

**ID3** stands for **Iterative Dichotomiser 3**.

- **Iterative**: Refers to the repeated process of selecting attributes and splitting the dataset during the construction of the decision tree.
- **Dichotomiser**: Indicates the method of dividing the data into subsets (though not necessarily binary splits, despite the name suggesting "two-way" splits).
- **3**: Signifies that this is the third iteration or version of the algorithm, developed by Ross Quinlan in 1986 as part of his work on decision trees.

ID3 uses information gain to iteratively select the best attribute for splitting the data at each step, building the tree in a top-down, greedy manner.

---

The **main difference between ID3 and C4.5** decision tree algorithms lies in their handling of data and their splitting criteria:

### 1. Handling of Data Types:

- **ID3** works primarily with **categorical data**. It cannot handle continuous (numerical) data directly and requires that continuous data be discretized beforehand.
- **C4.5**, an extension of ID3, can handle both **categorical and continuous data**. It dynamically selects split points for continuous attributes by dividing them into two intervals based on a threshold.

### 2. Splitting Criteria:

- **ID3** uses **Information Gain** to choose the attribute for splitting the data. However, this criterion tends to favor attributes with a large number of distinct values, which can lead to overfitting.
- **C4.5** uses **Gain Ratio**, which is a modification of Information Gain. Gain Ratio penalizes attributes with many distinct values, helping to reduce overfitting.

### 3. Handling of Missing Data:

- **ID3** does not provide a mechanism to handle missing values.
- **C4.5** can handle **missing values**. When a missing value is encountered, C4.5 distributes the instance probabilistically based on the distribution of known values.

### 4. Tree Pruning:

- **ID3** does not include pruning; it grows the tree until it perfectly classifies the training data, which can lead to overfitting.
- **C4.5** includes a **post-pruning** mechanism, which removes branches that add little predictive power, reducing overfitting and improving generalization.

## 5. Output Format:

- **ID3** generates a **non-binary tree**, meaning the nodes can have multiple branches depending on the number of possible values for a given attribute.
- **C4.5** produces a **binary tree** by converting multi-way splits into binary ones if required, further simplifying the tree structure.

In summary, **C4.5** is a more advanced version of **ID3**, with added capabilities such as handling continuous attributes, dealing with missing values, improved splitting criteria, and tree pruning for better generalization.