**BTCOL606: Machine Learning Lab**

# Experiment NO.: 03

*Instructor:* Mr Arvind Sardar.

*Name: Abdul Rehman , Roll No:CS3101*

**AIM**

**Implement and demonstrate the Decision Tree Algorithm**

**INTRODUCTION TO DECISION TREE ALGORITHM**

A decision tree is a type of machine learning algorithm that is used for classification and regression analysis. It is a tree-like model that represents a set of decisions and their possible consequences. Decision trees are widely used in data analysis, data mining, and machine learning because they are easy to understand and interpret.

Decision tree learning is a method for Approximating Discrete-Valued Target Functions, in which the learned function is represented by a decision tree. The structure of a decision tree consists of nodes and branches. The nodes represent decisions or tests, while the branches represent the possible outcomes or consequences of those decisions.

The tree is constructed by recursively splitting the dataset into smaller and smaller subsets based on the values of the input features, until a decision can be made about the class or value of the target variable.

There are different types of decision trees, including binary trees, multi-way trees, and regression trees. Binary decision trees have two branches at each node, while multi-way trees can have more than two branches. Regression trees are used for continuous data, while classification trees are used for categorical data.

The construction of a decision tree involves choosing the best attribute to split the data at each node, which is determined by a metric such as information gain, Gini impurity, or entropy. Overall, decision trees are a powerful tool in machine learning and data analysis because they can be used to make accurate predictions and are easy to interpret and visualize. There is no single formula for a decision tree, as the structure and content of the tree depend on the specific problem and dataset being analyzed. However, there are several key formulas and metrics that are commonly used in the construction and evaluation of decision trees. Here are a few examples:

**Information Gain (IG):** This formula is used to select the best feature to split the data at each node. It measures the reduction in entropy or uncertainty that results from splitting the data based on the feature. The formula for information gain is:

$$IG = Entropy(parent) - [weightedaverage] * Entropy(children)$$

**Gini Impurity:** This formula is another measure of the impurity or uncertainty of a set of data. It is commonly used in decision trees to evaluate the quality of a split. The formula for Gini impurity is:

$$Gini = 1 - [(p1)^2 + (p2)^2 + ... + (pk)^2]$$

**Entropy:** Entropy is a measure of the impurity or uncertainty of a set of data. It is commonly used in decision trees to evaluate the quality of a split. The formula for entropy is:

$$Entropy = -(p1 * log2(p1) + p2 * log2(p2) + ... + pk * log2(pk))$$

Here, pi represents the proportion of samples in a given class, and k is the number of classes.

**Advantages of the Decision Tree**

1. It is simple to understand as it follows the same process which a human follow while making any decision in real-life.
2. It can be very useful for solving decision-related problems.

3. It helps to think about all the possible outcomes for a problem.

### Disadvantages of the Decision Tree

1. The decision tree contains lots of layers, which makes it complex.
2. It may have an overfitting issue, which can be resolved using the Random Forest algorithm.
3. For more class labels, the computational complexity of the decision tree may increase

## DATASET DECRIPTION :

A decision tree is a machine learning algorithm that is used for classification and regression analysis, and as such, it requires a dataset to train and test the algorithm. The dataset used for a decision tree will depend on the specific problem and application being addressed, as well as the type of decision tree being used (e.g., binary tree, multi-way tree, regression tree, classification tree). Datasets are used in a variety of applications, such as research, data analysis, and machine learning. In order to be useful, a dataset must be well-organized and annotated, with clear descriptions of the data and its features.

In order to evaluate the performance of the decision tree, the dataset will also need to be split into a training set and a testing set. The training set is used to build the decision tree, while the testing set is used to evaluate the accuracy of the tree's predictions on new, unseen data.

## IMPLEMENTATION CODE FOR DECISION TREE ALGORITHM

```
# Import necessary libraries
from sklearn . datasets import load_iris
from sklearn . tree import Decision Tree Classifier , plot_tree from
sklearn . model_selection import  train_test_split import matplotlib .
pyplot as plt

# Load the Iris dataset iris =
load_iris ()

# Split the dataset into training and testing sets
X_train , X_test , y_train , y_test = train_test_split ( iris . data , iris . target , test_size
    =0.3 , random_state =42)

# Create a decision tree classifier and fit it to the training data clf = Decision
Tree Classifier ()
clf . fit ( X_train , y_train )

# Plot  the  decision  tree plt .
figure ( figsize =(10 ,8) )
plot_tree ( clf , filled = True )
plt . show ()

# Make predictions on the testing set and calculate accuracy predictions
= clf . predict ( X_test )
accuracy = clf . score ( X_test , y_test ) print (
f" Accuracy : { accuracy }")

```

**Output Of Decision Tree Algorithm**

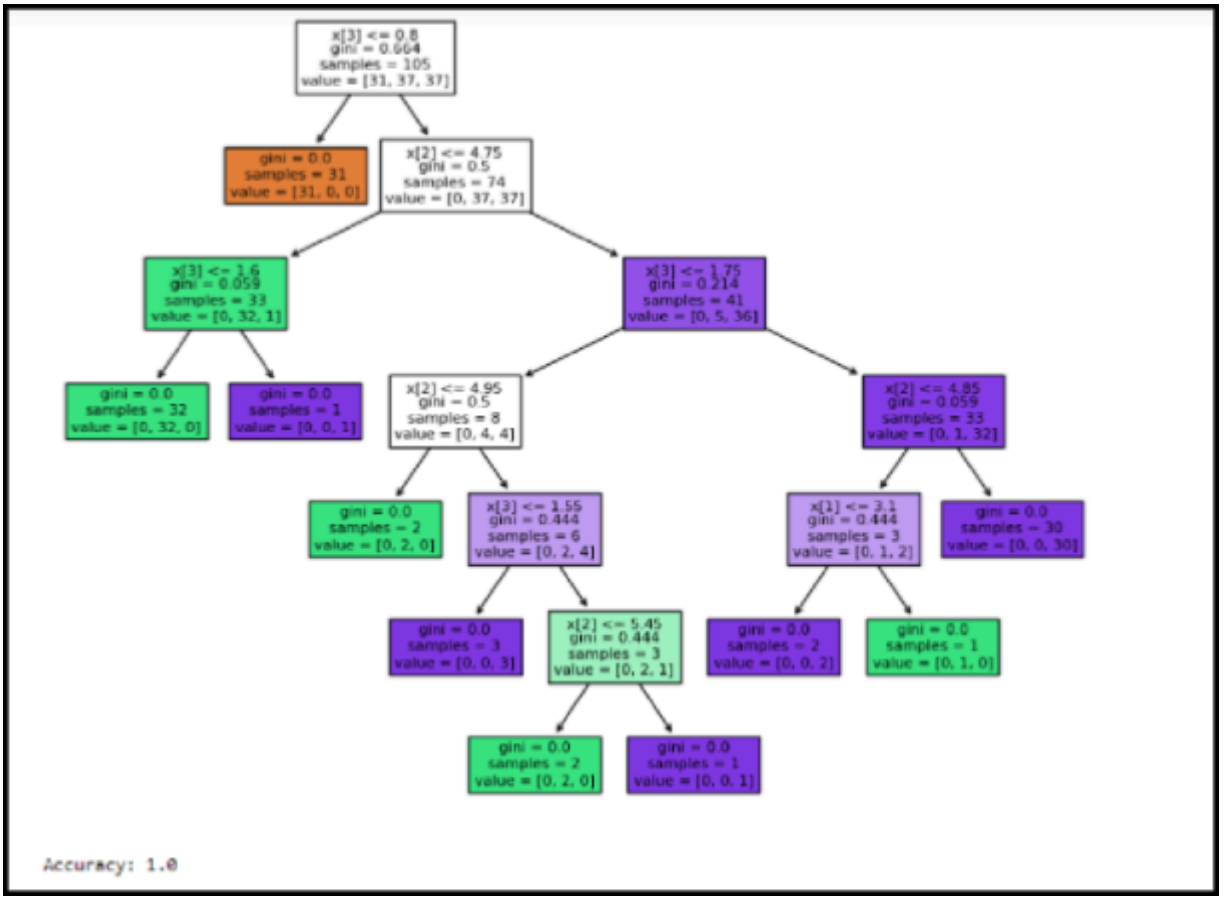


Figure 1: Decision Tree Output

**CONCLUSION**

While decision trees are a powerful tool in machine learning and data analysis, they do have some limitations. Overall, decision trees are a valuable addition to any machine learning or data analysis toolkit, and can be used to make accurate predictions and gain insights into complex datasets.