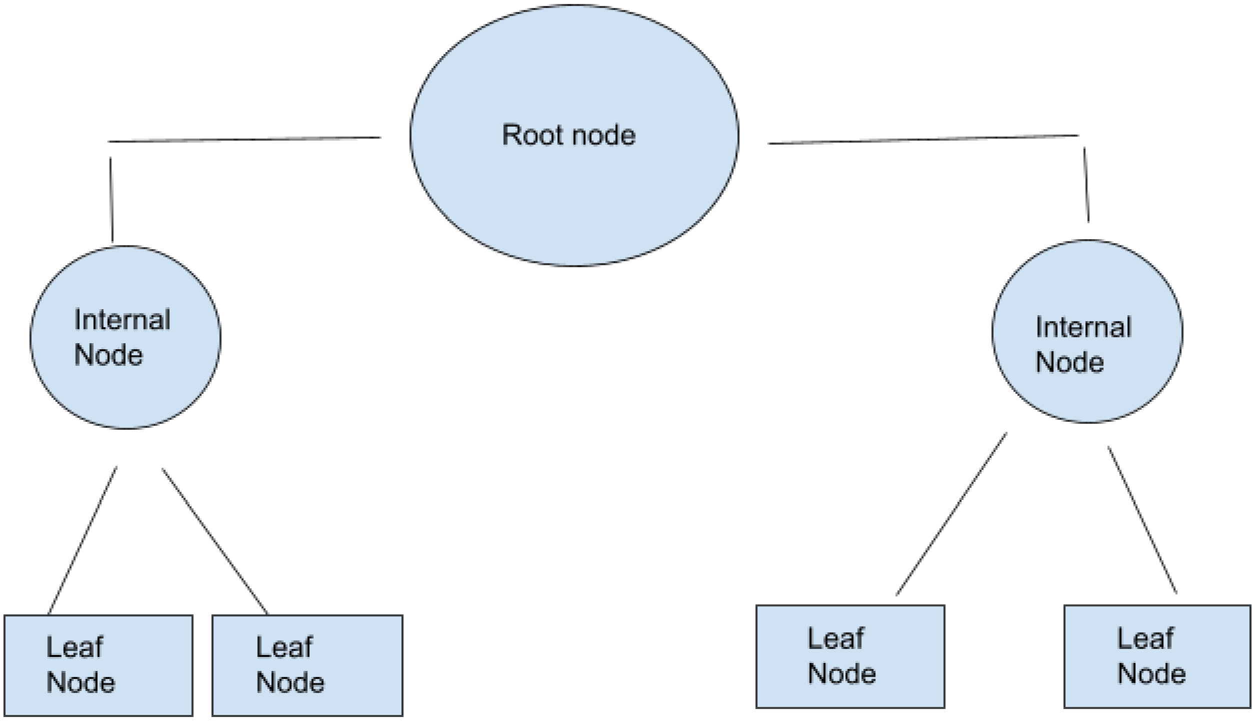
Theory [25 pts]

(a) Explain what a decision tree is. How does it make classification predictions? [5 pts]

Answer: Decision Tree is a very powerful ML algorithm which is easy to implement and interpret. Decision Tree is a supervised machine learning algorithm which is widely used for classification and regression tasks. It has a tree structure and is hierarchical. A decision tree consists of branches, root node, internal node and leaf nodes.



Internal Nodes are also known as Decision Nodes.

Decision Tree classification predictions work in the way where the algorithm uses a series of nested “if-else” statements to check all possible conditions are checked until the model reaches a final solution.

On each node of the Decision Tree we try to create a condition on the features to separate all the labels or classes contained in the dataset to the complete purity.

(b) In a classification problem, using your own words explain how attributes are split. What mathematical principle can be used to define a good split? [5 pts]

Answer: In a classification problem, a decision tree makes decisions by splitting nodes into sub-nodes. This process is repeated until only homogenous nodes are left. To define a good split we can use Gini purity or entropy. In both cases lower value means a better split.

(c) What is information gain and why is it useful in the training of a decision tree? [5 pts]

Answer: In Decision Tree learning, information gain is used to evaluate the quality of a split. Information gain represents how much information a split presents about the target variable. It is useful because it helps to identify the best attribute to split the data at each node.

(d) Given training, validation, and test sets, explain the process of selecting the best hyperparameters of a classification algorithm. [5 pts]

Answer: to select the best hyperparameters of a classification algorithm, we should follow the below steps:

- □ Split the data into training, validation, and test sets
- □ Train the ML algorithm on the training set with different hyperparameter settings
- □ Evaluate the model performance on the validation set and select the hyperparameters which have the best performance on the validation set

(e) If a Decision Tree is overfitting the training set, is it a good idea to try decreasing or increasing maximum depth? Why? [5 pts]

Answer: When the Decision Tree is overfitting the training set, we should try decreasing the maximum depth not increasing it.

By decreasing the maximum depth, we reduce the complexity of the model and prevent overfitting. This can also help to improve the performance of the model on the validation and test sets. It can also make it easier to understand.

(f) Information theory. Recall the definition of the entropy of a discrete random variable X with probability mass function p: [10 bonus pts/optional]

$$H(X) = \sum p(x)\log2(1/p(x))$$

Here the summation is over all possible values of $x \in X$, which (for simplicity) we assume is finite (e.g., 1, 2, ..., N).

Prove that the entropy H(X) is non-negative [1 pt]

Answer:

To prove that the entropy H(X) is non-negative, we need to prove that H(X) >= 0

Let's consider random variable X with four possible outcomes:

X = {x1, x2, x3, x4}

and the probability mass function p is:

p(x1) = 0.2
p(x2) = 0.3
p(x3) = 0.1
p(x4) = 0.4

Using the formula for entropy, we can calculate the entropy of X as follows:

$$H(X) = \sum p(x)\log2(1/p(x))$$

H(X) = p(x1)log2(1/p(x1)) + p(x2)log2(1/p(x2)) + p(x3)log2(1/p(x3)) + p(x4)log2(1/p(x4))

H(X) = 0,2 log2(1/0.2) + 0,3 log2(1/0.3) + 0,1 log2(1/0.1) + 0,4 log2(1/0,4)

H(X) = 0,2 log2(5) + 0,3 log2(10/3) + 0,1 log2(10) + 0,4 log2(5/2)

H(X) = 0,2 * 2,3219 + 0,3 * 1,7369 + 0,1 * 3,3219 + 0,4 * 1,3219

H(X) = 0,4644 + 0,5211 + 0,3322 + 0,5288

H(X) = 1.8465

Since the probabilities are non-negative and sum up to one, the logarithms in the formula are non-negative. As a result, the entropy H(X) is always non-negative, greater than or equal to zero.