58	c3	c3
A_3	8	4	7.43	1.95	6.52	c2	c2
B_1	5	8	2.50	3.13	5.70	c2	c1
B_2	7	5	6.02	0.56	5.7	c2	c2
B_3	6	4	6.26	1.85	4.53	c2	c2
C_1	1	2	7.76	6.89	1.58	c3	c3
C_2	4	9	7.12	4.51	6.04	c1	c2

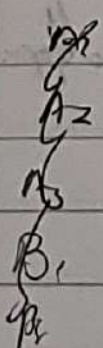
New centroids

$c_1 = \left(\frac{2+5}{2}, \frac{10+8}{2}\right) = (3.5, 9)$

$c_2 = (8+7+6+4)/4, (4+5+4+9)/4$

$= (6.25, 5.5) \quad (7, 4.33)$

$\therefore c_3 = (1.5, 3.5)$



			c1	c2	c3	cluster	New cluster
			3.61 9.8	~~8.6 5.53~~	1.5 3.5		
A_1	2	10	1.94	7.66	6.52	c1	c1
A_2	2	5	4.33	5.06	1.58	c3	c3
A_3	8	4	6.62	1.05	6.52	c2	c2
B_1	5	8	1.67	4.18	5.70	c1	c1
B_2	7	5	5.21	0.67	5.7	c2	c2
B_3	6	4	5.52	1.05	4.53	c2	c2
C_1	1	2	7.49	6.44	1.58	c3	c3
C_2	4	9	0.33	0.55	6.04	c1	c1

New centroids

Cluster (A_1, B_1), (A_3, B_2, B_3, C_2)
(A_2, C_1)

1(w) Consider the data points $A_1(2,2)$ $A_2(3,3)$ $A_3(6,8)$, $A_4(8,8)$, $A_5(7,5)$, $A_6(9,7)$

Initial centroids - $c_1(2,0)$ $c_2(8,8)$
Find out the cluster using K Means Algorithm

K-Medoid Clustering:-

→ It is also an unsupervised learning algorithm

a Consider the dataset $x_1(2,6)$, $x_2(3,4)$, $x_3(3,8)$ $x_4(4,2)$ $x_5(6,2)$ $x_6(6,4)$ $x_7(7,3)$ $x_8(7,4)$ $x_9(8,5)$ $x_{10}(7,6)$. The medoids are $c_1=(3,4)$ $c_2(7,4)$. Find out the cluster using K-medoid algorithm

manhattan distance

			c_1 (3,4)	c_2 (7,4)	Cluster
x_1	2	6	3	7	c_1
x_2	3	4	0	4	c_1
x_3	3	8	4	8	c_1
x_4	4	2	3	5	c_1
x_5	6	2	5	3	c_2
x_6	6	4	3	1	c_2
x_7	7	3	5	1	c_2
x_8	7	4	4	0	c_2
x_9	8	5	6	2	c_2
x_{10}	7	6	6	2	c_2

New medoids are

{(2,6), (3,4), (3,8), (4,2)}

{(6,2), (6,4), (7,3) (7,4), (8,5), (7,6)}

Cost = Cost(c_1) + Cost(c_2)

min of value

$[3+0+4+3] + [3+1+1+0+2+2]$

$= 19$

HW

	Datapoints	c1 (2,3)	c2 (8,8)	clusters
A_1	2, 3	0	7.81	c1
A_2	3, 3	1	7.07	c1
A_3	6, 8	6.40	2	c2
A_4	8, 8	7.81	0	c2
A_5	7, 5	5.38	3.16	c2
A_6	9, 7	8.06	1.41	c2

Initial centroid (2,3) (8,8)

New centroids

c1 = (2.5, 5)

c2 = (2.5, 2)

	Datapoints	c1 (7.5, 3)	c2 (7.5, 2)	old	new
A_1	2, 3	0.5	6.80	c1	c1
A_2	3, 3	0.5	6.02	c1	c1
A_3	6, 8	6.10	1.128	c2	c2
A_4	8, 8	7.43	1.12	c2	c2
A_5	7, 5	4.92	2.06	c2	c2
A_6	9, 7	2.63	1.5	c2	c2

Clusters = (A_1, A_2) , (A_3, A_4, A_5, A_6)

Remove the first medoid

			C1 (7,4)	C2 (7,3)	clusters (having random centroid)
x1	2	6	7	8	C1
x2	3	4	4	5	C1
x3	3	8	8	9	C1
x4	4	2	5	4	C2
x5	6	2	3	2	C2
x6	6	4	1	2	C1
x7	7	3	1	0	C2
x8	7	4	0	1	C1
x9	8	5	2	3	C1
x10	7	6	2	3	C1

Cost = Cost(C1) + Cost(C2)

= [7+4+8+1+0+2+2] + [4+2+0]

30

Previous Cost = 19

Current Cost = 30

Current Cost > Previous Cost

∴ The final medoids are (3,4), (7,4)

K-means

Exam → Explain steps in K-means algo - Part A
Problem - Part B.

Write the steps we have done in class rather than the notes

K-medoid

Improved Version of K-means algorithm.

K-medoid clustering algorithm

Exam → Difference and similarity btw K-means and K-medoid

Density Based Clustering

→ Unsupervised learning
→ Clustering is based on density of data.
→ The commonly used density based algo - DBSCAN

→ Noise of data is possible in density based clustering.
DBSCAN - Density Based Spatial Clustering of Applications with Noise

* - Parameter of DBScan
  - Epsilon (ε)
  - Minimum points

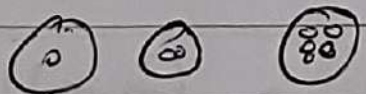
Epsilon:-
Epsilon is defined as the radius of each data point around which the density is considered.

minpoints:
Minimum no of data within the cluster.

Core point If cluster has
- More than minimum point within epsilon


core point

Border points or boundary points
- Fewer than minimum points


boundary point

Noise point
- A point which is not a core point or boundary point

Steps in DBSCAN

* Diff btw K-means and DBSCAN's

Clique Algo

The algo uses density and grid based technique. i.e by taking density threshold and no of grids as i/p parameters.

To handle datasets with large no of dimensions rows and columns

Use the Apriori Property effectively

Find divide data space into grids by dividing each dimension into equal intervals called units

---

5th Module

~~*~~

Essay
- Clustering Definition?
- What is Clustering
- Clustering Algorithm

Q Consider the dataset and find out the cluster using agglomerative method and draw the dendrogram.

Sample	X	Y
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

I Compute the distance matrix — Euclidean distance

$d(p_1, p_2) = \sqrt{0.25 + 0.25} = 0.707$

$d(p_1, p_3) = \sqrt{25 + 25} = 5.657$

$d(p_1, p_4) = \sqrt{4 + 9} = 3.606$

$d(p_1, p_5) = \sqrt{9 + 9} = 4.242$

$d(p_1, p_6) = \sqrt{4 + 6.25} = 3.201$

$d(p_2, p_3) = 4.94$

$d(p_2, p_4) = 2.92$

$d(p_3, p_5) = 3.53$

$d(p_2, p_6) = 2.5$

$d(p_3, p_4) = 2.24$

$d(p_3, p_5) = 1.41$

$d(p_3, p_6) = 2.5$

$(p_4, p_5) = 1.0$

$(p_4, p_6) = 0.5$

$(p_5, p_2) = 1.12$

	p_1	p_2	p_3	p_4	p_5	p_6
p_1	0	0.				
p_2	0.707	0				
p_3	5.656	4.94	0			
p_4	3.606	4.92	2.24	0		
p_5	4.242	3.53	1.41	1.0	0	
p_6	3.201	2.5	2.5	(0.5)	1.12	0

Minimum value = 0.5

Step II Merging the two closed members in the distance matrix

	p_1	p_2	p_3	p_4, p_6	p_5
p_1	0				
p_2	0.707	0			
p_3	5.656	4.94	0		
p_4, p_6	3.201	(2.5)	2.24	0	
p_5	4.242	3.53	1.41	1.12	0

$$d_{min}[(p_4, p_6), p_1] = d_{min}[(p_4, p_1), (p_6, p_1)]$$

224000

classmate
Date
Page

$= \max(3.606, 3.201)$

$= 3.201$

$d(p_4, p_5), p_2 = d[(p_4, p_2), (p_6, p_2)]$

$= 2.92, 2.5$

$= 2.5$

$d(p_3, p_6), p_3 = d(p_4, p_5), (d p_6, p_5)$

$= \max(2.24, 2.5)$

$= 2.24$

$d(p_4, p_6), p_5) = d(p_4, p_5), d(p_6, p_5)$

$1.41, 1.12$

$= 1$

Minimum value = 0.707

Ensemble Classifier :-

Combine multiple models to improve the overall performance

Types
- Bagging → Bootstrap Aggregating → Voting
- Boosting

Agglomerative and draw dendrogram

$d(p_1, p_2) = 1$
$d(p_1, p_3) = 4$
$d(p_1, p_4) = 5$
$d(p_2, p_3) = 2$
$d(p_2, p_4) = 6$
$d(p_3, p_4) = 3$

	p_1	p_2	p_3	p_4
p_1	0			
p_2	(1)	0		
p_3	4	2	0	
p_4	5	6	3	0

Smallest 1 belong to p_2, p_1

	p_1, p_2	p_3	p_4
p_1, p_2	0		
p_3	2	0	
p_4	5	3	0

$d((p_3, (p_1, p_2)) = d[(p_3, p_1)], d(p_3, p_2)$

$= 4, 2$

$\min(4, 2) = 2$

$d[p_4, (p_1, p_2)] = d[(p_4, p_1)], d(p_4, p_2)]$

$= 5, 6$

$\min = 5$

minimum in matrix = 2

p_1, p_2, p_3 p_4

~~$p_1 p_2$~~

$p_4 p_2 p_3$ 0

p_4 (3) 0

$d(p_4, (p_1 p_2, p_3)) = d[(p_4, p_1), (p_4, p_2), (p_4, p_3)]$

$= 5, 6, 3$

$= 3$

min p_4

The Clusters are: $\{(p_1, p_2), p_3, p_4\}$