Decision Trees

Introduction

Decision Trees are a widely used supervised learning algorithm that can handle both classification and regression tasks. They represent decision-making as a tree-like structure, where each internal node applies a decision rule based on a feature, branches correspond to possible outcomes, and leaf nodes provide the final prediction. For classification, Decision Trees assign discrete labels to data points, while Regression Trees predict continuous numerical values. These models are especially effective in capturing non-linear relationships between features and target variables.

Structure of a Decision Tree

A Decision Tree consists of the following components:

- Root Node: The starting point, representing the entire dataset.
- Internal Nodes: Decision points where data splits based on feature values.
- Branches: Represent the possible outcomes of each decision rule.
- Leaf Nodes: Contain the final predictions (class label or numerical value).

How Decision Trees Work

1. Splitting the Data

At each node, the algorithm selects the feature and threshold that best partitions the data, aiming to increase homogeneity within the resulting subsets.

2. Metrics for Splitting

For Classification:

- Gini Impurity: Measures the likelihood of incorrect classification if a random sample is chosen:

  $$Gini = 1 - \sum_i p_i^2 \tag{1}$$

  where p_i represents the proportion of samples belonging to class i.

- Entropy: Measures the disorder or uncertainty in a dataset:

$$H(S) = -\sum_i p_i \log_2 p_i \tag{2}$$

  The Information Gain is calculated as:

$$IG = H(\text{parent}) - \sum \frac{N_i}{N} H(\text{child}_i) \tag{3}$$

For Regression:

- Mean Squared Error: Measures variance within a node:

$$MSE = \frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y})^2 \tag{4}$$

  where y_i is the actual value, $\hat{y}$ is the predicted value, and N is the number of samples.

3. Recursive Splitting

The tree grows recursively until stopping criteria are met:

- Maximum tree depth is reached.
- Minimum number of samples per leaf node is satisfied.
- Further splits do not significantly improve homogeneity.

4. Making Predictions

- For Classification: Assigns the majority class within a leaf node.
- For Regression: Uses the mean target value of the leaf node.

Algorithm Steps

1. Select the best feature and threshold to split the dataset.
2. Recursively repeat the process for each subset until stopping criteria are met.
3. Assign predictions based on the majority class (classification) or mean target value (regression) at each leaf node.

Key Characteristics

- Type: Supervised learning model for classification and regression.
- Interpretability: Decision trees are easy to visualize and understand.
- Scalability: Can handle large datasets but may require pruning or ensemble methods to generalize well.

Strengths

- Simple to interpret and visualize.
- Handles both categorical and numerical data.
- No need for feature scaling.
- Provides insights into feature importance.

Weaknesses

- Prone to overfitting if the tree grows too deep.
- Sensitive to small changes in the dataset (can lead to different tree structures).
- Biased toward features with more unique values.

Applications

Decision Trees are widely applied across industries:

- Medical Diagnosis: Identifying diseases based on patient symptoms.
- Stock Market Prediction: Forecasting stock prices using historical data.
- Customer Segmentation: Categorizing customers based on purchasing behavior.
- Real Estate Pricing: Estimating house prices based on features like location and size.

Conclusion

Decision Trees and Regression Trees are powerful and interpretable models used in many real-world applications. While they are prone to overfitting, pruning and ensemble methods significantly enhance their generalization ability. Their balance between interpretability and predictive power makes them a valuable tool in machine learning.