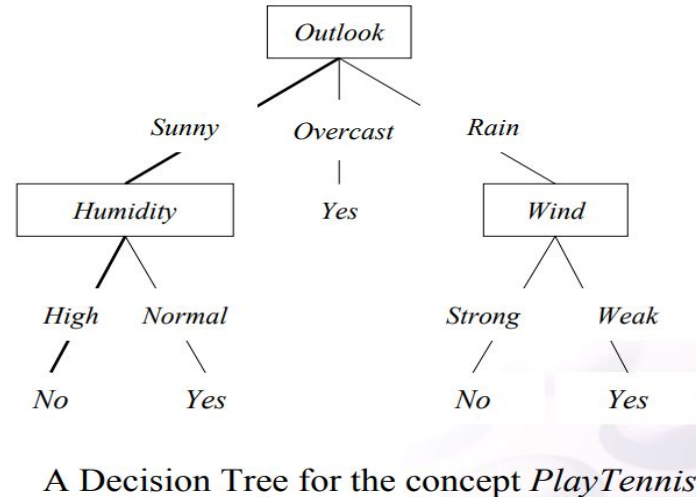
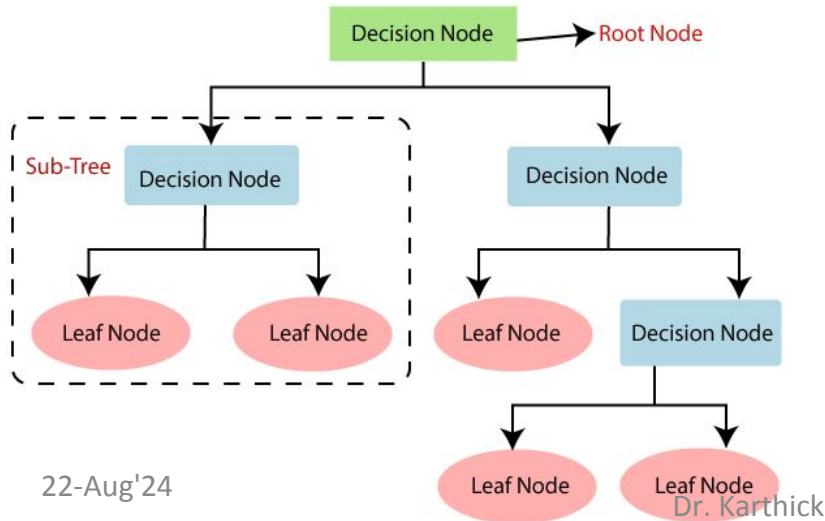
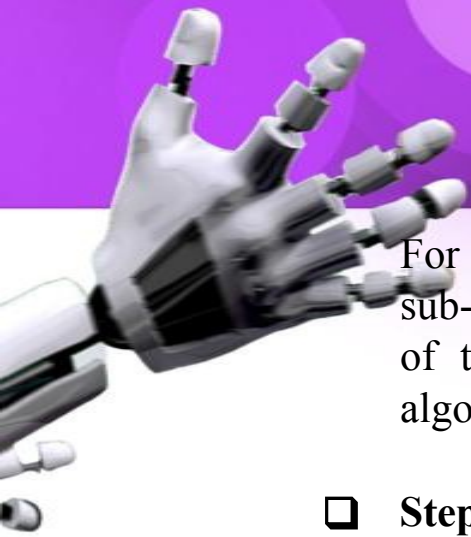


Learning Decision

❑ Decision Tree is a **Supervised learning technique** that can be used for both classification and Regression problems, but mostly it is preferred for solving Classification problems. It is a tree-structured classifier, where **internal nodes represent the features of a dataset, branches represent the decision rules** and **each leaf node represents the outcome**.

- ❑ It is a graphical representation for getting all the possible solutions to a problem/decision based on given conditions.
- ❑ In a Decision tree, there are two nodes, which are the **Decision Node** and **Leaf Node**. Decision nodes are used to make any decision and have multiple branches, whereas Leaf nodes are the output of those decisions and do not contain any further branches.





Inductive inference with Decision Trees

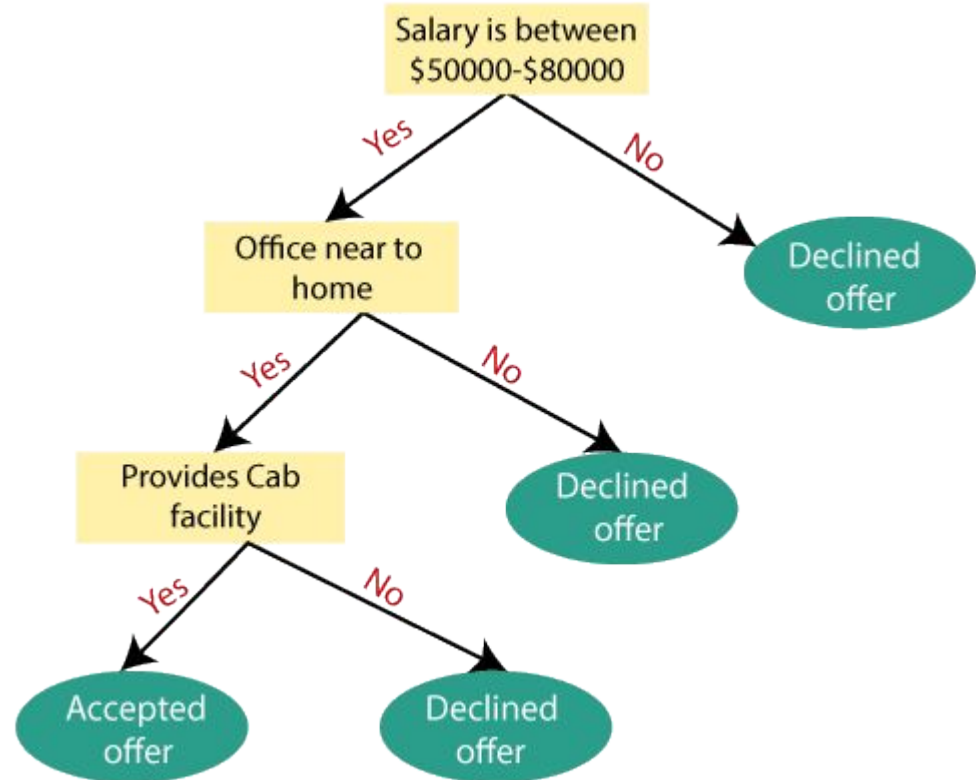
For the next node, the algorithm again compares the attribute value with the other sub-nodes and move further. It continues the process until it reaches the leaf node of the tree. The complete process can be better understood using the below algorithm:

- ❑ **Step-1:** Begin the tree with the root node, says S, which contains the complete dataset.
- ❑ **Step-2:** Find the best attribute in the dataset using **Attribute Selection Measure (ASM)**.
- ❑ **Step-3:** Divide the S into subsets that contains possible values for the best attributes.
- ❑ **Step-4:** Generate the decision tree node, which contains the best attribute.
- ❑ **Step-5:** Recursively make new decision trees using the subsets of the dataset created in step -3. Continue this process until a stage is reached where you cannot further classify the nodes and called the final node as a leaf node.

Learning Decision

Tree

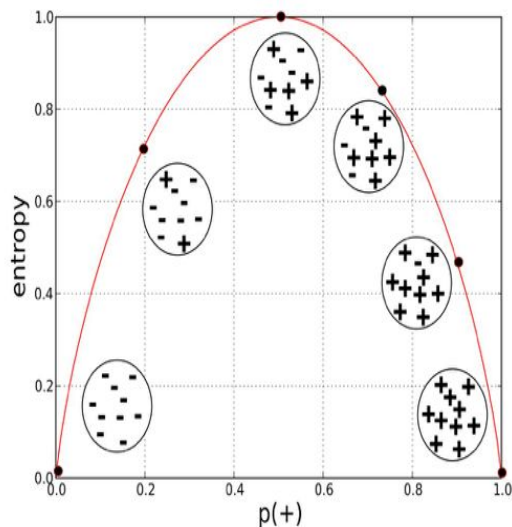
Example: Suppose there is a candidate who has a job offer and wants to decide whether he should accept the offer or Not. So, to solve this problem, the decision tree starts with the root node (Salary attribute by ASM). The root node splits further into the next decision node (distance from the office) and one leaf node based on the corresponding labels. The next decision node further gets split into one decision node (Cab facility) and one leaf node. Finally, the decision node splits into two leaf nodes (Accepted offers and Declined offer). Consider the diagram:





Entropy

- Entropy is a measure of disorder or impurity in the given dataset.
- In the decision tree, messy data are split based on values of the feature vector associated with each data point. With each split, the data becomes more homogenous which will decrease the entropy. However, some data in some nodes will not be homogenous, where the entropy value will not be small. The higher the entropy, the harder it is to draw any conclusion. When the tree finally reaches the terminal or leaf node maximum purity is added.
- **According to information theory** – Optimal length code assigns $-\log_2 p$ bits to message having probability p





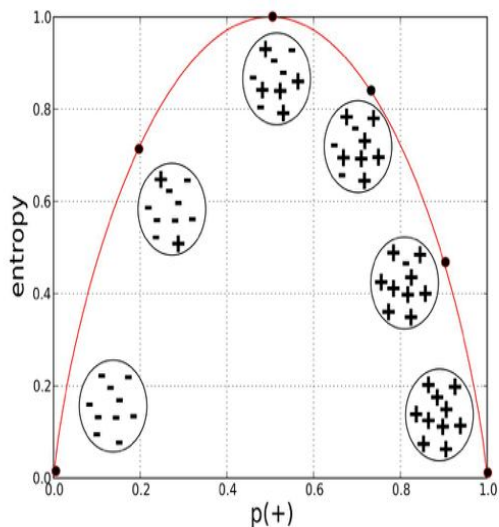
Entropy

- For a dataset that has C classes and the probability of randomly choosing data from class, i is P_i . Then entropy $E(S)$ can be mathematically represented as

$$E(S) = \sum_{i=1} -p_i \log_2 p_i$$

If we have a dataset of 10 observations belonging to two classes YES and NO. If 6 observations belong to the class, YES, and 4 observations belong to class NO, then entropy can be written as below.

$$E(S) = -(P_{yes} \log_2 P_{yes} + P_{no} \log_2 P_{no})$$





Entropy

$$E(S) = -(P_{yes} \log_2 P_{yes} + P_{no} \log_2 P_{no})$$

P_{yes} is the probability of choosing Yes and P_{no} is the probability of choosing a No. Here P_{yes} is 6/10 and P_{no} is 4/10.

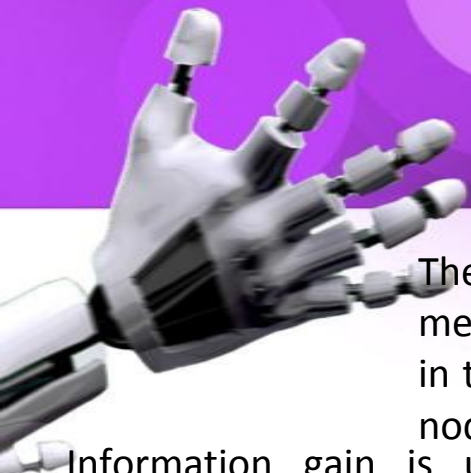
$$E(S) = - (6/10 * \log_2 6/10 + 4/10 * \log_2 4/10) \approx 0.971$$

If all the 10 observations belong to 1 class then entropy will be equal to zero. Which implies the node is a pure node.

$$E(S) = - (1 \log_2 1) = 0$$

If both classes YES and NO have an equal number of observations, then entropy will be equal to 1.

$$E = - \left(\frac{5}{10} * \log_2 \frac{5}{10} + \frac{5}{10} * \log_2 \frac{5}{10} \right) = -2(0.5 \log_2 0.5) = 1$$



Information Gain

The Information Gain measures the expected reduction in entropy. Entropy measures impurity in the data and information gain measures reduction in impurity in the data. The feature which has minimum impurity will be considered as the root node.

Information gain is used to decide which feature to split on at each step in building the tree. The creation of sub-nodes increases the homogeneity, that is decreases the entropy of these nodes.

The more the child node is homogeneous, the more the variance will be decreased after each split. Thus Information Gain is the variance reduction and can calculate by how much the variance decreases after each split.

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

✓ $Values(A)$: the set of all possible values for attribute A

✓ S_v : the subset of S for which attribute A has value v

- First term: the entropy of the original collection
- Second term: the expected value of the entropy after S is partitioned using attribute A



ID3

ID3 stands for **Iterative Dichotomiser 3** and is named such because the algorithm iteratively (repeatedly) dichotomizes(divides) features into two or more groups at each step.

ID3 is an algorithm invented by Ross Quinlan used to generate a decision tree from a dataset and is the most popular algorithms used to constructing trees.

Steps:

- a) Take the Entire dataset as an input.
- b) Calculate the Entropy of the target variable, As well as the predictor attributes
- c) Calculate the information gain of all attributes.
- d) Choose the attribute with the highest information gain as the Root Node
- e) Repeat the same procedure on every branch until the decision node of each branch is finalized.



ID3 Example

Forecast whether the match will be played or not according to the weather condition. Here we can see the table-

Day	Outlook	Temp.	Humidity	Wind	Decision
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

ID3 Example

Day	Outlook	Temp	Humidity	Wind	Play Tennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

Attribute: Outlook

Values (Outlook) = Sunny, Overcast, Rain

$$S = [9+, 5-]$$

$$Entropy(S) = -\frac{9}{14} \log_2 \frac{9}{14} - \frac{5}{14} \log_2 \frac{5}{14} = 0.94$$

$$S_{Sunny} \leftarrow [2+, 3-]$$

$$Entropy(S_{Sunny}) = -\frac{2}{5} \log_2 \frac{2}{5} - \frac{3}{5} \log_2 \frac{3}{5} = 0.971$$

$$S_{Overcast} \leftarrow [4+, 0-]$$

$$Entropy(S_{Overcast}) = -\frac{4}{4} \log_2 \frac{4}{4} - \frac{0}{4} \log_2 \frac{0}{4} = 0$$

$$S_{Rain} \leftarrow [3+, 2-]$$

$$Entropy(S_{Rain}) = -\frac{3}{5} \log_2 \frac{3}{5} - \frac{2}{5} \log_2 \frac{2}{5} = 0.971$$

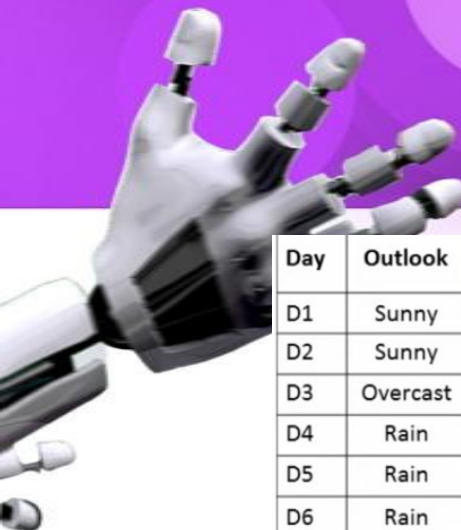
$$Gain(S, Outlook) = Entropy(S) - \sum_{v \in \{Sunny, Overcast, Rain\}} \frac{|S_v|}{|S|} Entropy(S_v)$$

$$Gain(S, Outlook)$$

$$= Entropy(S) - \frac{5}{14} Entropy(S_{Sunny}) - \frac{4}{14} Entropy(S_{Overcast}) - \frac{5}{14} Entropy(S_{Rain})$$

$$Gain(S, Outlook) = 0.94 - \frac{5}{14} 0.971 - \frac{4}{14} 0 - \frac{5}{14} 0.971 = 0.2464$$

ID3 Example



Day	Outlook	Temp	Humidity	Wind	Play Tennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

Attribute: Temp

Values (Temp) = Hot, Mild, Cool

$$S = [9+, 5-]$$

$$Entropy(S) = -\frac{9}{14} \log_2 \frac{9}{14} - \frac{5}{14} \log_2 \frac{5}{14} = 0.94$$

$$S_{Hot} \leftarrow [2+, 2-]$$

$$Entropy(S_{Hot}) = -\frac{2}{4} \log_2 \frac{2}{4} - \frac{2}{4} \log_2 \frac{2}{4} = 1.0$$

$$S_{Mild} \leftarrow [4+, 2-]$$

$$Entropy(S_{Mild}) = -\frac{4}{6} \log_2 \frac{4}{6} - \frac{2}{6} \log_2 \frac{2}{6} = 0.9183$$

$$S_{Cool} \leftarrow [3+, 1-]$$

$$Entropy(S_{Cool}) = -\frac{3}{4} \log_2 \frac{3}{4} - \frac{1}{4} \log_2 \frac{1}{4} = 0.8113$$

$$Gain(S, Temp) = Entropy(S) - \sum_{v \in \{Hot, Mild, Cool\}} \frac{|S_v|}{|S|} Entropy(S_v)$$

$$Gain(S, Temp)$$

$$= Entropy(S) - \frac{4}{14} Entropy(S_{Hot}) - \frac{6}{14} Entropy(S_{Mild}) - \frac{4}{14} Entropy(S_{Cool})$$

$$Gain(S, Temp) = 0.94 - 4/14(1.0) - 6/14(0.9183) - 4/14(0.8113) = 0.0289$$

ID3 Example



Day	Outlook	Temp	Humidity	Wind	Play Tennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

Attribute: Humidity

Values (Humidity) = High, Normal

$$S = [9+, 5-]$$

$$Entropy(S) = -\frac{9}{14} \log_2 \frac{9}{14} - \frac{5}{14} \log_2 \frac{5}{14} = 0.94$$

$$S_{High} \leftarrow [3+, 4-]$$

$$Entropy(S_{High}) = -\frac{3}{7} \log_2 \frac{3}{7} - \frac{4}{7} \log_2 \frac{4}{7} = 0.9852$$

$$S_{Normal} \leftarrow [6+, 1-]$$

$$Entropy(S_{Normal}) = -\frac{6}{7} \log_2 \frac{6}{7} - \frac{1}{7} \log_2 \frac{1}{7} = 0.5916$$


$$Gain(S, Humidity) = Entropy(S) - \sum_{v \in \{High, Normal\}} \frac{|S_v|}{|S|} Entropy(S_v)$$

Gain(S, Humidity)

$$= Entropy(S) - \frac{7}{14} Entropy(S_{High}) - \frac{7}{14} Entropy(S_{Normal})$$

$$Gain(S, Humidity) = 0.94 - \frac{7}{14} 0.9852 - \frac{7}{14} 0.5916 = 0.1516$$

ID3 Example



Day	Outlook	Temp	Humidity	Wind	Play Tennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

Attribute: Wind

Values (Wind) = Strong, Weak

$$S = [9+, 5-]$$

$$Entropy(S) = -\frac{9}{14} \log_2 \frac{9}{14} - \frac{5}{14} \log_2 \frac{5}{14} = 0.94$$

$$S_{Strong} \leftarrow [3+, 3-]$$

$$Entropy(S_{Strong}) = 1.0$$

$$S_{Weak} \leftarrow [6+, 2-]$$

$$Entropy(S_{Weak}) = -\frac{6}{8} \log_2 \frac{6}{8} - \frac{2}{8} \log_2 \frac{2}{8} = 0.8113$$

$$Gain(S, Wind) = Entropy(S) - \sum_{v \in \{Strong, Weak\}} \frac{|S_v|}{|S|} Entropy(S_v)$$

$$Gain(S, Wind) = Entropy(S) - \frac{6}{14} Entropy(S_{Strong}) - \frac{8}{14} Entropy(S_{Weak})$$

$$Gain(S, Wind) = 0.94 - \frac{6}{14} 1.0 - \frac{8}{14} 0.8113 = 0.0478$$

ID3 Example

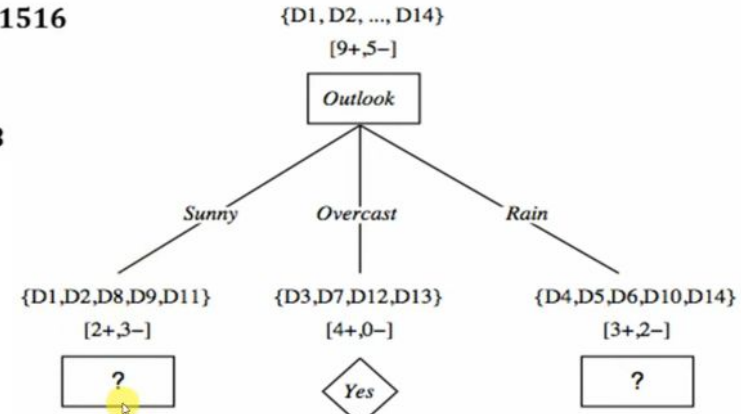
Day	Outlook	Temp	Humidity	Wind	Play Tennis
D1	Sunny	Hot	High	Weak	No
D2	Sunny	Hot	High	Strong	No
D3	Overcast	Hot	High	Weak	Yes
D4	Rain	Mild	High	Weak	Yes
D5	Rain	Cool	Normal	Weak	Yes
D6	Rain	Cool	Normal	Strong	No
D7	Overcast	Cool	Normal	Strong	Yes
D8	Sunny	Mild	High	Weak	No
D9	Sunny	Cool	Normal	Weak	Yes
D10	Rain	Mild	Normal	Weak	Yes
D11	Sunny	Mild	Normal	Strong	Yes
D12	Overcast	Mild	High	Strong	Yes
D13	Overcast	Hot	Normal	Weak	Yes
D14	Rain	Mild	High	Strong	No

$$\text{Gain}(S, \text{Outlook}) = 0.2464$$

$$\text{Gain}(S, \text{Temp}) = 0.0289$$

$$\text{Gain}(S, \text{Humidity}) = 0.1516$$

$$\text{Gain}(S, \text{Wind}) = 0.0478$$



ID3 Example



Day	Temp	Humidity	Wind	Play Tennis
D1	Hot	High	Weak	No
D2	Hot	High	Strong	No
D8	Mild	High	Weak	No
D9	Cool	Normal	Weak	Yes
D11	Mild	Normal	Strong	Yes

Attribute: Temp

Values (Temp) = Hot, Mild, Cool

$S_{Sunny} = [2+, 3-]$

$$Entropy(S_{Sunny}) = -\frac{2}{5}\log_2\frac{2}{5} - \frac{3}{5}\log_2\frac{3}{5} = 0.97$$

$S_{Hot} \leftarrow [0+, 2-]$

$$Entropy(S_{Hot}) = 0.0$$

$S_{Mild} \leftarrow [1+, 1-]$

$$Entropy(S_{Mild}) = 1.0$$

$S_{Cool} \leftarrow [1+, 0-]$

$$Entropy(S_{Cool}) = 0.0$$

$$Gain(S_{Sunny}, Temp) = Entropy(S) - \sum_{v \in \{Hot, Mild, Cool\}} \frac{|S_v|}{|S|} Entropy(S_v)$$

$$Gain(S_{Sunny}, Temp)$$

$$= Entropy(S) - \frac{2}{5} Entropy(S_{Hot}) - \frac{2}{5} Entropy(S_{Mild})$$

$$- \frac{1}{5} Entropy(S_{Cool})$$

$$Gain(S_{Sunny}, Temp) = 0.97 - \frac{2}{5} 0.0 - \frac{2}{5} 1 - \frac{1}{5} 0.0 = 0.570$$

ID3 Example



Day	Temp	Humidity	Wind	Play Tennis
D1	Hot	High	Weak	No
D2	Hot	High	Strong	No
D8	Mild	High	Weak	No
D9	Cool	Normal	Weak	Yes
D11	Mild	Normal	Strong	Yes

Attribute: Humidity

Values (Humidity) = High, Normal

$$S_{Sunny} = [2+, 3-]$$

$$Entropy(S) = -\frac{2}{5} \log_2 \frac{2}{5} - \frac{3}{5} \log_2 \frac{3}{5} = 0.97$$

$$S_{High} \leftarrow [0+, 3-]$$

$$Entropy(S_{High}) = 0.0$$

$$S_{Normal} \leftarrow [2+, 0-]$$

$$Entropy(S_{Normal}) = 0.0$$

$$Gain(S_{Sunny}, Humidity) = Entropy(S) - \sum_{v \in \{High, Normal\}} \frac{|S_v|}{|S|} Entropy(S_v)$$

$$Gain(S_{Sunny}, Humidity) = Entropy(S) - \frac{3}{5} Entropy(S_{High}) - \frac{2}{5} Entropy(S_{Normal})$$

$$Gain(S_{Sunny}, Humidity) = 0.97 - \frac{3}{5} 0.0 - \frac{2}{5} 0.0 = 0.97$$

ID3 Example



Day	Temp	Humidity	Wind	Play Tennis
D1	Hot	High	Weak	No
D2	Hot	High	Strong	No
D8	Mild	High	Weak	No
D9	Cool	Normal	Weak	Yes
D11	Mild	Normal	Strong	Yes

Attribute: Wind

Values (Wind) = Strong, Weak

$$S_{Sunny} = [2+, 3-]$$

$$Entropy(S) = -\frac{2}{5} \log_2 \frac{2}{5} - \frac{3}{5} \log_2 \frac{3}{5} = 0.97$$

$$S_{Strong} \leftarrow [1+, 1-]$$

$$Entropy(S_{Strong}) = 1.0$$

$$S_{Weak} \leftarrow [1+, 2-]$$

$$Entropy(S_{Weak}) = -\frac{1}{3} \log_2 \frac{1}{3} - \frac{2}{3} \log_2 \frac{2}{3} = 0.9183$$

$$Gain(S_{Sunny}, Wind) = Entropy(S) - \sum_{v \in \{Strong, Weak\}} \frac{|S_v|}{|S|} Entropy(S_v)$$

$$Gain(S_{Sunny}, Wind) = Entropy(S) - \frac{2}{5} Entropy(S_{Strong}) - \frac{3}{5} Entropy(S_{Weak})$$

$$Gain(S_{Sunny}, Wind) = 0.97 - \frac{2}{5} 1.0 - \frac{3}{5} 0.918 = 0.0192$$

ID3 Example

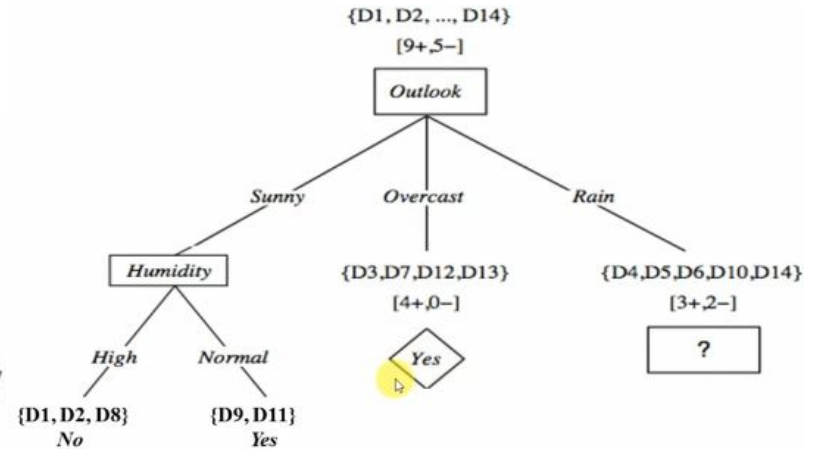


Day	Temp	Humidity	Wind	Play Tennis
D1	Hot	High	Weak	No
D2	Hot	High	Strong	No
D8	Mild	High	Weak	No
D9	Cool	Normal	Weak	Yes
D11	Mild	Normal	Strong	Yes

$$\text{Gain}(S_{\text{sunny}}, \text{Temp}) = 0.570$$

$$\text{Gain}(S_{\text{sunny}}, \text{Humidity}) = 0.97$$

$$\text{Gain}(S_{\text{sunny}}, \text{Wind}) = 0.0192$$



ID3 Example

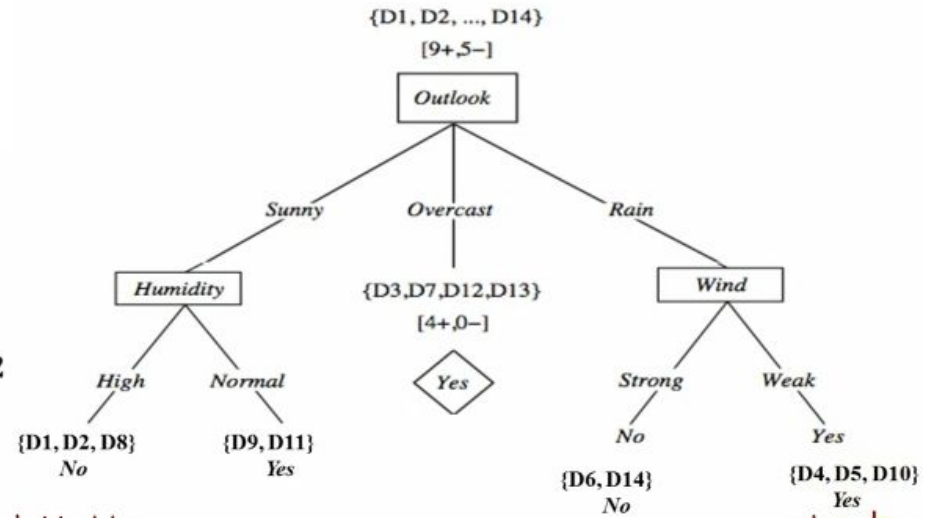


Day	Temp	Humidity	Wind	Play Tennis
D4	Mild	High	Weak	Yes
D5	Cool	Normal	Weak	Yes
D6	Cool	Normal	Strong	No
D10	Mild	Normal	Weak	Yes
D14	Mild	High	Strong	No

$$\text{Gain}(S_{\text{Rain}}, \text{Temp}) = 0.0192$$

$$\text{Gain}(S_{\text{Rain}}, \text{Humidity}) = 0.0192$$

$$\text{Gain}(S_{\text{Rain}}, \text{Wind}) = 0.97$$





Learning Decision

Tree

Advantages of the Decision Tree

- It is simple to understand as it follows the same process which a human follow while making any decision in real-life.
- It can be very useful for solving decision-related problems.
- It helps to think about all the possible outcomes for a problem.
- There is less requirement of data cleaning compared to other algorithms.

Disadvantages of the Decision Tree

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



Issues in Decision Tree Learning

- **Determining how deeply to grow the decision tree,**
- **Handling continuous attributes,**
- **Choosing an appropriate attribute selection measure,**
- **Handling training data with missing attribute values,**
- **Handling attributes with differing costs**



Random Forest Algorithm

- Random Forest algorithm is a powerful tree learning technique in Machine Learning. It works by creating a number of Decision Trees during the training phase. Each tree is constructed using a random subset of the data set to measure a random subset of features in each partition.
- This randomness introduces variability among individual trees, reducing the risk of overfitting and improving overall prediction performance.

Random Forest Algorithm

In prediction, the algorithm aggregates the results of all trees, either by voting (for classification tasks) or by averaging (for regression tasks). This collaborative decision-making process, supported by multiple trees with their insights, provides an example of stable and precise results. Random forests are widely used for classification and regression functions, which are known for their ability to handle complex data, reduce overfitting, and provide reliable forecasts in different environments.

