Theory -

Question 1: What is a Decision Tree, and how does it work in the context of classification?

Ans : A Decision Tree is a type of supervised machine learning algorithm that is mainly used for classification and regression tasks. In classification, it helps predict the category or class of a data point based on input features.

The structure of a decision tree is similar to a flowchart. It starts at the top with a root node, which represents the entire dataset. From there, the data is split into branches using decision rules based on feature values. Each split leads to a new internal node or a leaf node, which holds the final prediction.

How it works:

1. The algorithm looks for the feature that best divides the data into classes.
2. It uses metrics like Gini Impurity or Information Gain to determine the best splits.
3. This process continues recursively, creating new branches until stopping criteria are met (like maximum depth or pure leaves).

Example: Suppose we want to classify whether a person will buy a product or not based on their age and income. The tree might first split by age (>30 or <=30), then by income level.

Conclusion: Decision Trees are easy to understand and interpret. They mimic human decision-making, making them popular in business and educational settings.

Question 2: Explain the concepts of Gini Impurity and Entropy as impurity measures. How do they impact the splits in a Decision Tree?

In a decision tree, Gini Impurity and Entropy are used to measure how mixed the classes are in a dataset. These help the algorithm decide where to split the data for the best classification.

1. Gini Impurity:

Measures the probability of wrongly classifying a randomly chosen element. Formula: (Gini = 1 - p_i^2) where (p_i) is the probability of class (i). A Gini value of 0 means perfect classification.

2. Entropy:

- Comes from information theory. Measures disorder or uncertainty.
- Formula: (Entropy = -p_i _2(p_i))
- Entropy is highest when classes are equally mixed.

Impact on Splits:

- The decision tree selects the feature and threshold that results in the greatest reduction in impurity (either Gini or Entropy).
- This helps create pure child nodes where samples mostly belong to one class.

Example: If a node has 10 class A and 10 class B samples, impurity is high. A good split will create child nodes like one with 9A, 1B and another with 1A, 9B.

Question 3: What is the difference between Pre-Pruning and Post-Pruning in Decision Trees? Give one practical advantage of using each.
Pre-Pruning (Early Stopping):
- Stops the tree from growing too large during training.
- It uses rules like max_depth, min_samples_split, or min_samples_leaf to limit growth.
- Prevents overfitting by simplifying the tree early.

Advantage:
- Faster training time since it avoids building a large tree unnecessarily.

Post-Pruning:
- First builds a full tree, then removes branches that do not improve accuracy.
- Also called cost-complexity pruning.
- Advantage: Leads to a more accurate and generalized model, since pruning is done after seeing the full data. Conclusion: Both methods help avoid overfitting. Pre-pruning saves time, while post pruning improves model performance.

Conclusion: Both methods help avoid overfitting. Pre-pruning saves time, while post pruning improves model performance.
Question 4: What is Information Gain in Decision Trees, and why is it important for choosing the best split?
Information Gain is a metric used to choose the feature that best splits the dataset in a Decision Tree.
It measures the reduction in entropy after a dataset is split based on a feature. The idea is that a good split gives us more “pure” groups.
Formula: (IG = Entropy(parent) - Entropy(child))
Why it’s important:
- A higher information gain means better separation between classes.
- The tree selects the feature with the highest information gain at each step.

Example:
If we split data by the feature “Age > 30”, and this split results in two child nodes where each node has mostly one class, the entropy decreases and information gain increases.
Conclusion: Information Gain helps build trees that make better decisions by focusing on the most informative features.

Question 5: What are some common real-world applications of Decision Trees, and what are their main advantages and limitations?
Applications:
- 1. Healthcare: Diagnosing diseases based on symptoms.
- 2. Finance: Approving loans based on credit score, income.
- 3. Marketing: Predicting customer churn or product purchase.
- 4. Education: Predicting student performance.

Advantage
● Easy to understand and visualize
● Can handle both numerical and categorical data
● Requires little data preprocessing (no need for normalization)
Limitations:
● Prone to overfitting on noisy data
● Small changes in data can change the structure drastically
● Greedy approach may not lead to the optimal tree
Conclusion:
Decision Trees are powerful tools for classification and regression tasks, especially when interpretability is important.

Question 6 to 9
https://colab.research.google.com/drive/1qVgNwsApOaSTU7G6qF72ZgVK8r8ja50B?usp=drive_link

Question 10 :
Step-by-Step Process:
1. Handling Missing Values:
o Use imputation methods like SimpleImputer to fill missing values.
o Mean for numerical features, most frequent or mode for categorical ones.
2. Encoding Categorical Features:
o Use OneHotEncoder or LabelEncoder based on whether features are nominal or ordinal.
3. Training Decision Tree Model:
o Use DecisionTreeClassifier() from scikit-learn.
o Train the model on the cleaned dataset.