Practical Assignment No. 5	
Title:	**Decision Tree Algorithm**
Problem Statement:	Using the Car Evaluation Dataset from the Kaggle repository(originally from the UCI Machine Learning Repository), design a decision tree algorithm for classification.
Objective:	To understand application development using techniques of data science and machine learning.
Outcome:	CO606.3: Apply machine learning techniques to develop real world applications.
Software or Hardware Requirements:	Anaconda/Java/GCC
Theory:	**Decision Tree:** Classification is one of the most fundamental tasks in machine learning, where the goal is to assign predefined class labels to input data. A Decision Tree is a supervised learning algorithm that can be used for both classification and regression problems. It works by recursively splitting the dataset into subsets based on the feature values that result in the best separation of classes, forming a tree-like structure of decision nodes and leaf nodes. **Decision Tree Algorithm Overview** A Decision Tree models decisions and their possible outcomes in a tree structure. It consists of: ● Root Node: Represents the entire dataset. ● Decision Nodes: Represent tests on attributes. ● Branches: Represent outcomes of those tests. ● Leaf Nodes: Represent final class labels (predicted output). The algorithm works by partitioning data recursively to maximize purity of classes at each step. **Working Principle of Decision Tree**

1. Select the Best Attribute:
   Choose the feature that best splits the dataset based on a certain criterion (e.g., Information Gain, Gini Index).

2. Split the Dataset:
   Partition the data into subsets where each subset corresponds to a possible value of the chosen attribute.

3. Repeat Recursively:
   Apply the same logic to each subset until:

   - All records belong to the same class, or
   - There are no remaining attributes to split.

4. Label the Leaves:
   Each leaf node represents a class label.

Splitting Criteria

(a) Information Gain (Entropy)

Entropy measures the amount of uncertainty or impurity in a dataset.

$$Entropy(S) = -\sum_{i=1}^{c} p_i \log_2(p_i)$$

Where:

- p_i = proportion of samples in class i
- c = number of classes

Information Gain:

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

Select the attribute with maximum Information Gain for splitting.

(b) Gini Index

Gini Index measures the probability of misclassification.

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

The attribute with the **lowest Gini Index** is chosen for splitting.
Decision Tree Algorithms:
ID3 (Iterative Dichotomiser 3): Uses **Information Gain** as a splitting measure.

C4.5: Extension of ID3, handles both categorical and continuous data, uses **Gain Ratio**.

CART (Classification and Regression Trees): Uses **Gini Index**, supports both classification and regression tasks.

Evaluation Metrics:

To evaluate the performance of the classifier:

- Accuracy: Percentage of correctly classified instances
- Precision, Recall, and F1-Score: Measure performance per class
- Confusion Matrix: Shows distribution of predictions vs. actual classes

Advantages of Decision Tree

- Easy to understand and visualize
- Handles both numerical and categorical data
- Requires little data preprocessing
- Non-parametric (no assumptions about data distribution)

Limitations

- Can easily overfit with complex data
- Sensitive to small data changes
- Works best with smaller, clean datasets

Applications

	• Customer behavior prediction • Credit risk analysis • Medical diagnosis • Product recommendation • Vehicle evaluation and selection
Input/Datasets/Test Cases:	(write details from UCI repository) Dataset- Name of the Dataset: Description of the Dataset: Dataset Characteristics: Subject Area: Associated Tasks: Feature Type: # Instances: # Features:
Results:	Execute code decision tree algorithm for classification. Take a print of this code with output for submission as part of results.
Analysis and conclusion:	Write your own analysis of output and conclusion(Minimum 1 statement of Analysis, Minimum 1 Statement Conclusion)
References:	Reference /Links(min Any 2, include dataset ref.). Write references in IEEE format.