1)

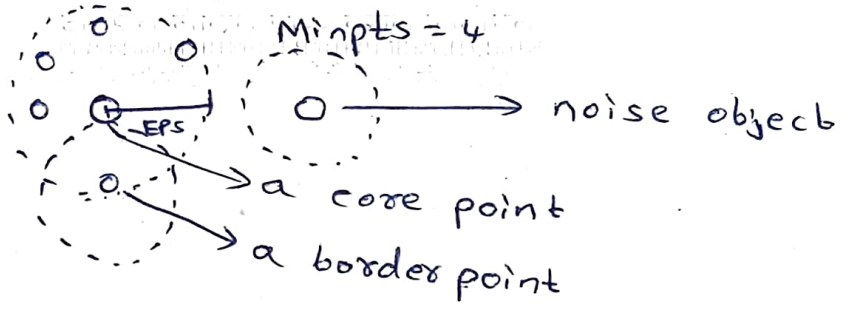
a) Describe the following terms related to DBSCAN clustering algo.

DBSCAN → a clustering algorithm

Parameters: Eps, Minpts

Eps-neighbourhood: The region that is atmax Eps distance from the given point is called Eps neighbourhood of that point

(i) Core object: An object (instance) in the dataset is call a core object if it has atleast Minpts number of objects in its Eps-neighbourhood



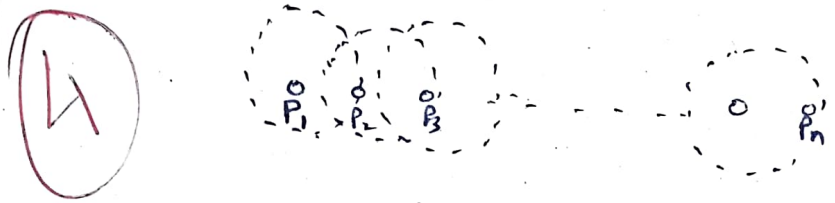
(ii) Border point/object: An object is called a border object if it doesn't have atleast Minpts number of objects in its Eps neighbourhood, but it is in the Eps neighbourhood of some other core point.

Noise object: An object that is not a core object and not a border object is called a noise object

→ outliers and noise.

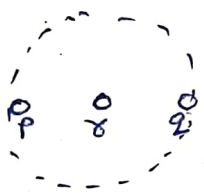
(ii) Density reachability

~~if th~~ ~~fo~~ There exists a sequence of points $P_1, P_2 \ldots P_n$ and P_{i+1} is directly density reachable from P_i for $i = 1$ to $n-1$. Then P_n is density reachable from P_1



(iii) Density connectivity

objects P, q are density connected if there exists a point r such that P, q are density reachable from r.



(b) Discuss the following two conditions which are to be satisfied by the DBSCAN clustering algorithm

(i) Maximality condition

If object P belongs to cluster C and q is density reachable from P, then q should also belong to C

$P \in C$ & AND q density reachable from P $\Rightarrow q \in C$

(ii) Connectivity condition

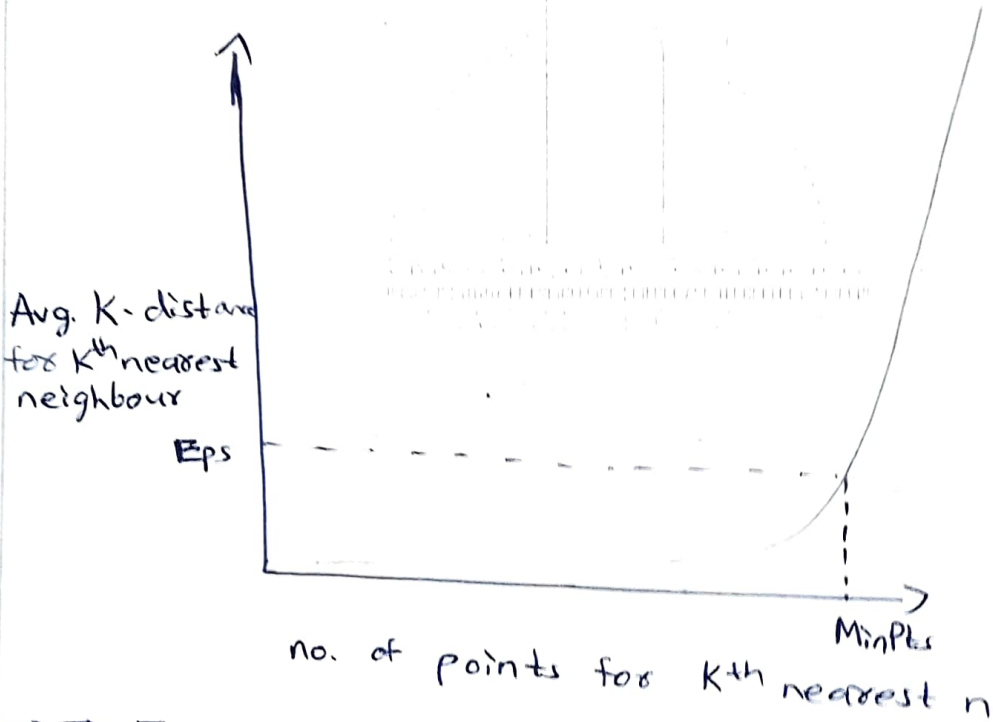
If 2 objects P and q belong to a single cluster

c, then q should be density reachable from p.

(c) Explain how two parameters, Eps and MinPts of DBSCAN algorithm are determined.

2 parameters → Eps
MinPts

→ We have to determine the best values of these 2 points for DBSCAN algorithm.



3

→ In this graph as the no. of points for kth nearest neighbour increases, the average k-distance also increases gradually

→ But as we move along the x-axis, we can see a sudden drastic increase in Avg k-distance with a little increase in no of points for K^{th} nearest neighbour.

→ This is the point from where we get the values of Eps and Minpts of DBSCAN

→ The corresponding value on x-axis is Minpts

→ The corresponding value on Y-axis is Eps

∴ The 2 parameters Eps and Minpts of DBSCAN algorithm are determined.

4)

a) Identify the root of a decision tree for the following dataset

We know that

Information Gain (~~A~~) =

$$= I(P,n) = -\frac{P}{P+n}\log_2\frac{P}{P+n} - \frac{n}{P+n}\log_2\frac{n}{P+n}$$

for root $P=6 \quad n=4$

$$I(6,4) = \frac{-6}{10}\log\frac{6}{10} - \frac{4}{10}\log\frac{4}{10}$$

$$= -\frac{3}{5}[\log_2 3 - \log_2 5] - \frac{2}{5}[\log_2 2 - \log_2 5]$$

$$= -\frac{3}{5}[1.6 - 2.3] - \frac{2}{5}[1 - 2.3]$$

$$= -0.6[-0.7] - 0.4[-1.3]$$

$$= 0.42 + 0.52$$

$$= 0.94$$

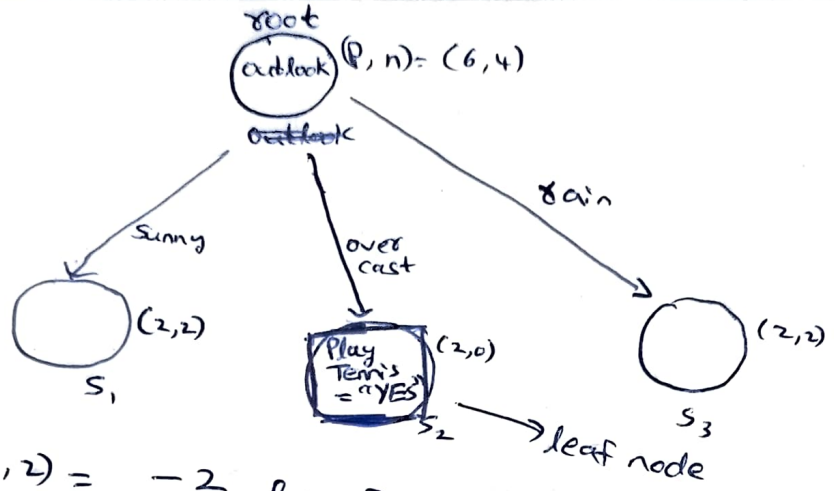
We know that

Entropy = 1

$$E(A) = \sum \frac{P_i + n_i}{P+n} I(P_i, n_i)$$

let us split using Outlook

Sunny = 2,2



$$I(2,2) = -\frac{2}{4}\log\frac{2}{4} - \frac{2}{4}\log\frac{2}{4}$$

$$= -\log\frac{1}{2}$$

$$= -\log 2^{-1}$$

$$= \log_2 2$$

$$= 1$$

$$I(2,0) = -\frac{2}{2}\log\frac{2}{2} - \frac{0}{2}\log\frac{0}{2}$$

$$= 0$$

$$\text{Gain} = I(P,n) - \text{Entropy}(A)$$

$$\text{Entropy}(A) = \frac{4}{10}\times 1 + \frac{2}{10}\times 0 + \frac{4}{10}\times 1$$

$$= \frac{2}{5} + 0 + \frac{2}{5} = \frac{4}{5} = 0.8$$

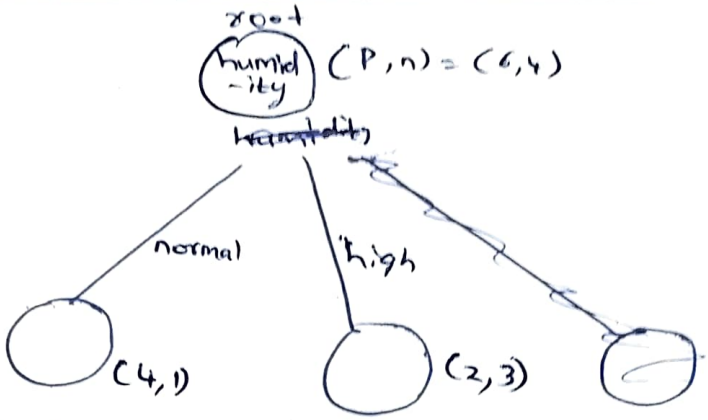
$$\text{Gain} = I(P,n) - \text{Entropy}(A)$$

$$= 0.94 - 0.8$$

$$= 0.14$$

$\therefore$ $\text{Gain}_{outlook} = 0.14$

lets split based on Humidity



$I(P,n) = I(6,4) = 0.94$

$I(4,1) = -\frac{4}{5}\log\frac{4}{5} - \frac{1}{5}\log\frac{1}{5}$

$= -\frac{4}{5}[\log 4 - \log 5] - \frac{1}{5}[\log 1 - \log 5]$

$= -\frac{4}{5}[2 - 2.3] - \frac{1}{5}[0 - 2.3] = +\frac{4}{5} \times 0.3 + \frac{1}{5} \times 2.3$

$= 0.24 + 0.46$

$= 0.7$

$I(2,3) = -\frac{2}{5}\log\frac{2}{5} - \frac{3}{5}\log\frac{3}{5} = -0.4[1-2-3] - \frac{3}{5}[1.6 - 2.3]$

$= 0.52 + 0.42 = 0.94$

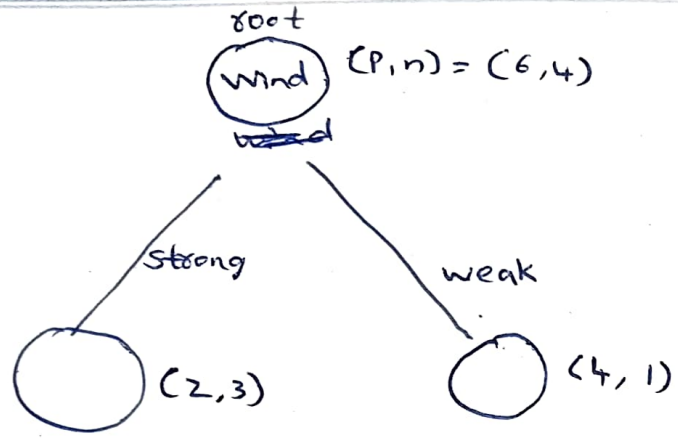
$E(A) = \frac{5}{10} \times 0.7 + \frac{5}{10} \times 0.94 = 0.35 + 0.47 = 0.82$

$Gain_{humidity} = I(6,4) - E(A) = 0.94 - 0.82 = 0.12$

$\therefore Gain_{humidity} = 0.12$

Let's try to split based on Wind

$I(P,n) = I(6,4) = 0.94$



$I(2,3) = 0.94$ $\quad I(4,1) = 0.7$

$E(A) = \frac{5}{10} \times 0.94 + \frac{5}{10} \times 0.7 = 0.82$

$Gain_{wind} = 0.94 - 0.82$

$= 0.12$

Conclusion

So we have seen that

$Gain_{outlook} = 0.14$

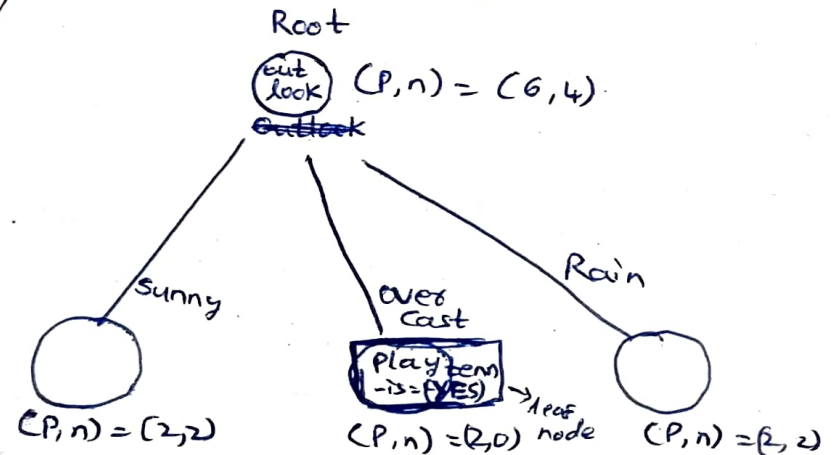
$Gain_{humidity} = 0.12$

$Gain_{wind} = 0.12$

→ We get maximum gain if we split based on outlook.

→ The test at root node is whether the outlook is sunny or overcast or Rain

→ The root is as follows



b) What are the possible ~~exit~~ terminating criteria of a decision tree algorithm

When any of the below 3 criteria, then the decision tree algorithm will stop/terminate.

1) All the instances at the node belongs to the same class. The label of this node (leaf) will be that class

3

2) No more attributes left to perform test on. If more than one class samples present, probability for label is used

3) No samples left in the node.

c) What are the causes of model overfitting? How does it solve in decision tree.

Model overfitting the training data occurs when ~~when~~ there is insufficient generalization of training data.

2½

→ insufficient data in training set

→ coincidental regularities in the training set

→ different distributions in training and testing sets

How to solve it in decision trees

1) Early pruning: Terminate decision tree algorithm before the model overfits the training data.

2) Pruning: Remove branches that represents noise or outliers from a fully grown tree

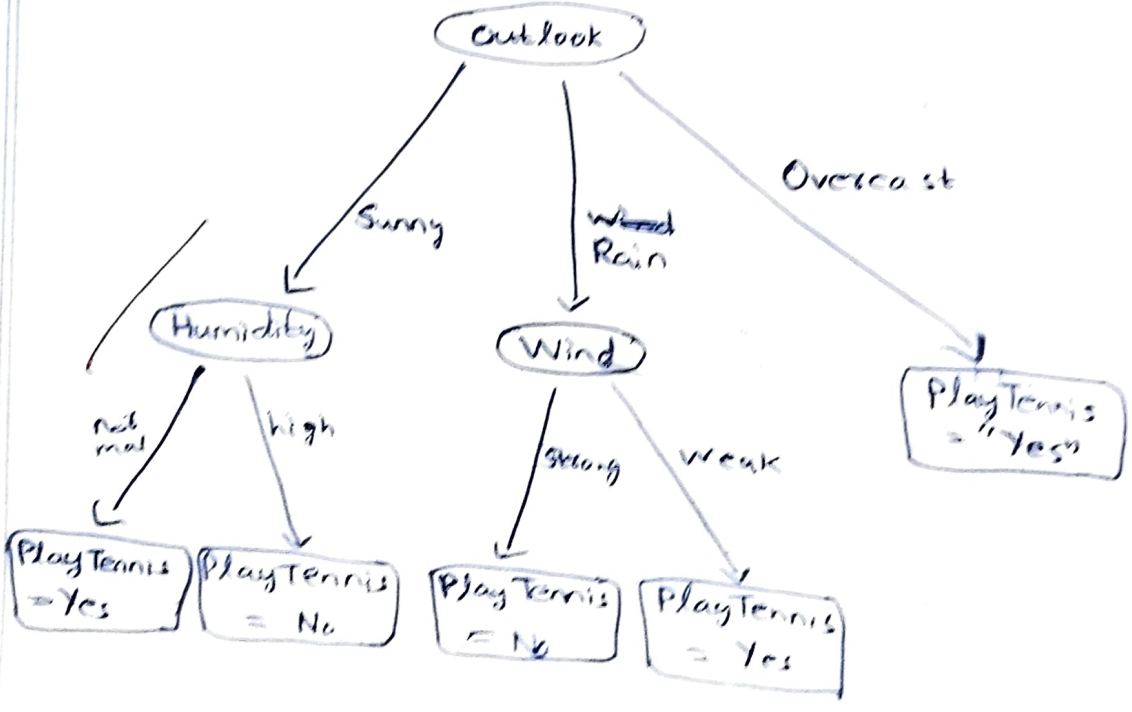
→ This is more useful

→ To identify such branches, use the model to classify a set of instances that are not in training set.

6)

a) Let the decision tree model trained by the dataset given in question 4 (c) is as follows.

Apply the decision tree model on the following test data



Outlook	Humidity	Wind	PlayTennis (actual)	PlayTennis (predicted)	
Sunny	Normal	Strong	Yes	Yes	TP
Overcast	Normal	Strong	No	Yes	FP
Rain	High	strong	Yes	No	FN
Sunny	High	Weak	No	No	TN
Rain	High	Strong	No	No	TN

4

Construct confusion matrix. And compute the following evaluation metrics for the model.

Predicted / Actual	Yes	No
Yes	1 True Positive	1 False Negative
No	1 False Positive	2 True Negative

(i) Precision

$$\text{Precision} = \frac{TP}{TP+FP} = \frac{1}{1+1} = \frac{1}{2} = 0.5$$

(ii) Recall

$$\text{Recall} = \frac{TP}{TP+FN} = \frac{1}{1+1} = \frac{1}{2} = 0.5$$

(ii) F-score

$$F_\beta\text{-score} = \frac{(1+\beta^2) \times \text{precision} \times \text{recall}}{\beta \times \text{precision} + \text{recall}}$$

$$F_1 \text{ score} = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$$

$$= \frac{2 \times \frac{1}{2} \times \frac{1}{2}}{\frac{1}{2} + \frac{1}{2}} = \frac{\frac{1}{2}}{1} = \frac{1}{2} = 0.5$$

b) Describe the following methods that are used for estimating different evaluation metrics of a classification model

(i) Hold out method

In this method, in every iteration, we divide the dataset into training set and testing set with ratio of no. of samples in them as $\frac{2}{3} : \frac{1}{3}$. This is done using random sampling.

After training the model on training set $(\frac{2}{3})$, we test it on testing set $(\frac{1}{3})$ and we calculate error in each iteration

(ii) Cross-validation

→ Techniques such as k-fold validation comes under cross validation

→ The given dataset is divided into k subsets of equal sizes using random sampling

→ In every iteration, ~~one~~ ~~one~~ ~~selection~~ k-1 subsets are selected, they are used as training data and the subset that is left out ~~test~~ is used as testing data set

→ total k iterations

eg: 10-fold cross validation.

(iii) Bootstrap

→ ~~this~~ here we divide the given dataset into training and testing data sets using random sampling with replacement.

(c) What are the main purposes of ensembles of classifiers? How does the Adaboost algorithm work?

Main purpose of ensembles of classifiers

1) The objective is not to build a high accuracy model, but to build a set of low accuracy models whose results can be combined to predict the class labels with high accuracy

Ensembles of classifiers is needed because to overcome the below problems

1) Statistical Problem :- When there are more than 1 hypothesis resulting in same accuracy. If the model chooses any of those hypothesis, then the model can perform poorly on unseen data

2) Computational Problem :- ~~When the best~~ When finding the best ~~heuristic~~ hypothesis cannot be guaranteed considering computational constraints

3) Representational Problem :- Any hypothesis in hypothesis space cannot give a good approximation of target classes.

→ Ensemble of classifiers solves these 3 problems

Adaboost Algorithm

→ an iterative algorithm

→ builds a sequence of models, ~~while~~ every new model focus more on the samples misclassified by previous models

→ The hypothesis are complementary interms of samples that are misclassified.

→ ~~Weighted sum/majority~~ ✓

→ Weighted voting or weighted average of all the models considered for final prediction

→ The weights of each model result depends on the performance of the model.

Pseudocode

~~→ Assi →~~

~~Iterative Algorithm~~ N → no. of samples in data set

1) Assign equal weights $\frac{1}{N}$ to each sample

2) Iterative algorithm

Repeat

→ Build the learning model, Predict for dataset

→ Calculate error rate [using weighted loss function]

→ For every misclassified sample, multiply its weight by $\frac{e}{1-e}$ ⟵ error rate

→ Normalize weights of each sample so that they will sum up to 1.

→ Store the result (predictions) of this model and its accuracy

3) Find weights of each model using its performance.

4) Final prediction is done by taking weighted majority voting / weighted average voting.

So, that's how the Adaboost algorithm works

3)

a) Define support and confidence of an association rule

When an item set is referred to as a frequent item set and an association rule is referred as an important rule

Consider the association rule $X \longrightarrow Y$

Support:- is the measure of how frequently the X ; Y ⇒ i.e. XUY are occurring in the transactions. Probability that X and Y appears in a transaction

$$\text{Support}(X \cup Y) = \frac{(X \cup Y).\text{count}}{N}$$

N → total no. of transactions

Confidence:- is the probability of Y to occur when X is occurring

$$\text{Confidence}(X \rightarrow Y) = \frac{(X \cup Y).\text{count}}{X.\text{count}}$$

$$= P(Y/X)$$

frequent item set :- An item set is called frequent when the support of the item set is atleast Minsupport

(4)

$$\text{Support(itemset)} \geq \text{Minsupport}$$

importat association rule :- An association rule is said to be ~~frequent~~ importat. if the confidence of the association rule is atleast Minconfidence

$$\text{Confidence}(X \rightarrow Y) \geq \text{Minconfidence}$$

b) Minsupport = 30% i.e. 0.3

Minconfidence = 80% i.e. 0.8

Determine frequent ~~ass~~ item set and association rules using well known a-priori algorithm

T_1	Bread, Butter
T_2	Bread, Milk, Butter
T_3	Bread, Jelly, Butter
T_4	Bread, Coke
T_5	Bread, Milk
T_6	Milk, Coke

Min Support = 0.3

C_1 = {Bread}:5 {Milk}:3 {Butter}:3 {Jelly}:1 {Coke}:2

F_1 = {Bread} {Milk} {Butter} {Coke}

5th

C_2 = {Bread, Butter}:3 {Bread, Milk}:2 {Bread, Coke}:1 {Milk, Butter}:1

{Milk, Coke}:1 {Butter, Coke}:0

F_2 = {Bread, Butter} {Bread, Milk}

$C_3 = \phi$

Frequent item sets are {Bread}, {Milk}, {Butter}, {Coke}, {Bread, Butter}, {Bread, Milk}

Association rules

consider {Bread, Butter}

~~Bread → Butter~~ ~~Confidence~~

Confidence (Bread → Butter) = $\frac{3}{5}$ = 0.6 < 0.8

Confidence (Butter → Bread) = $\frac{3}{3}$ = 1 > 0.8

Consider {Bread, Milk}

confidence (Bread → Milk) = $\frac{2}{5}$ = 0.4 < 0.8

Confidence (Milk → Bread) = $\frac{2}{3}$ = 0.66 < 0.8

∴ Important association rule Butter ~~→~~ Bread.

~~Conc~~

Answer

Frequent item sets are

{Bread, Butter}, {Bread, Milk}

{Bread}, {Milk}, {Butter} {Coke}

Important association rules are

Butter → Bread

c) What are the major drawbacks of the a-priori algorithm?

Some of the major drawbacks of a-priori algorithm are

~~1) It take →~~

1) High time complexity!

→ extracting C_i from F_{i-1} takes too much ~~combinatorial~~ time

2) As the no. of items in the $item set increases, the number of association rules explored by the apriori algorithm also increases. thus incurring more time complexity

3)