Decision Tree

Presented by : Muhammad Zaqeem

Overview

Decision Tree

Structure of Decision Tree

Entropy

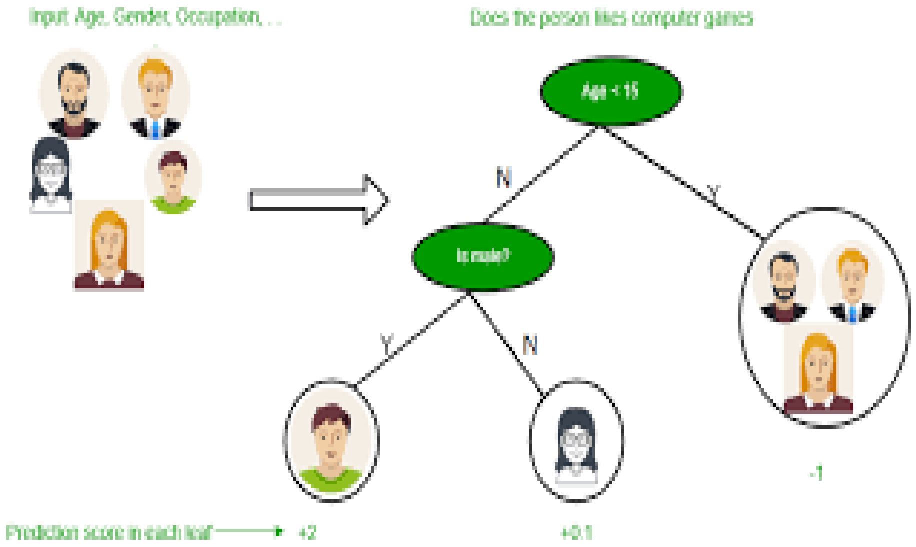
Information Gain

Building Decision Tree

Pruning

Decision Tree

- Decision trees are a popular supervised **machine learning algorithm** that can be used for both regression and classification tasks.
- **Decision Trees (DTs)** are a non-parametric supervised learning method
- It’s called a "tree" because it has a structure similar to a tree in nature, with **branches** and **nodes** leading to different outcomes
- Starting at a **root node**, it uses decision rules to split data at each **internal node** until reaching **leaf nodes** that provide the final prediction


Input: Age, Gender, Occupation, ...
Does the person likes computer games
Age < 15
N
Y
Is male?
Y
N
-1
Prediction score in each leaf
+2
+0.1

Structure of Decision Tree:

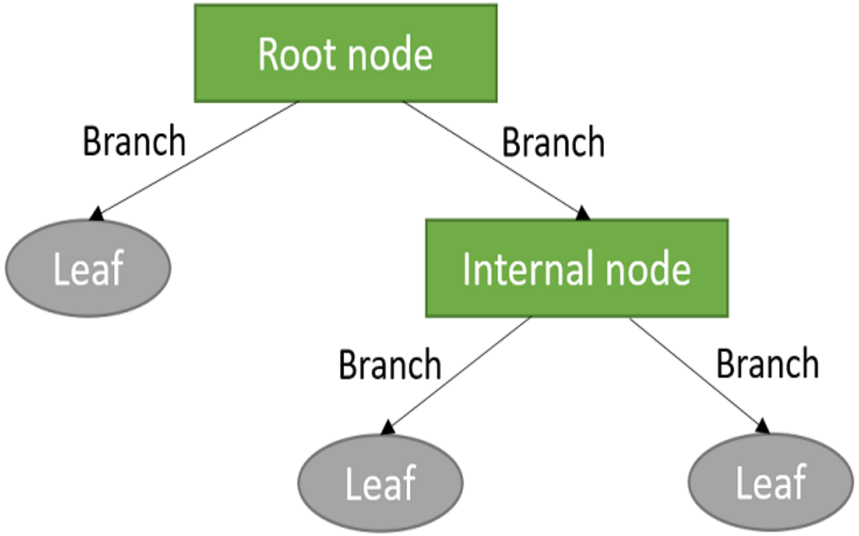
The structure of a **Decision Tree** resembles a flowchart, with three main types of nodes

Root Node: The topmost node in a decision tree. It represents the entire dataset, which is then split into two or more homogeneous sets.

Internal Nodes: Nodes that represent the features (attributes) tested to make decisions.

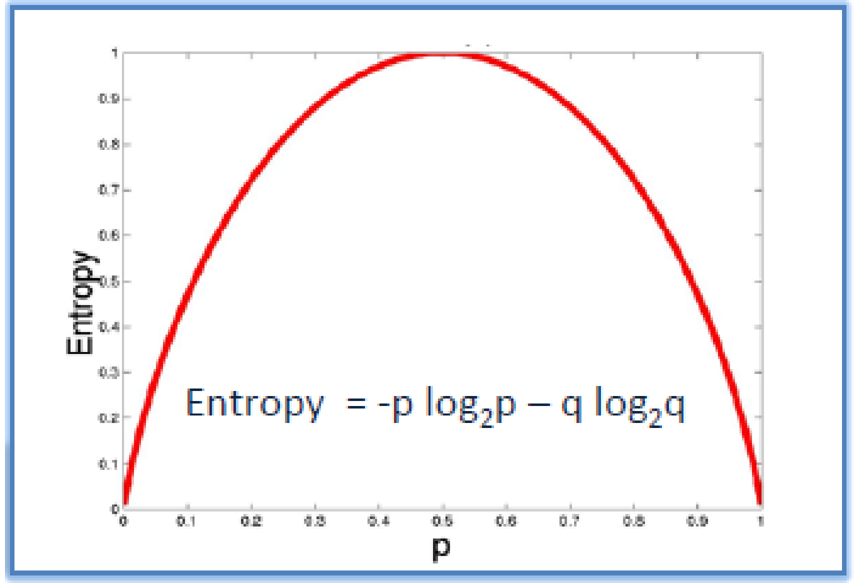
Leaf Nodes (Terminal Nodes): Nodes that represent the final decision or classification. These nodes do not split further.

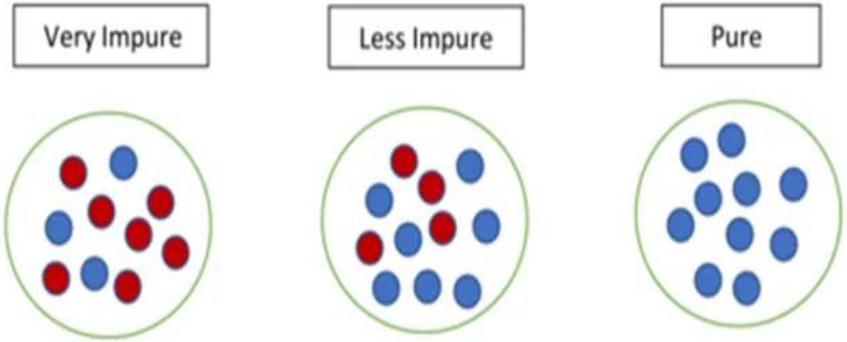
Branches (Edges): To connect nodes.

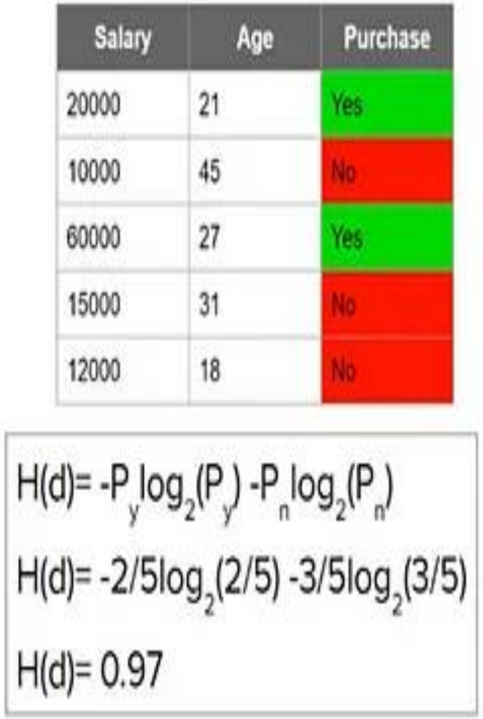


Entropy:

- Entropy is a measure of disorder or impurity in the given dataset.
- high entropy implies a higher level of disorder or uncertainty in the data. It suggests that the data is more heterogeneous, making it challenging for models to make accurate predictions.
- 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
- It helps determine the best split for building an informative decision tree model.





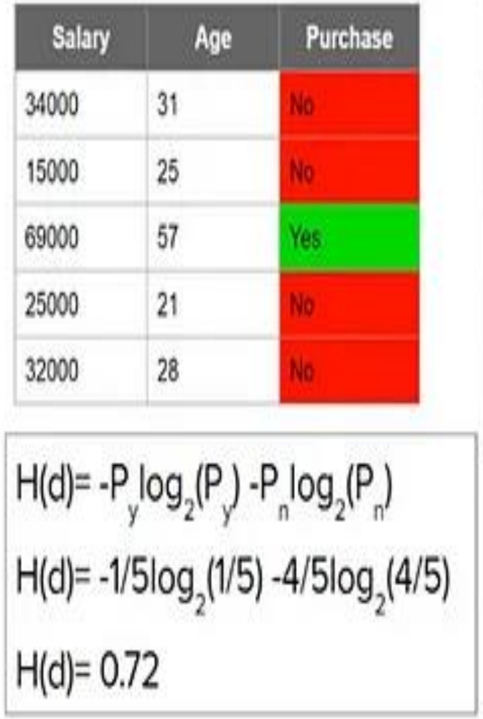


Salary	Age	Purchase
20000	21	Yes
10000	45	No
60000	27	Yes
15000	31	No
12000	18	No

$H(d) = -P_y \log_2(P_y) - P_n \log_2(P_n)$

$H(d) = -2/5 \log_2(2/5) - 3/5 \log_2(3/5)$

$H(d) = 0.97$

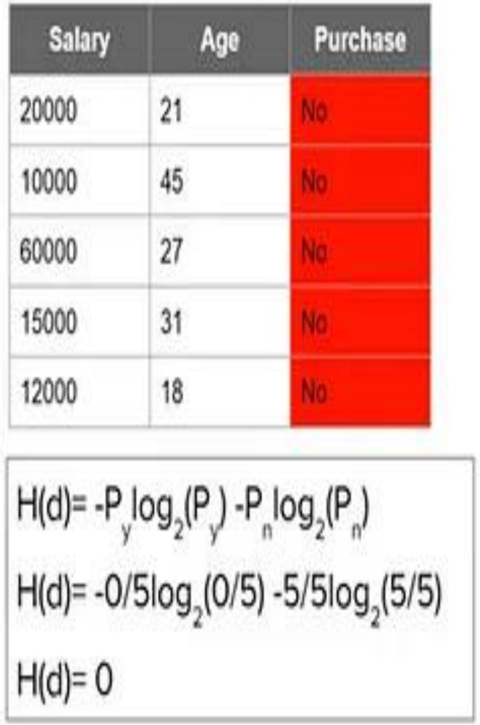


Salary	Age	Purchase
34000	31	No
15000	25	No
69000	57	Yes
25000	21	No
32000	28	No

$H(d) = -P_y \log_2(P_y) - P_n \log_2(P_n)$

$H(d) = -1/5 \log_2(1/5) - 4/5 \log_2(4/5)$

$H(d) = 0.72$



Salary	Age	Purchase
20000	21	No
10000	45	No
60000	27	No
15000	31	No
12000	18	No

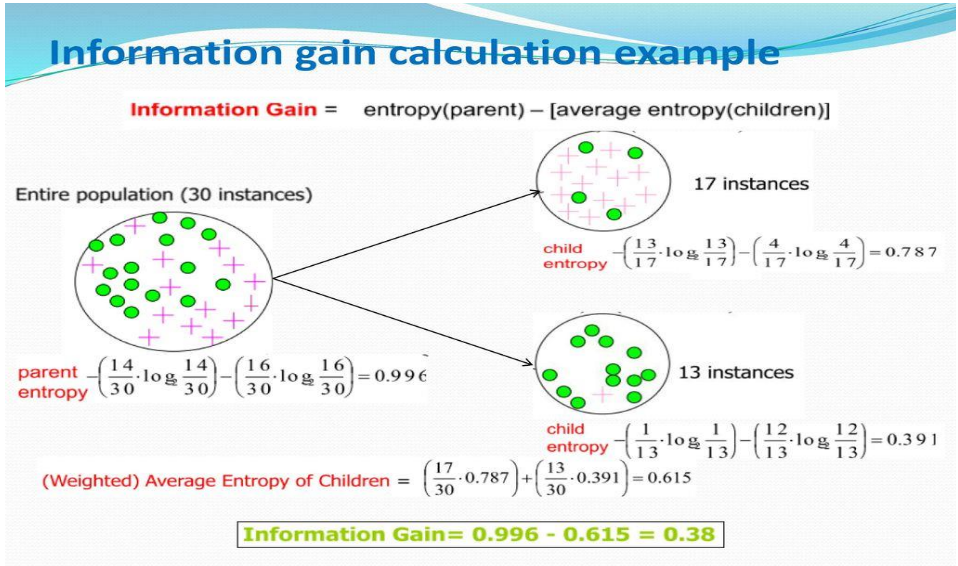
$H(d) = -P_y \log_2(P_y) - P_n \log_2(P_n)$

$H(d) = -0/5 \log_2(0/5) - 5/5 \log_2(5/5)$

$H(d) = 0$

Information Gain:

- The Information Gain measures the expected reduction in entropy.
- 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.
- Information gain of a parent node can be calculated as the entropy of the parent node subtracted entropy of the weighted average of the child node



Information gain calculation example

Information Gain = entropy(parent) – [average entropy(children)]

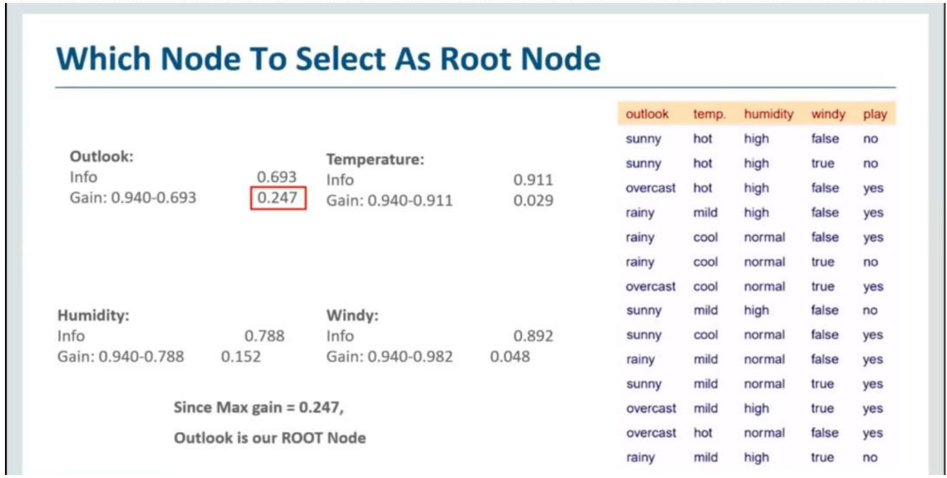
parent entropy $-\left(\frac{14}{30}\cdot\log\frac{14}{30}\right)-\left(\frac{16}{30}\cdot\log\frac{16}{30}\right)=0.996$

child entropy $-\left(\frac{13}{17}\cdot\log\frac{13}{17}\right)-\left(\frac{4}{17}\cdot\log\frac{4}{17}\right)=0.787$

child entropy $-\left(\frac{1}{13}\cdot\log\frac{1}{13}\right)-\left(\frac{12}{13}\cdot\log\frac{12}{13}\right)=0.391$

(Weighted) Average Entropy of Children = $\left(\frac{17}{30}\cdot 0.787\right)+\left(\frac{13}{30}\cdot 0.391\right)=0.615$

$\boxed{\text{Information Gain}= 0.996 - 0.615 = 0.38}$



Which Node To Select As Root Node

Outlook:
Info 0.693
Gain: 0.940-0.693 $\boxed{0.247}$

Temperature:
Info 0.911
Gain: 0.940-0.911 0.029

Humidity:
Info 0.788
Gain: 0.940-0.788 0.152

Windy:
Info 0.892
Gain: 0.940-0.982 0.048

Since Max gain = 0.247,

Outlook is our ROOT Node

outlook	temp.	humidity	windy	play
sunny	hot	high	false	no
sunny	hot	high	true	no
overcast	hot	high	false	yes
rainy	mild	high	false	yes
rainy	cool	normal	false	yes
rainy	cool	normal	true	no
overcast	cool	normal	true	yes
sunny	mild	high	false	no
sunny	cool	normal	false	yes
rainy	mild	normal	false	yes
sunny	mild	normal	true	yes
overcast	mild	high	true	yes
overcast	hot	normal	false	yes
rainy	mild	high	true	no

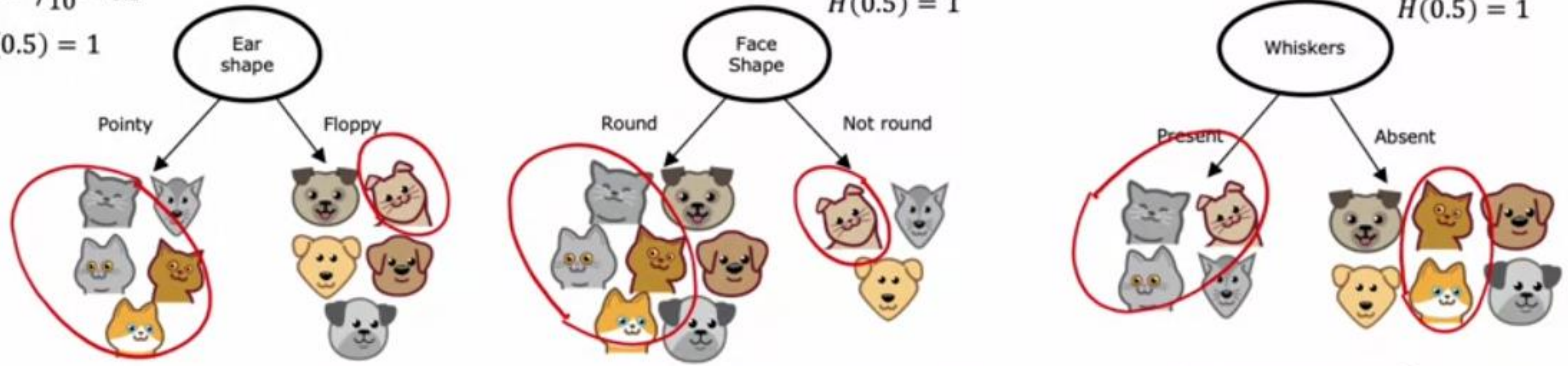
Choosing a split

$p_1 = {}^5/_{10} = 0.5$

$H(0.5) = 1$

$H(0.5) = 1$

$H(0.5) = 1$



$p_1 = {}^4/_5 = 0.8$ $p_1 = {}^1/_5 = 0.2$

$H(0.8) = 0.72$ $H(0.2) = 0.72$

$$H(0.5) - \left(\frac{5}{10}H(0.8) + \frac{5}{10}H(0.2)\right)$$

$= \boxed{0.28}$

$p_1 = {}^4/_7 = 0.57$ $p_1 = {}^1/_3 = 0.33$

$H(0.57) = 0.99$ $H(0.33) = 0.92$

$$H(0.5) - \left(\frac{7}{10}H(0.57) + \frac{3}{10}H(0.33)\right)$$

$= \boxed{0.03}$

$p_1 = {}^3/_4 = 0.75$ $p_1 = {}^2/_6 = 0.33$

$H(0.75) = 0.81$ $H(0.33) = 0.92$

$$H(0.5) - \left(\frac{4}{10}H(0.75) + \frac{6}{10}H(0.33)\right)$$

$= \boxed{0.12}$

Information gain

Building Decision Tree:

Select the Best Feature: Start by choosing the feature that best separates or classifies the data. we choose that feature which have high information gain

Split the Data: Divide the dataset based on the chosen feature's possible values, creating branches that represent the different outcomes of the feature.

Repeat for Each Branch: For each branch created, repeat the process by selecting the best feature to split on, further dividing the data until the data within each branch is as pure as possible (ideally, only one class remains).

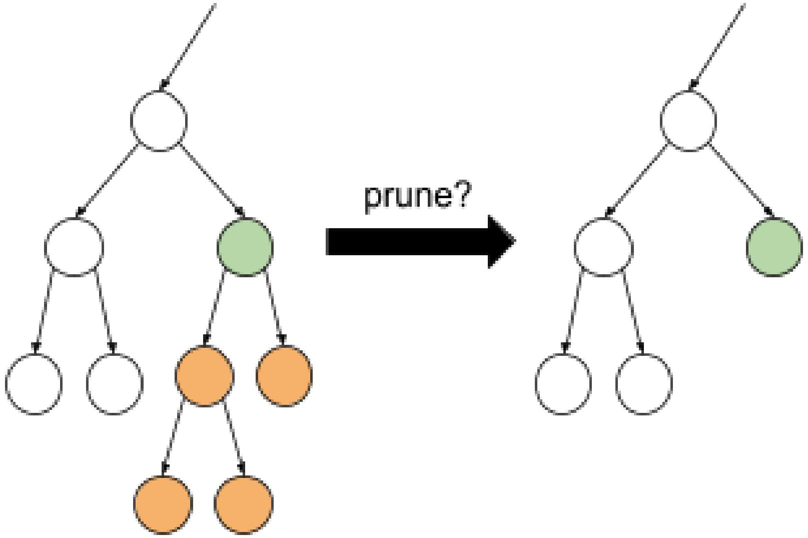
Stop Splitting: Stop dividing branches when one of the stopping conditions is met—such as reaching maximum tree depth, having too few data points in a branch, or achieving zero impurity within a branch. The final branches are called **leaf nodes**.

Pruning:

- Pruning is a technique used to prevent decision trees from overfitting the training data
- Pruning aims to simplify the decision tree by removing parts of it that do not provide significant predictive power, thus improving its ability to generalize to new data.

Pre-Pruning: In **pre-pruning**, we stop the tree from growing any further once a certain condition is met, even before the tree reaches its full depth.

Post-Pruning (Pruning After Training): In **post-pruning**, we first build the tree to its maximum depth and then prune it back.



Thank You