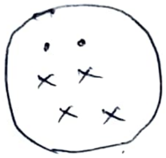
2 Ans :

(a) (i) Purity :

1) Purity is one of the simplest cluster validation indices.

2) It is calculated as the ratio of the number of dominant class ~~labeled~~ (in cluster) to the total size of the cluster.

3) Example -



I

Purity of cluster I : $\frac{\max(2,4)}{6}$

$= \frac{4}{6}$

4) High values of purity indicate that the clustering algorithm produces homogeneous clusters (desired). However, this validation index is biased as one can obtain highest possible purity value by having 'N' clusters, where N is the number of examples.

(ii) Rand Index :

1) Rand Index is calculated using the 'table' given below.

3

	Same cluster	Different cluster
Same ground truth label	A	B
Different ground truth label	C	D

2) The steps to calculate Rand Index are given below.

(1) Generate all possible pairs (of examples).

(2) Compare the ground truth labels of those pairs with the cluster that they belong to (according to the clustering algorithm) and fill up the table accordingly.

(3) Now, the value of Rand Index is given as :

$$\text{Rand Index} = \frac{A+D}{A+B+C+D}$$

4) Rand Index gives us a measure of accuracy of the clustering algorithm.

(b) (i) Intercluster distance:

1) Intercluster distance gives us a measure of seperation between the clusters generated by a clustering algorithm.
2) High value of intercluster distance indicates that the clusters generated are well seperated, which is desirable.
3) Some examples of intercluster distance measurements are :-

(i) Single linkage distance :-

It gives us the distance between the closest objects that are present in different clusters. It is given as:

$$sld :- \min_{\substack{x \in S \\ y \in T}} \{ d(x,y) \}.$$

Here; S and T are two different clusters, and $d(x,y)$ gives the distance (might be Euclidean, Manhattan, Chebyshev etc.) between the objects x and y. 'x' and 'y' belong to S and T respectively.

(ii) Centroid linkage distance:

It gives us the distance between the centroids of two different clusters. It is given as:

$$cld :- d(v_S, v_T)$$

where $v_S = \frac{1}{|S|} \sum_{x \in S} x$ and $v_T = \frac{1}{|T|} \sum_{y \in T} y$.

Here, v_S and v_T represent the centroids of clusters S and T respectively. $|S|$ and $|T|$ represent the number of objects that are present in clusters S and T respectively.

(B) Intracluster distance.

1) Intracluster distance gives us a measure of the compactness of the clusters generated by a clustering algorithm.

2) Low values of intracluster distance indicate that the clusters generated are compact, i.e., the elements (objects) in the cluster are close to each other, which is desired.

3) Examples of intracluster distance measurements are:

(i) Complete diameter distance.

It gives us the distance between the most remote objects that are present in the same cluster. It is given by:

$$cdd = \max_{x_1, x_2 \in S} \{ d(x_1, x_2) \}.$$

x_1 and x_2 are two objects belonging to the cluster S. $d(x_1, x_2)$ gives us the distance between the points x_1 and x_2 (objects)

(ii) Centroid diameter distance.

It gives us twice the average distance between the centroid of a cluster and all other points present in that cluster. It is given by:

5

$$\text{Centroid } dd = 2\left[\frac{\left\{\sum_{x \in S} d(x, v_S)\right\}}{|S|}\right]$$

Here, $v_S = \frac{1}{|S|}\sum_{x \in S} x$ is the centroid of the cluster S and $|S|$ is the number of objects present in the cluster S.

(c) Dunn's cluster validation index:

1. Dunn's cluster validation index is evaluated as shown by the following expression

$$\boxed{D\ index = \min_{1 \le i \le c} \left\{ \min_{1 \le j \le c,\, j \ne i} \left\{ \frac{\delta(X_i, X_j)}{\max_{1 \le k \le c} \{\Delta(X_k)\}} \right\} \right\}}$$

4

2. Dunn's index helps us to validate and evaluate the performance of the clustering algorithm and also helps us to choose the optimum number of clusters.
3. Dunn's index aims to maximize the intercluster distance while minimizing the intracluster distance, thereby generating clusters that are compact and well-seperated.
4. We can evaluate the value of Dunn's index for different number of clusters and choose the number of clusters where the value of Dunn's index is maximized, since high value of Dunn's index indicates good quality clustering.

3 Ans -

(a) Support - 1) We say that a rule holds with support 'sup' if sup % of transactions ~~that contain X~~ contain $X \cup Y$. Here, the rule is of the form $X \rightarrow Y$ where X and Y are itemsets.

4

2) Support = $\frac{(X \cup Y).count}{n}$ where n = no. of transactions

Confidence : 1) We say that a rule holds with confidence 'conf' if conf % of transactions that contain X also contain Y.

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

* An itemset is referred to as a 'frequent itemset' when it has a support value greater than / equal to the minimum support value specified by the user.

* An association rule is referred to as an 'important rule' when it has a confidence value greater than/equal to the minimum confidence value specified by the user.

(b) Ans. Let Bread : 1 . Given, minsup : 30% and minconf : 80%
Butter : 2
Milk : 3
Jelly : 4
Coke : 5

Then, the transaction set can be written as:

Transactions	Items
T1	1, 2
T2	1, 3, 2
T3	1, 4, 2
T4	1, 5
T5	1, 3
T6	3, 5

5

Now, C1 (candidates of size 1) :

	1	2	3	4	5
count	5	3	3	1	2
support	$\frac{5}{6}$ > minsup	$\frac{3}{6}$ > minsup	$\frac{3}{6}$ > minsup	$\frac{1}{6}$ < minsup	$\frac{2}{6}$ > minsup

∴ F1 : 1, 2, 3, 5.

and C2 :- {1,2}, {1,3}, {1,5}, {2,3}, {2,5}, {3,5}.

	{1,2}	{1,3}	{1,5}	{2,3}	{2,5}	{3,5}
Now, count	3	2	1	1	0	1
support	$\frac{3}{6}$ > minsup	$\frac{2}{6}$ > minsup	$\frac{1}{6}$	$\frac{1}{6}$	0	$\frac{1}{6}$

∴ F2 : $\boxed{\{1,2\}, \{1,3\}}$

Now, C3 : {1, 2, 3} has support $\frac{1}{6}$ ∴ F3 : ∅

The possible association rules are :-

$1 \rightarrow 2$	confidence : $2/6 / 5/6$	< min conf
$2 \rightarrow 1$	confidence : $2/6 / 2/6$	> min conf
$3 \rightarrow 1$	confidence : $2/6 / 3/6$	< min conf
$1 \rightarrow 3$	confidence : $2/6 / 5/6$	< min conf

$\therefore$ The important association rule : $\boxed{2 \rightarrow 1}$.

(c) Major drawbacks of the a-priori algorithm :

1) Apriori algorithm has a complexity of $O(2^m)$, where m is the number of transactions. Thus, it is pretty slow.

2) Apriori algorithm generates a large number of association rules (for big datasets, it is in the order of thousands), all of which may not be significant.

3) It ignores important information such as the amount of each item purchased, or the price paid while generating the association rules.

2

4 Ans :

(a) Given dataset has two values of class variable : 'Yes' and 'No'

Information required : $I(6,4)$ [6 Yes and 4 No].

$$= \frac{-6}{10} \log_2\left(\frac{6}{10}\right) - \frac{4}{10} \log_2\left(\frac{4}{10}\right)$$

$$= -0.6[1.6-2.3] - 0.4[1.0-2.3]$$

$$= 0.6 \times 0.7 + 0.4 \times 1.3$$

$$= 0.42 + 0.52 = \boxed{0.94}.$$

$\therefore$ If the dataset is split based on :

Outlook. Entropy : $\frac{4}{10} I(2,2) + \frac{4}{10} I(2,2) + \frac{2}{10} I(2,0)$

$= \frac{8}{10} I(2,2) + 0$

$I(2,2)$ is given as $-\frac{2}{4}\log_2\left(\frac{2}{4}\right) - \frac{2}{4}\log_2\left(\frac{2}{4}\right)$

$= -\frac{4}{4}\log_2\left(\frac{1}{2}\right)$

$= 1$

$\therefore$ Outlook has information gain $= 0.94 - (0.8)(1)$

$= 0.14$

For Humidity :- Entropy $= \frac{5}{10} I(4,1) + \frac{5}{10} I(2,3)$

$I(4,1) = -\frac{4}{5}\log_2\left(\frac{4}{5}\right) - \frac{1}{5}\log_2\left(\frac{1}{5}\right) = 0.8 \times 0.3 + 0.2 \times 2.3$

$= 0.7$

$I(2,3) = -\frac{2}{5}\log_2\left(\frac{2}{5}\right) - \frac{3}{5}\log_2\left(\frac{3}{5}\right) = 0.4 \times 1.3 + 0.6 \times 0.7$

$= 0.94$

$\therefore$ Entropy $= 0.5 \times 0.7 + 0.5 \times 0.94$

$= 0.35 + 0.47 = 0.82$

$\therefore$ Humidity has information gain $= 0.94 - 0.82$

$= 0.12$

5

For Wind :- Entropy $= \frac{5}{10} I(2,3) + \frac{5}{10} I(4,1)$.

$= 0.82$.

$\therefore$ Wind has information gain $= 0.12$.

$\therefore$ The root of the decision tree is 'Outlook'.

(b) Possible terminating criteria of a decision tree:

1) If all the instances / objects at a node have the same class label, the decision tree is terminated.

3

2) If there are no more attributes available to perform the splitting the decision tree is terminated and node is decided by the majority rule

3) If there are no instances left, the decision tree is terminated.

(c) Causes of model overfitting:

1) When the model constructed is more complex than the original / true function (which generated the data), the model is said to 'overfit' the data.

2) Some of the causes of model overfitting are:
   1. Insufficient generalization of training data
   2. Presence of noise (noisy data)
   3. Insufficient training data
   4. Too few attributes / unrelated attributes, etc.

3) The methods employed to solve overfitting in decision trees are:

4

1. Prepruning: The growth of the tree is stopped early, i.e., before a goodness measure drops below a certain threshold.

   Examples of pre-pruning criteria: Number of instances in a node, Depth of tree etc.

2. Post pruning: The tree is allowed to grow fully and after that, nodes (branches) are removed if it increases the performance of the tree on a certain validation set.

6 Ans

(a) Given,

Values of class variable.	Predicted class variable (decision tree)
Yes	Yes
No	Yes
Yes	No
No	No
No	No.

4

∴ The confusion matrix is as shown below.

	Actual Positive	Actual Negative
Predicted Positive	1 TP	1 FP
Predicted Negative	1 FN	2 ~~TP~~

∴ Precision = $\frac{TP}{TP+FP} = \frac{1}{1+1} = \frac{1}{2}$.

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

F_score = $\frac{2}{\frac{1}{Precision}+\frac{1}{Recall}} = \frac{2}{\frac{1}{(1/2)}+\frac{1}{(1/2)}} = \frac{2}{4} = \frac{1}{2}$.

(b) (i) Holdout method:

1) In holdout method, the entire dataset is split into three different parts. The majority of the dataset (ex: 60%) is used for training the model and is called the training set. The remaining portion is split into testing set (ex: 30%) and validation set (ex: 10%).

(ii) Cross-validation: (k-fold)

1) In this method, the entire training set is divided into k-parts. The model is trained on (k−1) parts and is evaluated on the k^{th} pa
This is repeated until the model is trained and evaluated on all k parts.

(iii) Bootstrap:

3

1) In this method, the dataset is sampled 'n' times (n is the size of the dataset). with replacement such that after the sampling, around 68% of the instances form the training set and ~32% form the testing set. This is the case when all the instances are equ-ally likely to get picked. If this is not the case, then the composit of the training and test sets changes accordingly.

(c) 1) The main purpose of ensembles of classifiers is to increase/obtain high accuracy by combining the predictions of each of the base classif
2) Here, the main goal is not to use base classifiers with high accur-acy values, rather than using base classifiers that make different kinds of errors, i.e., misclassify different training instances. Theref the accuracy obtained by combining the predictions of such classifiers be much higher (even when the individual base classifier accuracy is low).

3) Adaboost algorithm works by combining the predictions of progressively trained models, each of which has a different amount of weight in the final predictions. The models are trained in such a way that newly trained models are encouraged to become experts in classifying the misclassified inst-ances of previously trained models. Also, depending upon the performance of a model, its weight is decided $\left(-\log\left(\frac{e}{1-e}\right)\right.$; where e is the error rate). Initially, every training instance has equal weight. After training a model, the instances that are incorrectly classified have their weight increased (or, the instances that are correctly classified have their weight decreased). This leads to increased error if these instances are misclassified by future models.

3