# Experiment 8

## Objective:

To implement decision tree using C4.5 algorithm.

## Theory

The C4.5 algorithm, developed by Ross Quinlan, is an extension of the ID3 algorithm and is widely used for building decision trees, especially for classification tasks. C4.5 improves on ID3 by addressing several of its limitations, making it a more powerful and flexible algorithm. C4.5 can handle both categorical and continuous data, perform automatic pruning to reduce overfitting, and manage missing values effectively.

**Decision Tree Structure in C4.5**: A decision tree consists of nodes (where decisions are made based on features), branches (representing the outcomes of these decisions), and leaf nodes (which provide the final class label). Like ID3, C4.5 uses entropy and information gain to decide on splits but introduces gain ratio to avoid biases toward features with many distinct values.

**The C4.5 Algorithm Process**:

1. **Entropy and Information Gain**: Like ID3, C4.5 uses **entropy** to measure the impurity of a node and **information gain** to evaluate the effectiveness of each feature for splitting the data. However, C4.5 improves upon ID3 by implementing a new metric, called the **gain ratio**, to prevent the algorithm from favoring features with numerous unique values.

   Information gain is calculated as:

$$IG(S, A) = E(S) - \sum_{v \in \text{values}(A)} \frac{|S_v|}{|S|} E(S_v)$$

   where E(S)E(S) is the entropy of the dataset SS, and SvSv represents subsets of SS based on the values of feature AA.

2. **Gain Ratio**: While ID3 selects the feature with the highest information gain for splitting, C4.5 uses **gain ratio** to mitigate the bias toward features with multiple distinct values. Gain ratio is calculated as:

$$\text{Gain Ratio} = \frac{\text{Information Gain}}{\text{Split Information}}$$

   where **split information** measures the potential of each feature to split data into subsets. C4.5 selects the feature with the highest gain ratio, ensuring more balanced splits.

3. **Handling Continuous Data**: C4.5 can handle continuous or numeric data by determining a threshold value for splitting. For each feature, it identifies possible

thresholds and chooses the one that yields the highest gain ratio. This is a significant improvement over ID3, which only supports categorical features.
4. **Handling Missing Values**: C4.5 accommodates missing values by considering the probability of each possible outcome. When splitting on a feature with missing values, C4.5 assigns weights to branches based on the distribution of known values, allowing the algorithm to include data points with missing values rather than discarding them.
5. **Tree Pruning**: C4.5 includes a post-pruning step to reduce overfitting. **Subtree replacement** and **subtree raising** are used to prune branches that do not improve classification accuracy on test data. This step simplifies the model, enhancing its generalizability and interpretability.

**Advantages and Limitations of C4.5**: C4.5 produces interpretable decision trees, handling continuous and categorical data, missing values, and avoiding biases toward features with numerous values. Its automatic pruning reduces the risk of overfitting, making it suitable for a wide range of classification tasks. However, C4.5 is computationally more intensive than ID3, as calculating gain ratio and handling continuous features require additional processing. Additionally, it may not perform optimally on large datasets without optimization.

**Applications of C4.5**:

1. **Medical Diagnosis**: C4.5 helps build decision trees for diagnostic systems, classifying diseases based on symptoms and test results with rules that are easily interpretable.
2. **Customer Segmentation**: In marketing, C4.5 is used to classify customers based on purchasing behavior and demographics, allowing companies to tailor their strategies.
3. **Credit Scoring**: Financial institutions apply C4.5 to assess applicants' creditworthiness based on various financial and demographic features, generating interpretable decision rules.

**Implementing C4.5**: While C4.5 is not directly available in common libraries like `scikit-learn`, similar functionality can be achieved using `DecisionTreeClassifier` with advanced settings or by using the Quinlan-developed C4.5 in older software. Alternatively, tools like WEKA provide a direct implementation of C4.5.

In summary, C4.5 is a powerful, flexible algorithm that addresses several limitations of ID3, producing interpretable trees with improved handling of continuous data and missing values, as well as automatic pruning.

## Result

As a result of this Experiment, we successfully wrote and executed the program to implement decision tree using C4.5 algorithm.

## Learning Outcomes

Understand and implement the C4.5 algorithm to build robust decision trees for classification tasks, with an emphasis on gain ratio and pruning to improve accuracy and interpretability.