Decision Tree
Epoch IIT Hyderabad

Ankit Saha
AI21BTECH11004

15 Jul 2022

1. Introduction

Decision tree is a supervised machine learning algorithm that is used for both classification and regression. Decision tree learning is an intuitive algorithm because it mimics the human thought process.

A decision tree consists of two types of nodes: decision nodes and leaf nodes. A decision node has at least one child node. These are the nodes where a decision takes place based on the input variables and the tree branches out based on the outcome. Leaf nodes are the nodes without any children. These are the nodes that represent the final output produced by the model.

We are given a bunch of inputs and their corresponding labels. The input variables can be either categorical or real-valued or a combination of both. Our objective is to build a decision tree from this data that can best assign a label to a new data point.

2. The Classification and Regression Tree Algorithm

The Classification and Regression Tree algorithm, also known as the CART algorithm, is one of the most popular tree building algorithms. The algorithm consists of the following steps:

i) Start at the root node and choose the feature among the input data set that best classifies our target variable
ii) If the leaf obtained from the decision node is pure (perfect classification), then it becomes a leaf node and there is no further branching from this node. Else, choose the best classifying feature among the remaining features and continue branching
iii) Keep creating sub-trees recursively until there are only pure nodes or there are no more features left to branch out on

The CART algorithm uses something known as the Gini impurity index to measure the purity of a node that is used to select the best classifying feature. The Gini impurity index of a feature is the weighted average of the impurity indices of its leaves.

3. Mathematical Formulation

The Gini impurity index of a feaure is equal to the probability that a sample is misclassified if it is randomly classified based on the distribution of the classification of the given input samples by that feaure. Thus, the lower the index, the better is the feature for classification.

Let there be N_j input samples in total for leaf j of feature $\mathcal{F}$ which are classified into m categories. Let $n_{ji},\ i \in [m]$ denote the number of samples classified into the i^{th} category. The Gini impurity index of a leaf is given by

$$\mathbf{I}_G(j) = \sum_{i=1}^{m} p_{ji}(1 - p_{ji}) \tag{3.1}$$

$$= \sum_{i=1}^{m} \frac{n_{ji}}{N_j}\left(1 - \frac{n_{ji}}{N_j}\right) \tag{3.2}$$

$$= \frac{1}{N_j}\sum_{i=1}^{m} n_{ji} - \frac{1}{N_j^2}\sum_{i=1}^{m} n_{ji}^2 \tag{3.3}$$

$$= \frac{1}{N_j}N_j - \sum_{i=1}^{m}\left(\frac{n_{ji}}{N_j}\right)^2 \tag{3.4}$$

$$= 1 - \sum_{i=1}^{m} p_{ji}^2 \tag{3.5}$$

Observe that if a leaf is pure, $\exists i \in [m]$ such that

$$p_{jk} = \begin{cases} 1 & k = i \\ 0 & k \neq i \end{cases} \tag{3.6}$$

$$\implies \mathbf{I}_G(j) = 1 - 1 = 0 \tag{3.7}$$

Now, the Gini impurity index of a feature $\mathcal{F}$ that branches to N leaves is given by the weighted average of the indices of its leaves

$$\mathbf{I}_G(\mathcal{F}) = \sum_{j=1}^{N} \mathbf{I}_G(j)\frac{N_j}{N} \tag{3.8}$$

$$= \sum_{j=1}^{N}\left(1 - \sum_{i=1}^{m}\left(\frac{n_{ji}}{N_j}\right)^2\right)\frac{N_j}{N} \tag{3.9}$$

$$= \frac{1}{N}\sum_{j=1}^{N} N_j - \frac{1}{N}\sum_{j=1}^{N}\sum_{i=1}^{m}\left(\frac{n_{ji}}{N_j}\right)^2 N_j \tag{3.10}$$

$$= 1 - \frac{1}{N}\sum_{j=1}^{N}\sum_{i=1}^{m}\frac{n_{ji}^2}{N_j} \tag{3.11}$$

Therefore, at each step of the iteration, we are choosing the best classifying feature as

$$\hat{\mathcal{F}} = \arg\max_{\mathcal{F}} \mathbf{I}_G(\mathcal{F}) \tag{3.12}$$

4. Questions

i) Define a tree.

ii) What is the maximum number of nodes a binary decision tree can have if the input vector has L features?

iii) How is a decision taken when a feature is continuous real-valued?

iv) How to prevent overfitting in decision tree learning?

v) Calculate the Gini impurity index for a feature $\mathcal{F}$ that has the following two leaves

	Category A	Category B	Category C
Leaf 1	4	2	6
Leaf 2	0	3	5

5. Answers

i) A tree is defined as a connected acyclic undirected graph.

ii) There can be a maximum of $L+1$ levels for L features. The levels can contain a maximum of $1, 2, 4, \ldots, 2^L$ nodes each respectively. Thus, the maximum number of nodes in the entire tree is given by

$$1 + 2 + 4 + \cdots + 2^L = \frac{2^{L+1} - 1}{2 - 1} = 2^{L+1} - 1 \tag{5.1}$$

iii) The values of the feature for every input data point can be sorted and the sliding window midpoints for each consecutive pair of values can be calculated. For each midpoint, the decision rule can be a classification based on whether a value is smaller or greater than it. The Gini impurity index of the feature will be the minimum of the impurity indices of all midpoints.

iv) There can be overfitting when the leaf nodes contain too few points for us to make a reasonable decision rule. To prevent this, we can choose a threshold for the number of samples and only those leaves that exceed the threshold can contribute to the decision rule.

v) The Gini impurity index of leaf 1 is given by

$$\mathbf{I}_G(1) = 1 - \left(\frac{4}{12}\right)^2 - \left(\frac{2}{12}\right)^2 - \left(\frac{6}{12}\right)^2 \tag{5.2}$$

$$= 1 - \frac{1}{9} - \frac{1}{36} - \frac{1}{4} \tag{5.3}$$

$$= \frac{22}{36} = \frac{11}{18} \tag{5.4}$$

Similarly, the Gini impurity index of leaf 2 is given by

$$\mathbf{I}_G(2) = 1 - \left(\frac{0}{8}\right)^2 - \left(\frac{3}{8}\right)^2 - \left(\frac{5}{8}\right)^2 \tag{5.5}$$

$$= 1 - 0 - \frac{9}{64} - \frac{25}{64} \tag{5.6}$$

$$= \frac{30}{64} = \frac{15}{32} \tag{5.7}$$

Therefore, the Gini impurity index of the feature $\mathcal{F}$ is given by

$$\mathbf{I}_G(\mathcal{F}) = \frac{11}{18} \cdot \frac{12}{20} + \frac{15}{32} \cdot \frac{8}{20} \tag{5.8}$$

$$= \frac{11}{30} + \frac{3}{16} \tag{5.9}$$

$$= \frac{133}{240} \approx 0.554 \tag{5.10}$$