Q2 ——→ ?

a.) After clustering is done in the given data, the clusters need to be validated by various means. Discussing one of them,

ii) <u>Rand Index</u> : Rand index is calculated as -

$1\frac{1}{2}$

$$\boxed{RI = \frac{A+B}{A+B+C+D}}$$

A : No. of pairs of data points lying in ~~both~~ the same clusters (<u>True Positive</u>)

B : " " " " " " lying in different clusters (<u>True Negative</u>)

C : No. of pairs of data points earlier in one cluster, then in different clusters (<u>False Negative</u>).

D : " " " " " " " " different clusters, then in same clusters (<u>False Positives</u>)

- So, technically its a ratio of all true positives / negatives to all the other possible pairs. It clearly validates the clustering.

b.) <u>Intercluster Distance</u> : Intercluster Distance is essentially the distance b/w two different clusters, which can be measured in many ways. Two of them are discussed here

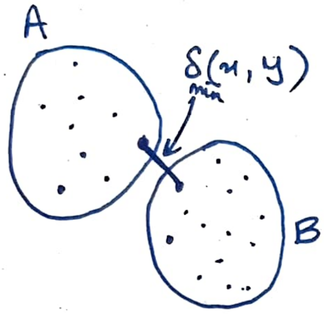
- <u>Minimum Intercluster Distance</u> : It is the distance b/w two nearest data points in two separate clusters.

P.T.O.

- Suppose x, y are two data points such that -

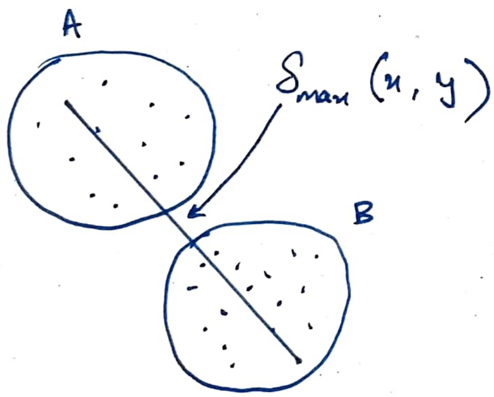
$x \in A$
$y \in B$ where A, B are two separate clusters, then

Min. Intercluster distance $= \min(\delta(x, y))$



- Maximum Intercluster Distance: It is the maximum possible intercluster distance b/w two clusters ie → distance b/w two farthest data points.

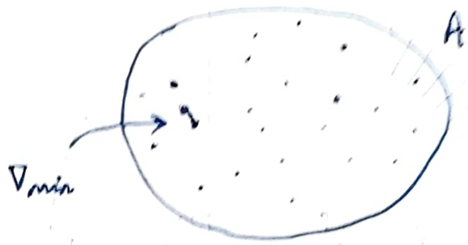
So, Max. Intercluster distance $= \max(\delta(x, y))$



Intracluster Distance: It is essentially the measure of two data points distance in the same cluster, which can be measured in many ways. Two of them are -

- **Minimum Intracluster Distance**: It is the distance b/w two nearest data points in the same cluster.

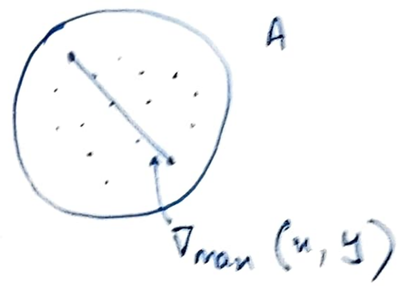
$\nabla_{min}(x, y) \Leftarrow$ Minimum Intracluster Distance



$x, y \in A$

- **Maximum Intracluster Distance**: The maximum possible diameter distance b/w two data points in the same cluster.

$\nabla_{max}(x, y) \Leftarrow$



$x, y \in A$

c.) ## Dunn's Cluster Validation Index :

- Dunn's cluster Validation Index is defined as -

$$\boxed{DI = \min_{1\leq i\leq k}\left(\min_{\substack{1\leq j\leq k \\ i\neq j}}\left(\frac{\delta(X_i, X_j)}{\max(\nabla X_k)}\right)\right)}$$, where

3

i, j : indices

k : k clusters

X_i : i^{th} cluster

δ : Inter Cluster Distance

∇ : Intra Cluster Distance

- The Dunn's cluster validation index primarily focus on the inter & intra cluster distance. For proper clustering, the intra cluster distance should be as low as possible & intercluster distance should be as large as possible.

- Greater the value of Dunn's Index, more efficient is the clustering.

- But there's a catch, even for fixed k values, multiple clusterings are possible which can be further validated by some other means.

Q3 — 1

a.) Support of an Association Rule: Support is defined as the ratio of number of times that association is valid out of the all possible data set entries.

Confidence of an Association Rule: Let, $A \rightarrow B$ is an association rule then,

4

$$\boxed{Sup(A \cup B) = \frac{n(A \cup B)}{N}}$$

; $n(A \cup B)$: No. of entries for which the rule is valid.

N : Total items in data set

$$\boxed{Conf(A \rightarrow B) = \frac{Sup(A \cup B)}{Sup(A)}}$$

- An item set is said to be a frequent item set when the item set has support greater than or equal to the minimum support decided (minsup)

An association rule is said to be an important rule when confidence of that rule is greater than or equal to the (minConf) minimum confidence decided.

b.) We have -

min Sup : 30 %

min Conf : 80 %

Given Transaction :

T_1 : Brad, Butter

T_2 : Bread, Milk, Butter

T_3 : Bread, Jelly, Butter

T_4 : Bread, Coke

T_5 : Bread, Milk

T_6 : Milk, Coke

Acc. to Apriori Algorithm, we start with single Items with support ≥ min Sup.

Note : C_K : Candidate Itemset of size K

F_K : Final Itemset of Size K

4

STEP 1 :

Total transactions = 6

min n(Item) $\geq \frac{30}{100} \times 6$

≥ 1.8

$F_1 \Rightarrow$ {Bread} : 5 ✓

{Butter} : 3 ✓

{Milk} : 3 ✓

{Jelly} : 1

{Coke} : 2 ✓ (Choose)

$C_2 \Rightarrow$ {Bread, Butter} : 3 ✓ (Choose)

{Bread, Milk} : 2 ✓

{Milk, Butter} : 1

{Bread, Coke} : 1

{Milk, Coke} : 1

$F_2 \Rightarrow$ {Bread, Butter}: 3, {Bread, Milk} : 2

$C_3 \Rightarrow$ {Bread, Butter, ~~Bread~~, Milk} : 1

So, The Frequent itemset is [{Bread, Butter}, {Bread, Milk}]

$\therefore$ The association rule is :

Confidence of [{Bread, Butter} $\longrightarrow$ {Milk}] $= \dfrac{\text{Sup}(\{\text{Bread, Butter, Milk}\})}{\text{Sup of }\{\text{Bread, Butter}\}}$

$$= \frac{1}{3} \approx 33\,\% < \text{minConf.}$$

So, no association rule can be formed.

c) Major drawbacks of Apriori Algorithm is that it has higher Time Complexity, as on each step it has to sweep over the whole data set once. Thereby, taking up much longer times if data set is large.

$1\frac{1}{2}$

- Plus, on each step all possible combination of itemsets (with sup > minSup) is created, so if it runs for too long, in worst case could easily go out of time bounds.

Q4 —?

a.) For a decision Tree, to decide the decision node or splitting point, we have to calculate "Information Gain" or "Gini Index"

- So, the attribute with highest Information Gain is selected as the root node. And the same process continues for entire tree.

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

(Information required to classify p & n elements in class P & N resp.)

For Outlook Attribute :

$$I(\text{Sunny, Rainy, Overcast}) = -\frac{\cancel{4}^{2}}{\cancel{10}_{5}}\log_2 \frac{\cancel{4}^{2}}{\cancel{10}_{5}} - \frac{\cancel{4}^{2}}{\cancel{10}_{5}}\log_2 \frac{\cancel{4}^{2}}{\cancel{10}_{5}} - \frac{\cancel{2}^{1}}{\cancel{10}_{5}}\log_2 \frac{\cancel{2}^{1}}{\cancel{10}_{5}}$$

$$= -\frac{2}{5}\times(\log 2 - \log 5) - \frac{2}{5}(\log 2 - \log 5) - \frac{1}{5}(\log 1 - \log 5)$$

$$= -\frac{2}{5}\times(-1.3) - \frac{2}{5}(-1.3) - \frac{1}{5}(-2.3)$$

$$= \frac{5.2}{5} + \frac{2.3}{5} = \frac{7.5}{5} = 1.5$$

For Attribute Humidity :

$$I(\text{Normal}, \text{High}) = -\frac{5}{10}\log_2\frac{5}{10} - \frac{5}{10}\log_2\frac{5}{10}$$

$$= -\frac{1}{2}\times(-1) - \frac{1}{2}\times(-1)$$

4

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

For Attribute Wind :

$$I(\text{Strong}, \text{Weak}) = -\frac{5}{10}\log_2\frac{5}{10} - \frac{5}{10}\log_2\frac{5}{10}$$

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

Similarly, Entropy, $E(p_i, n_i) = \sum_{i=1}^{n} \frac{p_i+n_i}{p+n} \cdot I(p_i, n_i)$

→ Information Gain : $\boxed{I(p,n) - E(p,n)}$

- Since, attribute "Outlook" has the highest information index, so it is better to select that as the root.

b.) The possible termination criteria of a decision tree can be—

- When maximum depth of tree is reached.
- All elements gets classified.
- No attribute is left to be selected.

2

- If no proper termination of decision tree is done, then it may lead to Overfitting.

c.) Overfitting in a decision tree occurs when it is not terminated properly and allowed to run over the training data for too long.

- The tree tries to compensate for the noise present in the training data set and so, instead to learning the relationships among elements, it just memorize the results.

$1\frac{1}{2}$

- It may give good results for the provided training data set but fails to predict / classify any future data accurately.

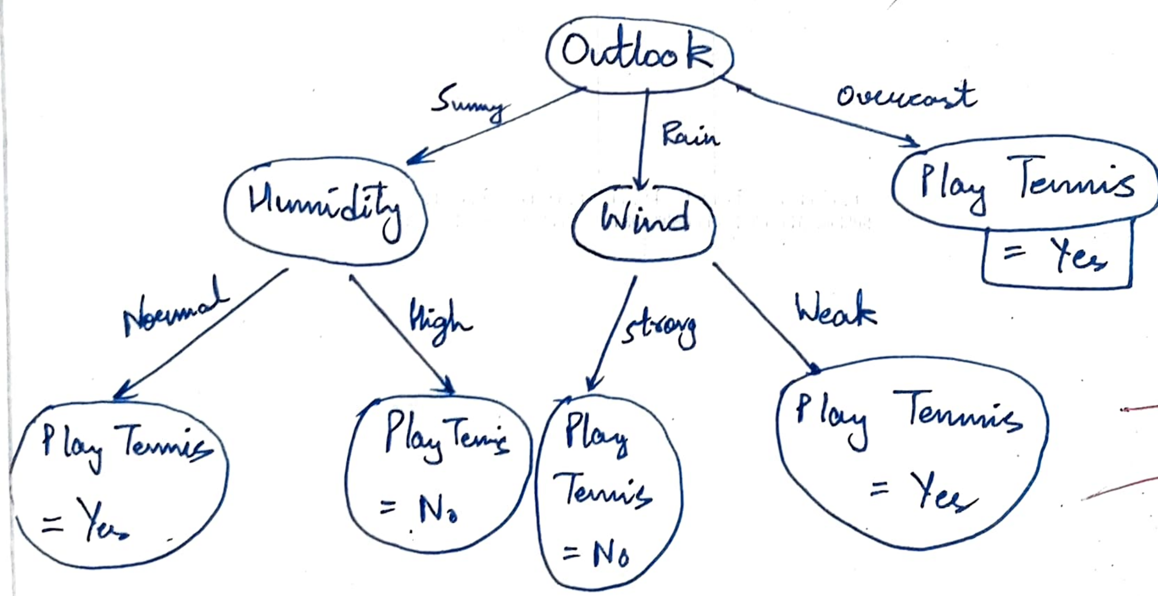
- Only remedy for prevention of Model Overfitting is to determine proper termination rules for it. It may be when maximum depth is reached or when no attribute is left to be used to split.

- Or we can even make sure that the training data set is as noise free as possible.

Q6 ——— ?

a.) Confusion Matrix is the mathematical representation of any anomaly that occurs when a decision tree is applied to some unknown data and it contradicts the rules determined by the tree.

We have the tree -



- Applying it to given data -

Outlook	Humidity	Wind	Play Tennis
Sunny	Normal	Strong	Yes ✓ (Holds)
Overcast	Normal	Strong	No ✗ (Anomaly)
Rain	High	Strong	Yes ✗ (Anomaly)
Sunny	High	Weak	No ✓ (Holds)
Rain	High	Strong	No ✓ (Holds)

b.) ii.) Cross Validation:

- Whenever a classification model is developed, it is a good practice to generate the model on a few subsets of the training data & then validating the result by checking over the remaining subsets.

- This ensures robustness of the model and is called cross validation.

c.) When a classifier is designed, an optimisation technique is adopted known as "Ensembles of Classifiers".

- The main working of ensembles can be understood by the analogy that it helps in generating models by using the outcome of many classifiers generating models over a training data; the final result being a majority vote or average of the results obtained from individual classifiers.

- Ensembles can be Independent (Bagging / Vote / Random Noise / Feature Selection), Co-ordinative (Boosting, Stacking) developed-

- AdaBoost or Adaptive Boost Algorithm is based on the co-ordinated ensembles classifier.

In this algorithm,

- Initially all the classifiers have the same weight.
- After each iteration, the error 'e' is calculated for each classifier. If $e=0$ or $e \geq 0.5$, then the classifier weight is unchanged.

2 Otherwise, multiply the weight by $\boxed{\log\left(\frac{e}{1-e}\right)}$

- Repeat the process.

- Let me explain, the AdaBoost essentially ignores the classifiers which have more precise results after each iteration and adapts the weights of the rest of the classifiers so that more focus is given to the rest of the classifiers.

- Major advantage of this algorithm is its adaptive nature & resilience to noise. But still, if proper termination is not kept in mind, then overfitting can occur.

EOF