PAGE No.

DATE : / /

NAIVE BAYESIAN PROBLEM

Consider the training data in the following table where play is a class attribute. In the table, the humidity the attributes has values "L" (for low) or "H" (for high), Sunny has values "Y" (for yes) or "N" (for No) Wind has values "S" (for Strong) or "W" (for weak) and play has "Yes" or "No"

Humidity	Sunny	Wind	play
L	N	S	No
H	N	W	Yes
H	Y	S	Yes
H	N	W	Yes
L	Y	S	No

what is the class label for the following day (Humidity = L, Sunny = N, wind = S)

GIVEN, No. of datasets = 5

Evidence (Humidity = L, Sunny = N, wind = S)

P(Play = "Yes") = 3/5 P(Play = "Yes", Humidity = "L") = 0

P(play = "No") = 2/5 P(play = "No", Humidity = "L") = 2/2 =

P(play = "Yes", Sunny = "N") = 2/3 = 0.66

P(play = "No", Sunny = "N") = 1/2 = 0.5

P(play = "Yes", wind = "S") = 1/3 = 0.33

P(play = "No", wind = "S") = 2/2 = 1

GIVEN Test set:-

P(Yes / Hum=L, Sunny=N, Wind=S) = 0 X 0.66 X 0.33
= 0.

P(No / Hum=L, Sunny=N, Wind=S) = 1 X 0.5 X 1
= 0.5

Since P(No) > P(Yes) play = "No"

DECISION TREE CLASSIFIER.

tree like predictive model used for classification and regression.

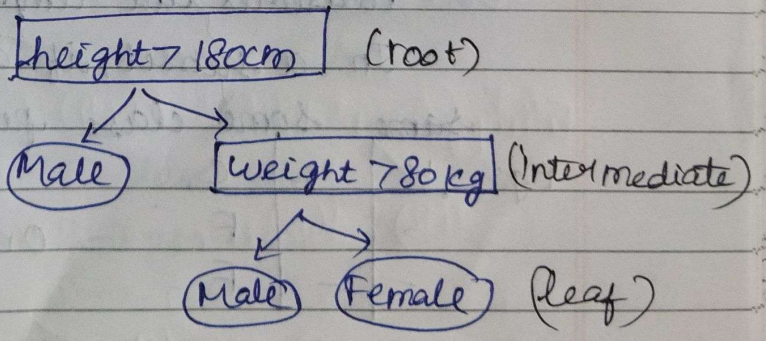
Supervised learning model.

ALGORITHMS USED:

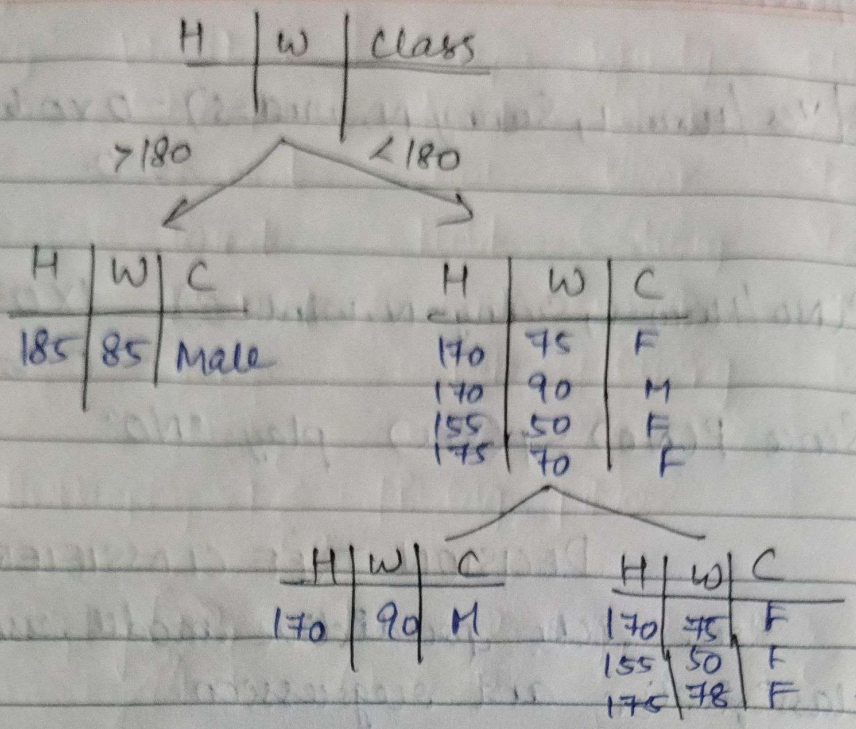
CART, ID3
↓ Regression ↓ Classification

Height (cm)	Weight (kg)	Class
170	75	Female
170	90	Male
185	85	Male
155	50	Female
175	70	Female

STRUCTURE:



In this DT classifier class label is Male / Female based on height and weight.



At each node of dt construction choose a feature as a splitting attribute.

Random attribute ⇒ Not a good idea
less accuracy.

Therefore we need to find best splitting attribute.

FEATURE SELECTION:

Information gain
Gini index

INFORMATION GAIN:

Identify entrophy (finding impurities)

ENTROPHY (S):

Measures the impurity in the dataset

In a subset, if all the samples are in same class pure node.

Eg:

H	W	C
-	-	F
-	-	F

Or

H	W	C
-	-	M
-	-	M

All the samples in the above are in the same class.

PAGE No.
DATE : / /

Eg: H W C

H	W	C
170	75	F
170	90	M
155	50	F
175	78	F

No of samples = 4
2 classes = F, M

$$E(S) = \sum_{i=1}^{c} -p_i \log_2 p_i$$

$$= -\frac{1}{4}\log_2\left(\frac{1}{4}\right) - \frac{3}{4}\log\left(\frac{3}{4}\right)$$

$$= \frac{-1}{4} \times (-2) \quad \frac{-3}{4} \times (-0.415)$$

$$= 0.81125$$

If the samples are in equal class this subset is impure.

Entrophy of pure node = 0

Impurity is high ⇒ entrophy = 1.

Value of entrophy range 0 - 1.

Dataset contains 2 class then the entrophy will be 0 or 1.

Dataset contains "n" classes the entrophy is $0 - \log_2 n$.

$$\text{Entrophy} = \sum_{i=1}^{c} -p_i \log_2 p_i$$

C - class label

P_i → probability of ith class

S → Subset.

PAGE No.
DATE : / /

Eg:

Day	Weather	Temp	Windy	play
1.	Rainy	Mild	Weak	No
2.	Normal	Hot	Weak	Yes
3.	Wind	Mild	Strong	Yes
4.	Normal	Cool	weak	No
5	Rain	hot	Strong	No

$$\text{Entropy} = \sum_{i=0}^{c} -p_i \log_2 p_i$$

$$\text{Entropy}[+2, -3] = \frac{-2}{5}\log_2\frac{2}{5} + \frac{3}{5}\log_2\frac{3}{5}$$

2 Yes
3 NO

$$\log_2\frac{2}{5} = \frac{\log 2/5}{\log 2}$$

$$= \frac{-2}{5}(-1.32) + \frac{3}{5}(0.736)$$

$$= \frac{2.64 + 2.2}{5} = \frac{9.84}{5} = 0.96$$

Information Gain Gain = Entropy(S) $- \sum \frac{|S_v|}{|S|}$ Entropy(S_v)

Gain(weather) = {Rainy, Normal, wind}

Entropy (Rain) $\Rightarrow$ Entropy(0, 2) $= \frac{0}{2}\log_2\frac{0}{2} - \frac{2}{2}\log_2$

0Y 2(N)

$= 0$

Entropy (Normal) $\Rightarrow$ (1, 1) $= \frac{-1}{2}\log_2\frac{1}{2} - \frac{1}{2}\log_2\frac{1}{2}$

1(Y) 1(N)

$= 1$

Entropy(wind) $\Rightarrow$ (1, 0) $= \frac{-1}{1}\log_2\frac{1}{1} - \frac{0}{1}\log_2\frac{0}{1}$

1(Y) 0(N)

$= 0$

DATE : / /

Gain (weather) = Entropy (0.968) $-\left(\frac{3}{5}\times 0\right)-\left(\frac{2}{5}\times 1\right)-\left(\frac{1}{5}\times 0\right)$

$= 0.568$

Gain (Temp) = {Mild, Hot, Cool}

Entropy (Mild) = (+1, -1) $= \frac{-1}{2}\log_2\frac{1}{2} - \frac{1}{2}\log_2\frac{1}{2}$

$= 1$

Entropy (Hot) = (+1, -1) $= \frac{-1}{2}\log_2\frac{1}{2} - \frac{1}{2}\log_2\frac{1}{2}$

$= 1$

Entropy (Cool) = (0, -1) $= \frac{-0}{1}\log_2\frac{0}{1} - \frac{1}{1}\log_2\frac{1}{1}$

$= 0.$

Gain (Tem) $= 0.97 - \left(\frac{2}{5}\times 1\right) - \left(\frac{2}{5}\times 1\right) - \left(\frac{1}{5}\times 0\right) = 0.17$

Gain (windy) = {weak, strong}

Entropy (weak) = (1, -2) $= \frac{-1}{3}\log_2\frac{1}{3} - \frac{2}{3}\log_2\frac{2}{3}$

$= 0.5283 + 0.3899$

$= 0.9182$

Entropy (Strong) = (1, -1) $= \frac{-1}{2}\log_2\frac{1}{2} - \frac{1}{2}\log_2\frac{1}{2}$

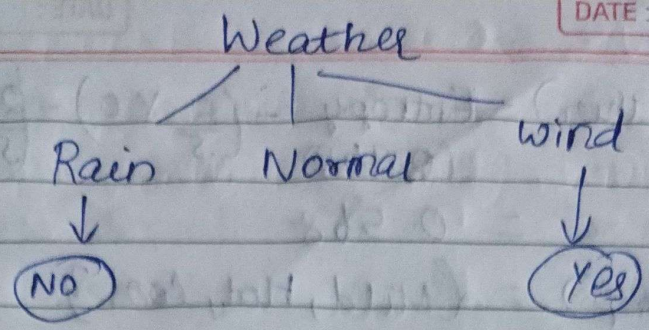
$= 1$

Gain (windy) $= 0.97 - \left(\frac{3}{5}\times 0.9182\right) - \left(\frac{2}{5}\times 1\right)$

$= 0.97 - 0.55 - 0.4$

$= 0.02.$

DATE : / /



Day	Temp	wind	play
2	Hot	weak	Yes
4	Cool	weak	No

Gain(Temp) = {HOT, COOL}

Entrophy (Hot) = (1, 0) = $\frac{-1}{1}\log_2\frac{1}{1} - \frac{0}{1}\log_2\frac{0}{1}$

1/2 = 0

Entropy(Cool) = (0, -1) = $\frac{-0}{1}\log_2\frac{0}{1} - \frac{-1}{1}\log_2\frac{1}{1}$

= 0

Gain(Temp) = $0.97 - \frac{1}{2}(0) - \frac{1}{2}(0)$

RULE BASED ALGORITHM

Rule based classifier are just another type of classifier, which makes the class decision depending by various "IF ELSE" rules.

IMPORTANT KEYWORDS:

If, And, then

SET OF IF THEN RULES FOR CLASSIFICATION:

If condition then conclusion.

"If" part of the rule is called rule Antecedent precondition.

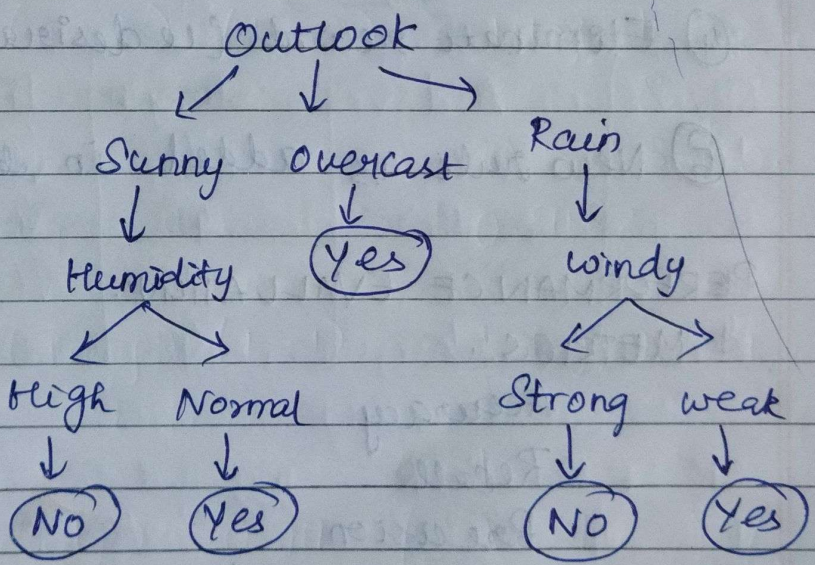
DATE: / /

"Then" part of the rule is called rule consequent

Consequent part consist of class prediction.

eg: If Outlook = "Sunny" AND Humidity = "High" ← Pre-Condition
then play = "NO" → Rule Consequent

To extract the rule from decision tree one rule is created for each path from the root to the leaf node.
Each splitting condition is logically AND
The leaf node holds the class prediction, forming the rule consequent.



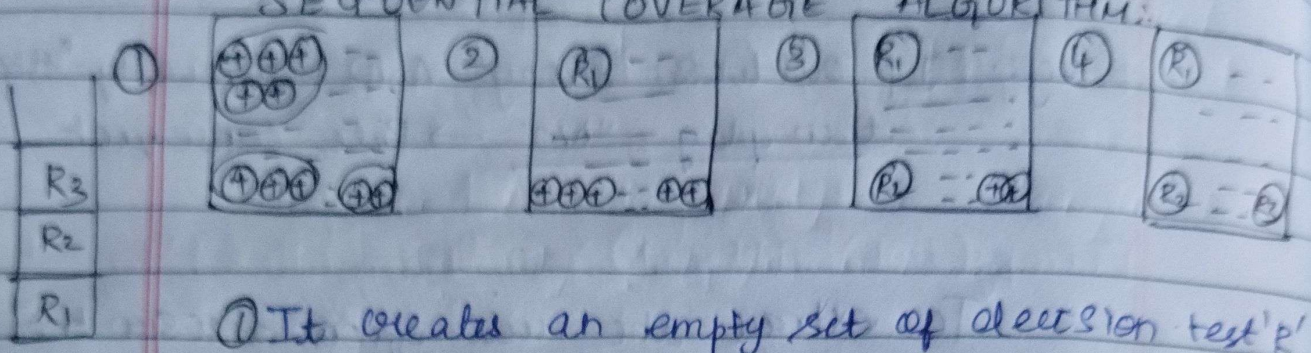
If outlook = "Sunny" AND Humidity = "High" then PLAY = "NO"

If Outlook = "Sunny" AND Humidity = "Normal" then Play = "Yes"

If Outlook = 'overcast', then play = "Yes"
If Outlook = "Rain" AND Windy = "Strong"

RULE BASED

SEQUENTIAL COVERAGE ALGORITHM:



① It creates an empty set of decision test 'R'

② A function called "Learn of rule" function is used. It extract the best rule for class "y"
If all the training records ∈ class y ⇒ +
If all the training records ∉ class y ⇒ -

③ Get desirable value (only true)

④ Eleminate records (ie desirable one)

⑤ New rule is added in the bottom of the 'R'

PERFORMANCE EVALUATION:

METRICS:

Accuracy
Recall
Precision
F1 Score

Eg: Binary Images.

[100] Cat + dog

For train 30% test

Image

PAGE No.
DATE : / /

TEST IMAGE:

test → trained Model → Dog

F1 → 1 → Good
F1 → 0 → Worst.

Dog → y-test → Correct label

Accuracy = $\frac{\text{NO. of Correct prediction}}{\text{Total no. of prediction}}$

$= \frac{TP + TN}{TP + TN + FP + FN}$

Recall = All the relavant = $\frac{TP}{TP + FN}$

Precision ⇒ only the relavant datapoint = $\frac{TP}{TP + FP}$

F1 Score ⇒ $\frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$

CONFUSION MATRIX:

N=165	Predicted NO	Predicted Yes	
Actual No	TN = 50	FP = 10	60
Actual Yes	FN = 5	TP = 100	105
	55	110	

Accuracy = $\frac{150}{165} = 0.90$

Recall = $\frac{100}{105} = 0.9$

Precision = $\frac{100}{110} = 0.9$

F1 Score = $\frac{2 \times 0.85}{1.85}$

$= \frac{1.7}{1.85}$

$= 0.91$

PAGE No.

DATE : / /

K - NEAREST NEIGHBOR

Calculate the KNN classification for the dataset & predict for Paper 5.

Sample pages	Acid durability	Strength	Quality
1	7	7	Bad
2	7	4	Bad
3	3	4	Good
4	1	4	Good

5th Data :- Acid durability = 3 Strength = 7

Quality = ? given k = 3

Identify k :

STEP-1 :- k = 3

Calculate the distance b/w the query instance and all the training samples

∴ Given: instance (3, 7) & Calculate using Euclidean distance

$$d(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

SP	A.d	Strength	Distance
1	7	7	$\sqrt{(3-7)^2+(7-7)^2} = 4$
2	7	4	$\sqrt{(3-7)^2+(7-4)^2} = 5$
3	3	4	$\sqrt{(3-3)^2+(7-4)^2} = 3$
4	1	4	$\sqrt{(3-1)^2+(7-4)^2} = \sqrt{13} = 3$

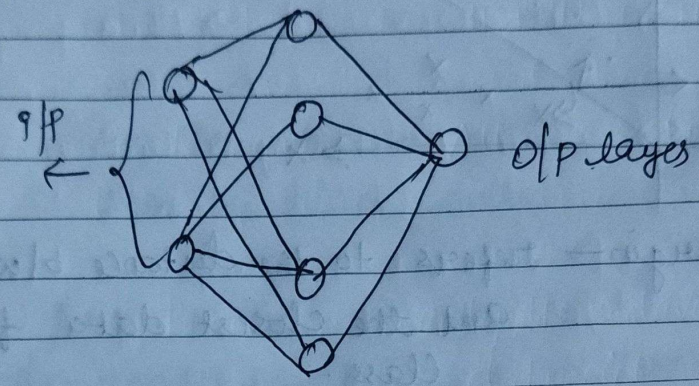
SP	ED	Rank	Included in 3 neighbor
1	4	3	Yes
2	5	4	No
3	3	1	Yes
4	3.6	2	Yes

PAGE No.
DATE : / /

Dataset : 1, 3, 4

SP	A.D	Strength	Quality
1	7	7	Bad
3	3	4	Good
4	1	4	Good
5	3	7	Good.

CLASSIFICATION BY BACK PROPAGATION :-



FORWARD PASS :-

① Σ
② efivation function.

BACKWARD PASS :-

① Error Comparison
② weight updation
③ Bias updation.

SVM CLASSIFICATION :-

SVC :-

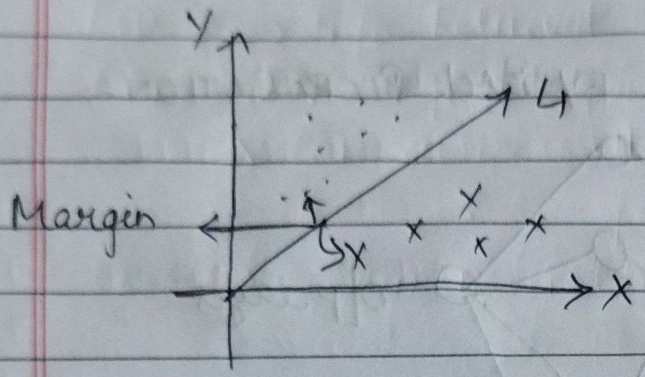
classification method for both linear & non-linear data.

An SVM classifies the data by

finding the best hyper plane that separates all data points of an class from those of the other class

It can handle high dimensional data.

Support Vector → datapoint in a dataset that are closest to the hyperplane



Margin — refers to the distance b/w hyperplane and the closest data from each class.

HYPER LINE:

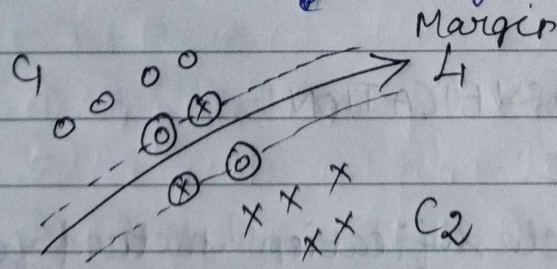
Decision boundary that separates a data points into different classes.

MAIN GOAL OF SVM:

to find the hyperline that maximise the margin b/w 2 classes.

MAXIMUM MARGIN CLASSIFIER:

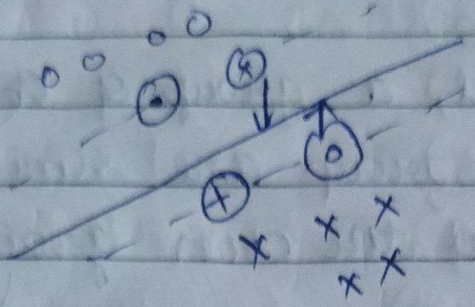
uses hard margin



MMC will not allow misclassification it won't allow for 2 weather it may belong to C2

DATE : / /

SVM allows misclassification datapoints
uses soft margin.



MARGIN: Cross Validation.
↳ used to determine the best soft margin.

Maximum Margin
Min-Mis classification.

LAZY	EAGER
→ instance based learning it is, → less time in training more time in predicting	→ Given a set of training peoples, construct a classification model & receives a new data to classify
Eg:- kNN, LazyNB	Eg:. DecisionTree, SVM, ANN

PAGE No.
DATE : / /

MODULE-IV
CLUSTERING

CLUSTERING:
In general, a group of objects such that objects in the group are similar and dissimilar with objects in the other groups.

APP: → Info. Retrieval
→ Earthquake prediction
→ Can also used → Data preprocessing for data Summarisation.
→ Dimensionality Reduction.

TYPES: * partitional clustering
* Hierarchical "

PARTITIONAL:
a division data objects subset (cluster) such that each data object is in exactly one subset
Algorithms: KNN, CLARA, PAM

HIERARCHICAL:
a set of nested cluster organised as a hierarchical tree.
STEPS:
① Compute the proximity Matrix
② Let each datapoint be a cluster
③ Merge the two closest cluster
④ update the proximity Matrix until only one cluster is defined.

PAGE No.
DATE : / /

TYPES:-

1. Agglomerative cluster
2. Divisive Cluster.

KNN CLUSTER ANALYSIS:

Consider the dataset (D) given below that records the programming skill rating of students in the scale of 0-10. Use K-Means algorithm and euclidean distance measure to cluster the students into two cluster.

(K) = 2 Initial:- $D_1(1,3)$, $D_2(3,2)$

Student Id	Java	Python
D_1	1	3
D_2	3	2
D_3	6	5
D_4	7	9
D_5	6	8

n=5 data object

Similarity Measure:- ED

STEP-1:- Find the k and fix it. k=2.

STEP-2:- Randomly choose 2 data points from D and assign it as cluster center or Centroids.

Here, D_1 & D_2.

(1,3) (3,2)

Cluster-1 Cluster 2.

STEP-3:- Calculate the Distance $\sqrt{(x_2-x_1)^2+(y_2-y_1)^2}$ b/w datapoints & cluster center.

Object	Datapoint	cluster center-1 (1,3)	CC 2 (3,2)	Assign Center cluster
D_1	(1,3)	0	$\sqrt{5}$	C_1
D_2	(3,2)	$\sqrt{5}$	$\sqrt{0}$	C_2
D_3	(6,5)	$\sqrt{29}$	$\sqrt{18}$	C_2
D_4	(7,9)	$\sqrt{72}$	$\sqrt{65}$	C_2
D_5	(6,8)	$\sqrt{50}$	$\sqrt{45}$	C_2

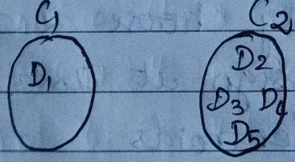
$(6,5)\ (1,3) = \sqrt{(1-6)^2+(3-5)^2}$
$= \sqrt{25+4} = \sqrt{29}$

$(6,5)\ (3,2) = \sqrt{(3-6)^2+(2-5)^2}$
$= \sqrt{9+9} = \sqrt{18}$

$(7,9)\ (1,3) = \sqrt{(1-7)^2+(3-9)^2}$
$= \sqrt{36+36} = \sqrt{72}$

$(7,9)\ (3,2) = \sqrt{(3-7)^2+(2-9)^2}$
$= \sqrt{16+49} = \sqrt{20}$.

STEP-4: Find the cluster center by calculating mean of each cluster.



cluster Mean $C_1 = (1,3)$
cluster Mean $C_2 = (5.5,6)$ $\frac{3+6+7+6}{4}, \frac{2+5+9+8}{4}$

Iteration-2:

Object	Datapoint	$C_1(1,3)$	$C_2(5.5,6)$	cluster
D_1	(1,3)	0	$\sqrt{29.25}$	C_1
D_2	(3,2)	$\sqrt{5}$	$\sqrt{22.5}$	C_1
D_3	(6,5)	$\sqrt{29}$	$\sqrt{1.25}$	C_2
D_4	(7,9)	$\sqrt{72}$	$\sqrt{15.25}$	C_2
D_5	(6,8)	$\sqrt{50}$	$\sqrt{6.25}$	C_2

PAGE No.

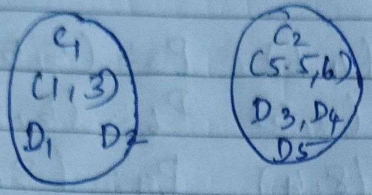
DATE: / /

$(1,3)\ (5.5,6) = \sqrt{(5.5-1)^2+(3)^2}$

$= \sqrt{(4.5)^2+9}$

$(7,9)\ (5.5,6) = \sqrt{(5.5-7)^2+(6-9)^2}$

$= \sqrt{15.25}$



$D_1 = (1,3)$ $\quad D_3\ (6,5)$

$D_2 = (3,2)$ $\quad D_4\ (7,9)$

$D_5\ (6,8)$

Mean $= \frac{1+3}{2}, \frac{3+2}{2}$ $\quad \frac{6+7+6}{3}, \frac{5+9+8}{3}$

$= (2, 2.5)$ $\quad = (6.33, 7.33)$

Iteration 3:

Object	Datapoint	$c_1(2,2.5)$	$c_2(6.33,7.33)$	cluster
D_1	(1,3)	$\sqrt{3.25}$	$\sqrt{46.58}$	C_1
D_2	(3,2)	$\sqrt{1.25}$	$\sqrt{38.98}$	C_1
D_3	(6,5)	$\sqrt{22.25}$	$\sqrt{5.38}$	C_2
D_4	(7,9)	$\sqrt{42.25}$	$\sqrt{5.78}$	C_2
D_5	(6,8)	$\sqrt{30.25}$	$\sqrt{0.79}$	C_2

$(1,3)\ (6.33,7.33) = \sqrt{(6.33-1)^2+(7.33-3)^2}$

$= \sqrt{(5.33)^2+(4.33)^2}$

$= \sqrt{28.4+18.74}$

$= \sqrt{47.14} \simeq \sqrt{46.58}$

$(6,8)\ (6.33,7.3) = \sqrt{(6-6.3)^2+(8-7.3)^2}$

$= \sqrt{(0.3)^2+(0.7)^2}$

$= \sqrt{0.09+0.49}$

$= \sqrt{0.79}$

DATE : / /

Apply the DBSCan algorithm to the given data points and create the cluster with min points = 4 and $\varepsilon = 1.9$ (Epsilon)

Data points:

$P_1 : (3, 7)$ $P_2 : (4, 6)$ $P_3 : (5, 5)$
$P_4 : (6, 4)$ $P_5 : (7, 3)$ $P_6 : (6, 2)$
$P_7 : (7, 2)$ $P_8 : (8, 4)$ $P_9 : (3, 3)$
$P_{10} : (2, 6)$ $P_{11} : (3, 5)$ $P_{12} : (2, 4)$

Min point: 4 $\varepsilon = 1.9$

~~P_1 P_2 P_3 P_4 P_5 P_6 P_7 P_8 P_9~~

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

	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8	P_9	P_{10}	P_{11}	P_{12}
(3,7) P_1	0											
(4,6) P_2	1.41	0										
(5,5) P_3	2.83	1.41	0									
(6,4) P_4	4.20	2.83	1.41	0								
(7,3) P_5	5.66	4.24	2.83	1.41	0							
(6,2) P_6	5.83	4.47	3.16	2.00	1.41	0						
(7,2) P_7	6.40	5.00	3.61	2.24	1.00	1.00	0					
(8,4) P_8	5.83	4.47	3.16	2.00	1.41	2.83	2.24	0				
(3,3) P_9	4.00	3.16	2.83	3.16	4.00	3.16	4.12	5.6	0			
(2,6) P_{10}	1.41	2.00	3.16	4.47	5.83	5.66	6.40	6.3	3.16	0		
(3,5) P_{11}	2.00	1.41	2.00	3.76	4.47	4.24	5.66	5.1	2.00	1.41	0	
(2,4) P_{12}	3.16	2.83	3.16	4.00	5.10	4.47	3.39	6.0	1.41	20	1.41	0

$P_1 \rightarrow P_2, P_{10}$
$P_2 \rightarrow P_3, P_1, P_{11}$
$P_3 \rightarrow P_2, P_4$
$P_4 \rightarrow P_3, P_5$

$P_5 \rightarrow P_4, P_6, P_8, P_7$
$P_6 \rightarrow P_5, P_7$
$P_7 \rightarrow P_5, P_6$
$P_8 \rightarrow P_5$
$P_9 \rightarrow P_{12}$
$P_{10} \rightarrow P_1, P_{11}$

PAGE No.
DATE : / /

$P_{11} \rightarrow P_2, P_{10}, P_{12}$
$P_{12} \rightarrow P_9, P_{11}$

Points	Noise/Core	Border/Not
P_1	Noise	Border
P_2	Core	
P_3	Noise	Border
P_4	Noise	Border
P_5	Core	
P_6	Noise	Border
P_7	Noise	Border
P_8	Noise	
P_9	Noise	
P_{10}	Noise	Border
P_{11}	Core	
P_{12}	Noise	Border