Decision Tree Classifier

Subham Roy
Department of Computer Science and Engineering
University at Buffalo, Buffalo, NY 14260
subhamro@buffalo.edu

1 Overview

A classification tree is used to predict a qualitative response. For a classification tree, we predict that each observation belongs to the most commonly occurring class of training observations in the region to which it belongs. In interpreting the results of a classification tree, we are often interested in the class prediction corresponding to a particular terminal node region, along with the class proportions among the training observations that fall into that particular region.

2 About Decision Tree Classification

A Decision Tree classification algorithm is one of the most popular machine learning algorithms. It uses a tree like structure and their possible branches to solve a particular problem. It belongs to the category of supervised learning where it can be used for both classification and regression purposes.

A decision tree is a structure that includes a root node or the starting/topmost point of the tree, branches, and leaf nodes. Each internal node denotes a test on a predictor, each branch denotes the outcome of a test, and each leaf node holds a particular class label.

The terms involved in Decision Tree algorithm are as follows:-

I. Root Node
II. Splitting
III. Decision Node
IV. Leaf/Terminal Node
V. Pruning
VI. Branch/Sub-Tree
VII. Parent and Child Node

The primary challenge in the Decision Tree algorithm implementation is to identify the attributes which we consider as the root node. This process is known as the attribute selection. There are different attribute selection measures to identify the attribute which can be considered as the root node at each level.

There are 2 popular attribute selection measures. They are as follows:-

I. Information gain (In our analysis we will work using this selection measure)
II. Gini index

Before we understand the concept of Information Gain, we need to first know another concept called "Entropy". Entropy is the measure of the impurity in the given dataset. Information gain is the decrease in Entropy. Information gain computes the difference between entropy before split and average entropy after split of the dataset based on given parameter values.

The formula for Entropy is represented as-

$$\text{Entropy} = \sum_{i=1}^{c} -pi * log2(pi)$$

Here, c is the number of classes and p_i is the probability associated with the i^{th} class.

The ID3 (Iterative Dichotomiser) Decision Tree algorithm uses entropy to calculate information gain. So, by calculating decrease in entropy measure of each parameter we can calculate their information gain. The attribute with the highest information gain is chosen as the splitting attribute at the node.

3 Experiment

The Dataset used in this experiment was taken from Kaggle <https://www.kaggle.com/datasets/elikplim/car-evaluation-data-set>

Initially, we start with loading the Dataset and checking the dimensions and the number of variables within the Dataset. In this case the dataset has 7 variables and a dimension of 1727 rows and 7 columns.

Next step is we perform data preprocessing. We perform the following steps before we can be certain that we can apply the regression model. This is almost similar for all the techniques we apply. First, we check for missing data, then check for the spread of the data that means where the Mean, Median of the data, are there any outliers, we take care of that. Now that we have cleaned the dataset, we check for categorical data, and if needed we format that to binary using One Hot Encoding or simply sequentially replacing it with factors. This helps the regression model evaluate all the parameters without any bias.

In this case study, the dataset had Null values and improper Column. Firstly, we assigned proper column names based on the Dataset's attributes description. Here we provided the following column names - ['buying', 'maint', 'doors', 'persons', 'lug_boot', 'safety', 'class']. Now, before removing the null values, we first split the dataset into Training and Test datasets with an 80:20 ratio respectively. Here we chose 'class' to be our response variable and the rest as the predictors. After this we transform X_train and X_test respectively with ordinal encoders which basically means we converted them to factors because the values in the dataset were categorical. One might ask why it is was needed to transform separately, rather than transform the whole dataset. This is because in reality we do not have the Test set. If we transform the whole set, we skew the outcome which we do not want. Just so that we can simulate the reality, we transform the training and test set separately. Now we fix the Null value issue. We firstly check for any significant outliers. Since we did not find any outliers, we replaced the Null values with Mean of their respective columns.

Finally, in the next step, by using X-test dataset we predict y-test.

4 Analysis

After the experiment we found some information about our prediction from this classification tree. Firstly, the model accuracy for all the predictions is over 96%, which was the best overall accuracy between the two Classification models we tested. The below plot depicts the Confusion matrix for the prediction. Confusion Matrix is generally used to validate the Type I and Type II errors occurred during the predictions. Here also, we see excellent results, as we see that both False Positives(Type I) and False Negatives(Type II) occurred only a few times, and the diagonal element in the matrix depicts the number of accurate predictions which we can see is very high–

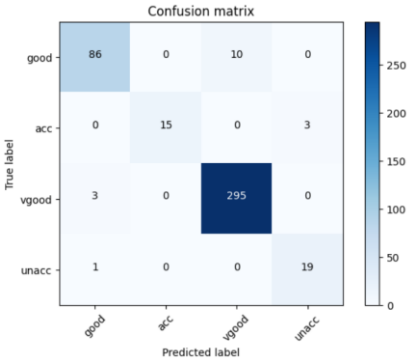

Fig: Confusion Matrix

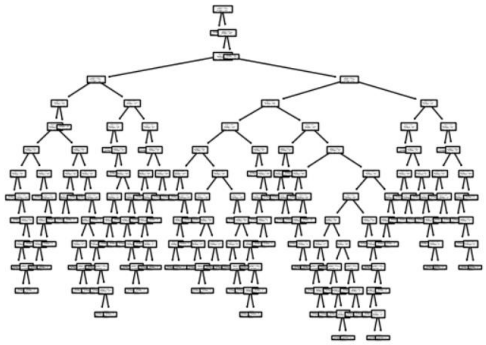

Fig: Decision Tree Classification

The above plot shows the Decision tree plot for the respective classes, but since there are so many branches, the text is not clear. Overall, it has very low error for any model and is great and it shows it is very suitable for this analysis.

References

[1] Gareth James [at], Daniela Witten [aut], Trevor Hastie [aut, cre], Rob Tibshirani [aut], Balasubramanian Narasimhan [ctb], *Introduction to Statistical Learning, Second Edition*.

[2] Tom M. Mitchell, *Machine Learning: A multistrategy approach, 1997 Edition*.