10 Decision Tree

- Decision trees are a popular machine learning algorithm used for both classification and regression tasks
- They work by recursively splitting the data into subsets based on feature values, ultimately forming a tree-like model of decisions

01 Decision Tree Structure

- **Root Node :** The top node of the tree, representing the entire dataset. The tree splits from this node based on the feature that provides the best split
- **Internal Nodes :** Nodes within the tree that represent decisions based on features. Each internal node corresponds to a feature test
- **Leaf Nodes (Terminal Nodes) :** The final nodes in the tree, representing the output or class label for classification tasks, or a continuous value for regression tasks

02 Entropy

- Entropy measures the randomness or impurity in the dataset
- It quantifies the uncertainty involved in predicting the class label
- Lower entropy indicates a more homogeneous (pure) set, while higher entropy indicates more disorder (impurity)

  $\text{Entropy}(S) = -\sum_{i=1}^{n} p_i \log_2(p_i)$
  - where (p_i) is the proportion of samples belonging to class (i) in the set (S)

03 Information Gain

- Information gain measures the reduction in entropy or impurity after a dataset is split based on a feature
- A higher information gain means a more effective split by the feature

  $\text{Information Gain}(S, A) = \text{Entropy}(S) - \sum_{v \in \text{Values}(A)} \frac{|S_v|}{|S|} \text{Entropy}(S_v)$

04 Gini Index (Gini Impurity)

- The Gini Index is another measure of impurity used in decision trees

- It calculates the probability of incorrectly classifying a randomly chosen element if it were randomly labeled according to the distribution of labels in the dataset
- A Gini Index of 0 indicates perfect purity (all instances in a node belong to a single class), while a Gini Index closer to 0.5 indicates maximum impurity (equal distribution among classes)

  $\text{Gini}(S) = 1 - \sum_{i=1}^{n} p_i^2$

05 Gain Ratio

- Gain ratio is an extension of information gain, which addresses the bias of information gain towards features with a large number of distinct values
- It normalizes the information gain by the intrinsic information
- The gain ratio provides a more balanced assessment when deciding which feature to split on, avoiding bias towards attributes with many distinct values

  $\text{Gain Ratio}(S, A) = \frac{\text{Information Gain}(S,A)}{\text{Intrinsic Information}(A)}$

06 Stopping Conditions

Decision trees require stopping conditions to prevent them from growing too large and overfitting

- **All Samples in a Node Belong to One Class :** If all instances in a node belong to the same class, that node becomes a leaf node with the class label
- **No More Features :** If no more features are left to split the data, the node becomes a leaf node. The class label is determined by the majority class in that node
- **Maximum Depth Reached :** The tree reaches the maximum depth specified by the user
- **Minimum Samples per Node :** A node contains fewer samples than the minimum number specified by the user, and further splitting is not performed
- **No Further Information Gain :** The tree stops if splitting further does not reduce impurity (entropy or Gini index)

07 Pruning

- Pruning is the process of removing sections of the tree that are not necessary for classification, typically done after the tree has been fully grown
- It helps prevent overfitting
  - **Pre-Pruning :** The tree stops growing early based on predefined conditions

- **Post-Pruning :** The tree is fully grown and then pruned by removing branches that provide little to no improvement

Decision Tree Algorithm

1. **Start with the entire dataset**
2. **Select the best feature** using an impurity measure (e.g., Entropy, Gini Index)
3. **Split the dataset** into subsets based on the selected feature
4. **Repeat** the process for each subset, creating a tree structure
5. **Stop** when you meet the stopping criteria (e.g., pure nodes, max depth)
6. **Prune** the tree if necessary to improve generalization and reduce overfitting