Note: Answer all questions. Make appropriate assumptions. Give a brief answer. ... question is for 2 marks. You can use the space in the extra page to provide answer.

1	Explain how information gain measure is biased towards attributes having large number of values.	
	Information Gain = Info (D) − $Info_A$(D) where A → attribute chosen And it tends toward purity (one class belonging to just yes or no or maximise towards it) Now, if our attribute is Unique_ID, it will be diff. for every tuple and hence be the best attribute to choose for decision tree, but in actual it won't give any information on it. Hence, we use information_split = $- \sum \frac{\lvert D_j \rvert}{\lvert D \rvert} \log_2 \frac{\lvert D_j \rvert}{\lvert D \rvert}$ and then Gain Ratio = $\frac{\text{Info Gain}}{\text{Info\_split}}$ to reduce this biaseness towards multivalued attributes. Now, Unique_ID will have large no. of splits, hence Info_split huge, ⇒ Gain Ratio small.	1.78
2	Explain the term "underfitting". Discuss the causes of underfitting and the methods to avoid it.	
	Underfitting refers to model training when the model is not able to understand and capture the underlying trends & patterns in the dataset. (Low Bias, High Variance) Perform poor on both training & test Dataset. Causes of Underfitting: → Imbalanced Dataset → Data leakage → complex attributes To avoid it we can → Generalisation ~~(Generalise attributes)~~ → Normalization → choose attributes that are actually useful eg. Covid No. from Patient History dataset instead of it's SSR No.	1.5

3 Explain the basic idea of Rainforest (a scalable decision tree classifier).

Decision Tree Classifier: Creates the decision tree, by iterating greedly with divide-and-Conquer approach.

Rainforest: ~~There are certain issu~~ Even though decision tree is highly explainable & inherently interpretable, it has certain issues like → Accuracy → Multi-valued attributes, etc

Hence, we scale our ~~decision~~ decision tree with diff algo's to reduce this.

Rainforest:

X 0

Decision tree
↳ MDL
↳ ESS
↳ CHAID

4

$$\boxed{P(C \mid A_1 A_2 \ldots A_n) = \frac{P(A_1 A_2 \ldots A_n \mid C)P(C)}{P(A_1 A_2 \ldots A_n)}}$$

Interpret the above formula with a real life example.

$$\underbrace{P(C|A_1 A_2 \ldots A_n)}_{\text{Posterior Prob}} = \frac{\underbrace{P(A_1 A_2 \ldots A_n|C)}_{\text{likelihood}} \; \underbrace{P(C)}_{\text{Prior Prob}}}{\underbrace{P(A_1 A_2 \ldots A_n)}_{\text{evidence}}}$$

→ Naive Bayes Classifier

Example, fruits can have multiple features like color, taste, cost, etc.

$$P(\text{Apple}|\text{Color, taste, Cost}) = \frac{P(\text{Color, taste, Cost}|\text{Apple})\, P(\text{Apple})}{P(\text{Color, taste, Cost})}$$

It Calculates the Condⁿ Probability using Bayes Classifier.
If it's naive Bayes Classifier and all attributes ~~apple~~, Color, taste, Cost are Independent of each other →

$$P(\text{Apple}|\text{Color, taste, Cost}) = \frac{P(\text{Color}|\text{Apple}) \cdot P(\text{taste}|\text{Apple}) \cdot P(\text{Cost}|A)\, P(A)}{P(\text{Color}) \cdot P(\text{taste}) \cdot P(\text{Cost})}$$

Hence, it can be read as: Prob. of Apple given we know the values of Color, taste & Cost is equal to Prob. Color, taste, and Cost given we know the fruit is apple, multiplied by prob. of fruit being apple in any case, whole divided by Prob. of any fruit being of this color, taste, and cost.

2 Nice!!

5 Explain the purpose of Laplacian correction in Naive Bayesian classifier.

In Naive Bayesian Classifier, as shown in Ques 4 we assume all attributes are Independent of each other.

$$P(C|A_1,A_2 \ldots A_n) = \frac{P(A_1|C_1)\cdot P(A_2|C_2)\ldots P(A_n|C_n)\cdot P(C)}{P(A_1)\cdot P(A_2)\ldots P(A_n)}$$

2

But if any one $P(A_i|C_i) = 0$ then whole conditional Prob. will become zero.

Hence, in huge datasets, we assume that one row exist for each case (adding one row in huge datasets don't affect that Much).

Hence, if $P(A_i|C_i) = \frac{N_i}{N} = 0$ → No. of elements in A_i, → No of total elements

After Laplace Corr → $P(A_i|C_i) = \frac{N_i + 1}{N + n}$, assuming n attributes

Now even if $N_i = 0 \Rightarrow P(A_i|C_i) \neq 0$ leading us to prevent the posterior prob. connecting this.

6 Explain the classification metrics, which will help you to understand the extent of class imbalance aspect of the classifier.

Class Imbalance: When a particular class has labels for one side comparatively very huge in number compared to other sides.

Eg. Cancer dataset Containing 97% No and 3% Yes cases. So, even if accuracy = 97% doesn't means it classifying no cases properly.

	Y	N
Y	TP	FN
N	FP	TN

Metrics Used: 1) Precision = How many positives are actually true positive

$$\text{Precision} = \frac{TP}{TP+FN}$$

2) Recall = % of tuples of +ve that are correctly classified as +ve

$$\text{Recall} = \frac{TP}{TP+FP}$$

1 Specificity ?

For Correction this class Imbalance, we use LIMO, SHAP, etc.

7 State the stream classification problem. Discuss how the notion of ensemble is extended to build stream classification?

Stream Classifier: Stream of data comes continuously ~~with~~ but we have a fixed storage space. It can be TV, running radio, etc.

Issues: • Fixed Storage Space
• Need to process a whole batch at Once ~~is broken~~
• ~~Context from previous batches~~ ~~are removed from~~ ~~as prev. batches are removed from~~

Ensemble: Create Multiple Models and then ~~dispose~~ either ~~sequentially~~ parallely (Bagging) or ~~Parallely~~ sequential (Boosting) predict the Best Model.

Here, what we do is → ~~(Sequential)~~

~~Here~~ 1) Let the stream Input come in ~~Best to this Input.~~

~~2) Then we find the Model~~ ~~performing worse from~~

~~3) And remove the model performing worse from the storage.~~

~~2) We update the Model with this Input if it increases the goodness of metrics considered.~~
~~↳ else the Input's discarded.~~

~~3) The parts in storage performing worse in metrics compared to this Input are removed, if this one is added.~~

~~4) And hence, the model is trained.~~

dividing into multiple chunks / classifiers

1.5

~~This is~~

2) Then we find the model performing best ~~according to goodness of Metrics to~~ to this Input.

3) ~~And remove~~ If the goodness of Metrics considered for this Model worse than the existing models → Discard this Model.
→ else update this Model in storage and remove the one performing worse from the existing Model.

4) Repeat this until the input Comes.

5) Then you will have the best j models at the end, use majority-vote, etc for choosing the best Model.

Ensemble

Intuition correct

1.5

8 Explain how the notion of Error-Correcting Codes are used to improve Multiclass Classification.

Multiclass Classification: Instead of Binary (0|1) classifier, we have more class labels.

eg. One +ve, and the rest all -ve's, etc.

We perform this classification through (K-NN (K nearest Neighbours), then it will use Euclidean distance. Hamming distance

Let 110000 / 000011 be two class labels

Dist. from 111111 same for both

$\sqrt{\Sigma(x_i - x_f)^2}$

But Numbers completely different.

~~Hence, we use error correcting codes (for ex. Hamming codes) to correct this.~~

0.5

eg →

Let 4 attributes be A, B, C, D $\{1,1,1,1\}$ / $\{1,3,2,4\}$

Case I → $P(a) = P(b) = P(c) = P(d) = \frac{1}{2}$ → uni-class

i) Prob = $4 \times \frac{1}{2} = 2$

ii) Prob = $\frac{1}{2}[1+3+2+4] = 5$

Case 2 → $P(a) = 1,\ P(b) = 3,\ P(c) = 1,\ P(d) = 2$ → multi-class

i) Prob = $1+3+1+2 = 7$

ii) Prob = $-\log_2 \frac{1}{4} - \log_2 \frac{3}{4} - \log_2 \frac{2}{4} - \log_2 \frac{4}{4}$

So, we use error-correcting codes (for ex → Hamming codes) to correct this.

0.5