Online Student Training for "Artificial Intelligence & Machine Learning"

(4th Feb, 2021 –17th Mar, 2021)

Decision Tree Classifier

Faculty Trainer

Naw Varsha Pipada

Department of Computer Science & Engineering

Engineering College Bikaner

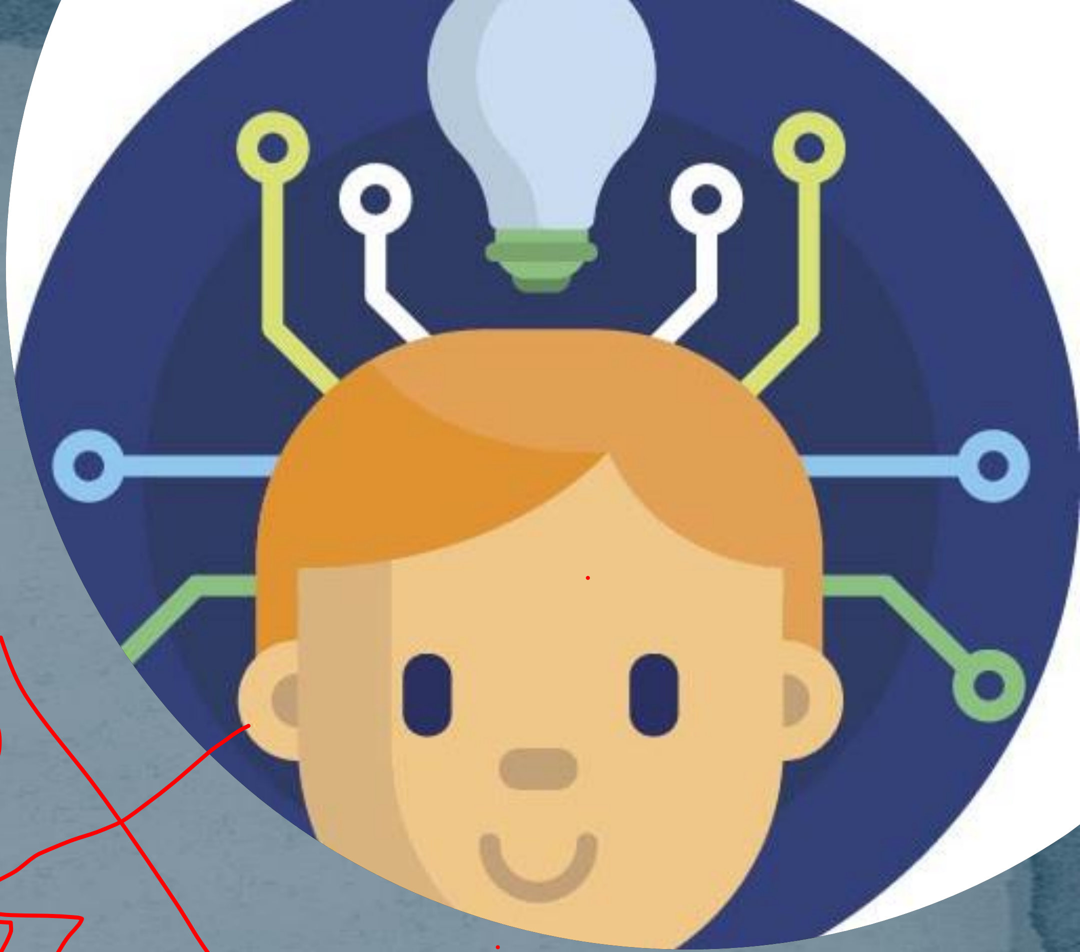
Contents

Guess the type of problem

- Approve or reject the loan ?
- Whether the car will give high mileage or low mileage?
- Should I play Tennis today or not?

>50

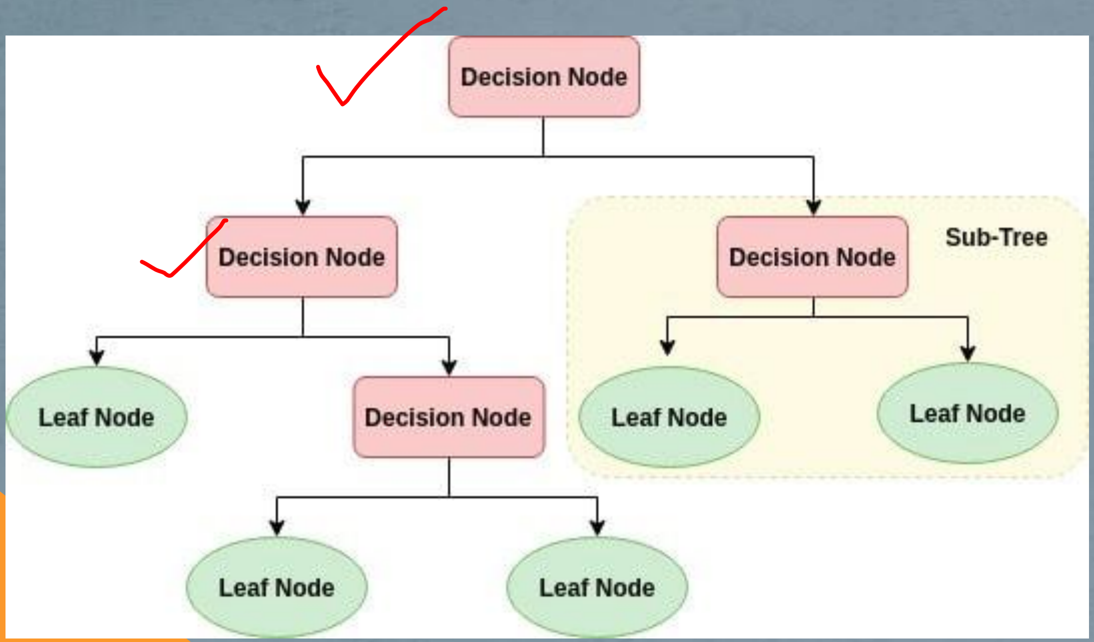
50>



Decision Tree

- Decision tree is one of the method to find solution to classification problem.
- A decision tree is a flowchart-like tree structure where an internal node represents feature(or attribute), the branch represents a decision rule, and each leaf node represents the outcome.
- The topmost node in a decision tree is known as the root node.
- It learns to partition on the basis of the attribute value using recursive partitioning. This flowchart-like structure helps you in decision making

Decision Tree

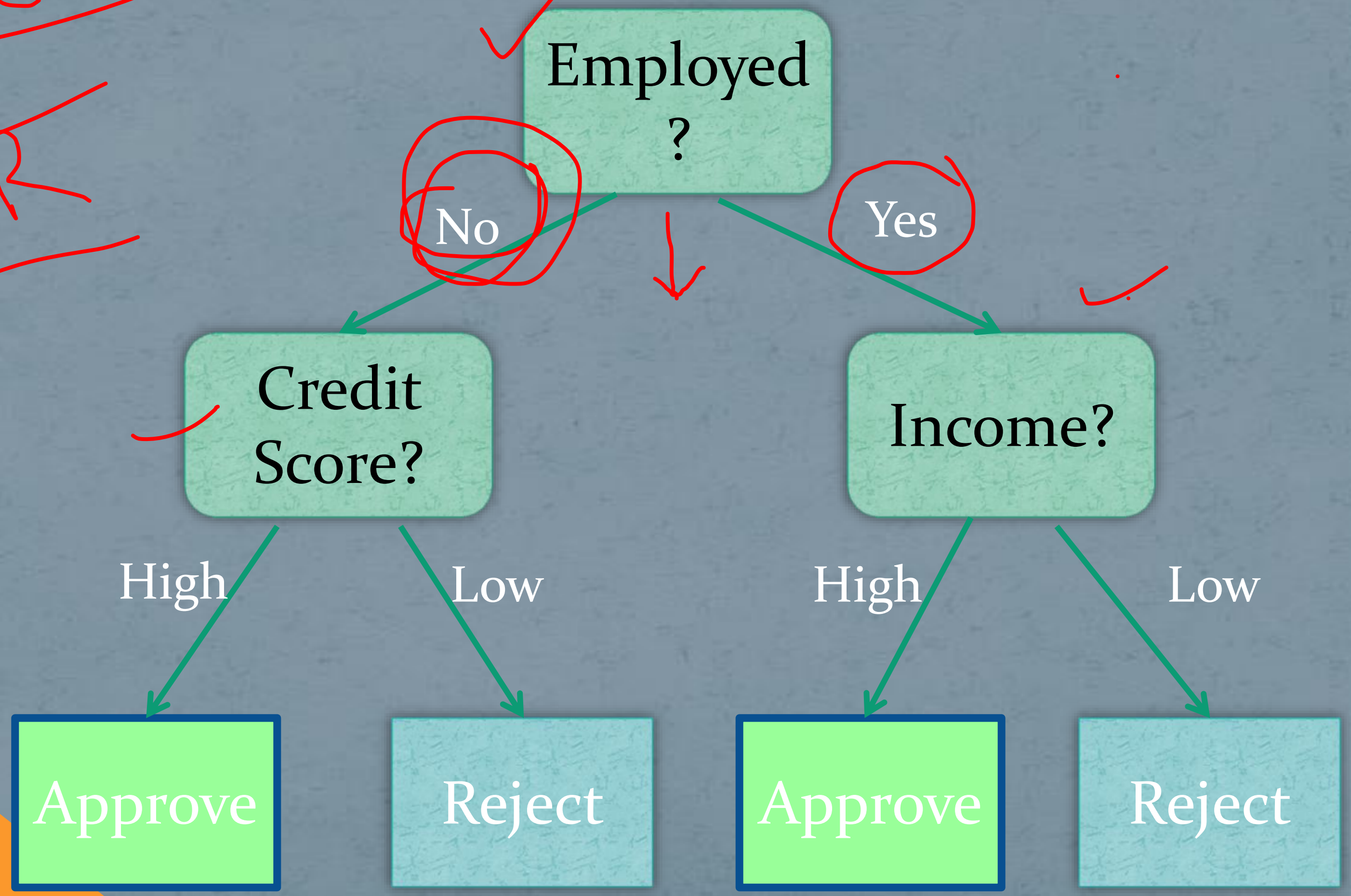


Decision node: Specifies a choice or test of some attribute, with one branch for each outcome

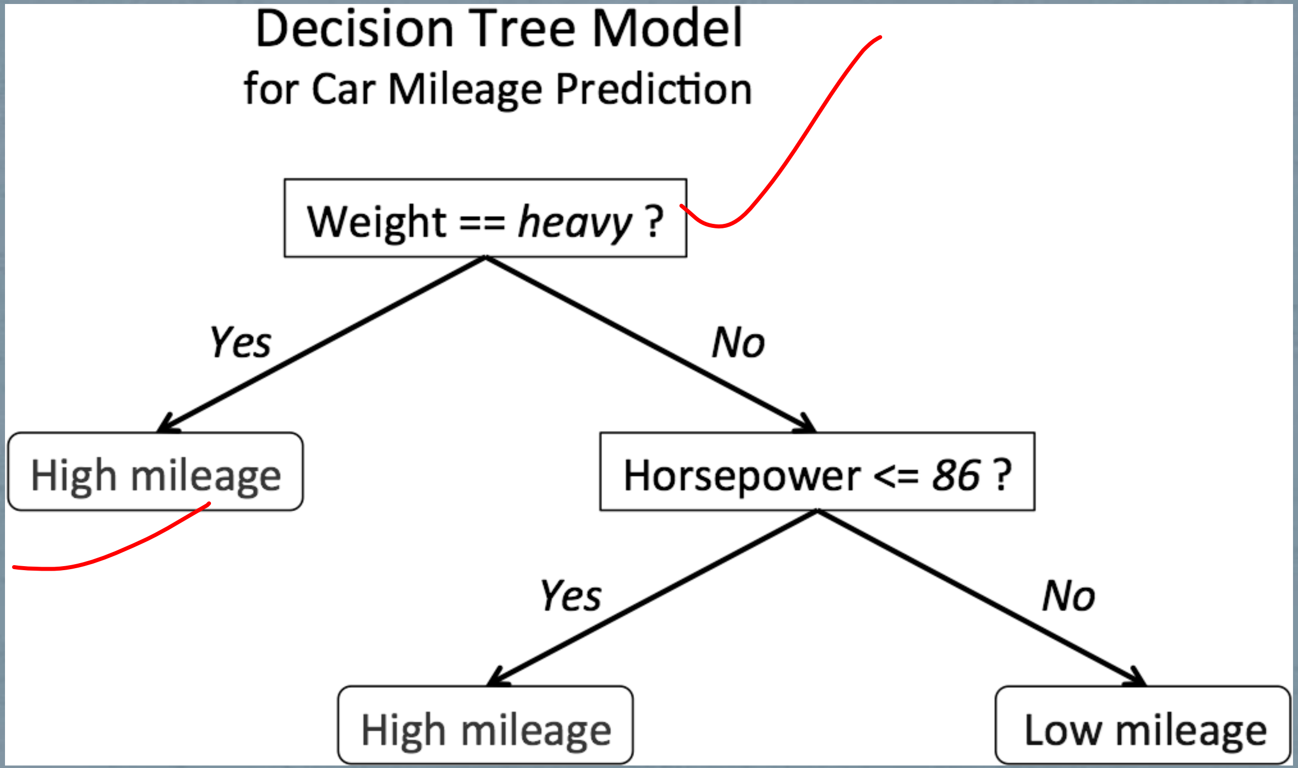
Leaf node: Indicates classification of an example

Example : Whether to approve a loan?

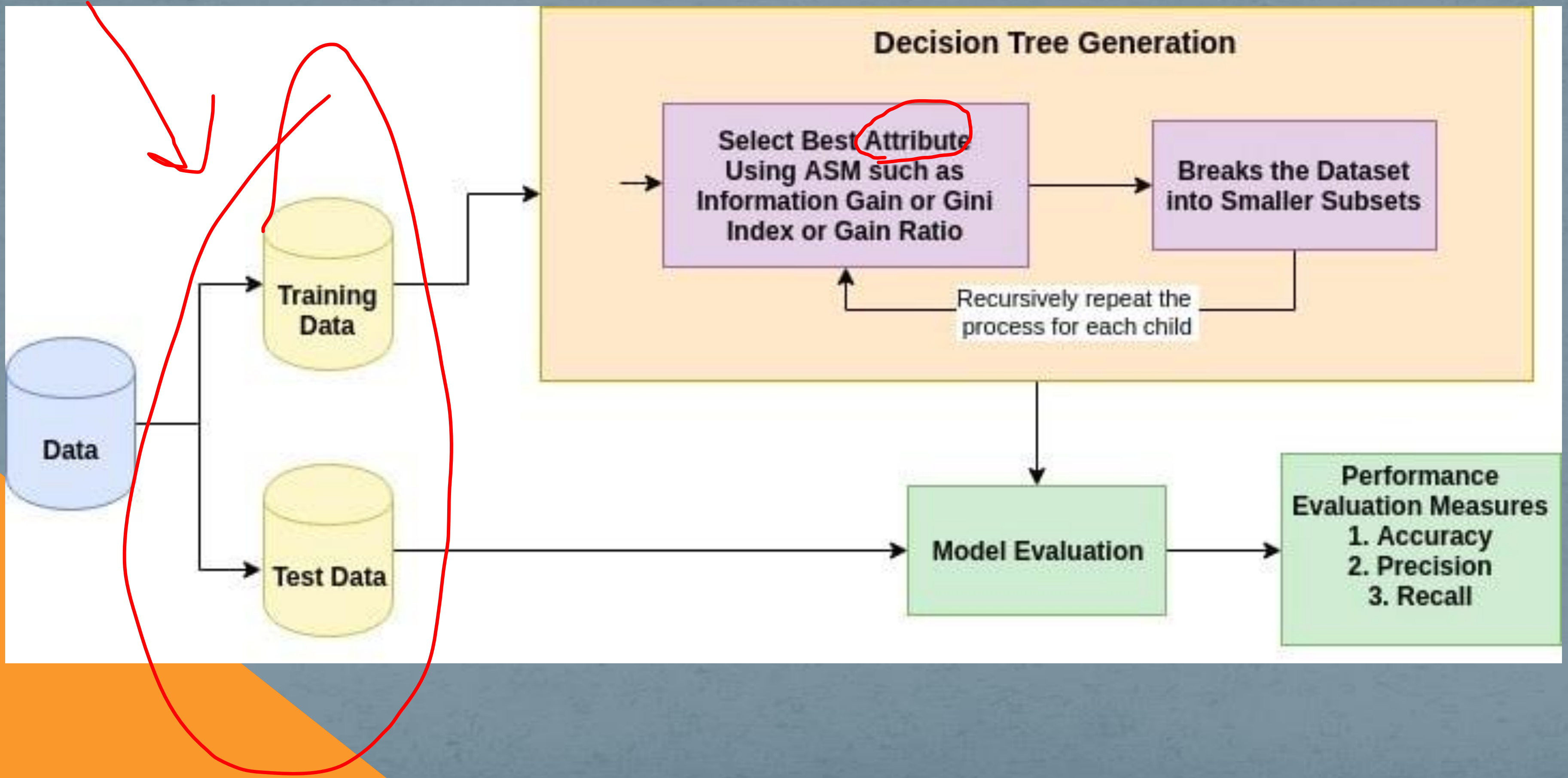
Em	cs	Income	Dec
Yes			A
No			R



Example: Predicting Car Mileage (High/Low)



How it Works



Decision Tree Generation
Select Best Attribute Using ASM such as Information Gain or Gini Index or Gain Ratio
Breaks the Dataset into Smaller Subsets
Recursively repeat the process for each child
Training Data
Data
Test Data
Model Evaluation
Performance Evaluation Measures
1. Accuracy
2. Precision
3. Recall

Learning with Decision Tree Classifier

- To predict the output of classification problem
  - Generate the Decision Tree from Training Set
  - Predict the output of test data with the decision tree formed.
  - Use the metrics (such as accuracy , precision, recall) to evaluate model
  - Tune hyper parameters if required.

How it Works: To generate Decision Tree

1. Select the best attribute using Attribute Selection Measures(ASM) to split the records.
2. Make that attribute a decision node and breaks the dataset into smaller subsets.
3. Starts tree building by repeating this process recursively for each child until one of the condition will match:
   - All the tuples belong to the same attribute value.
   - There are no more remaining attributes.
   - There are no more instances.

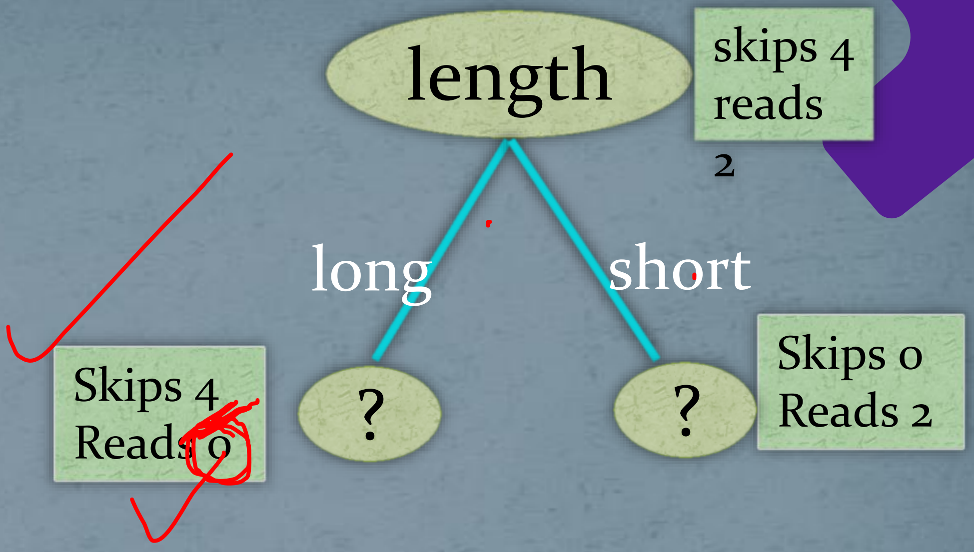
Example

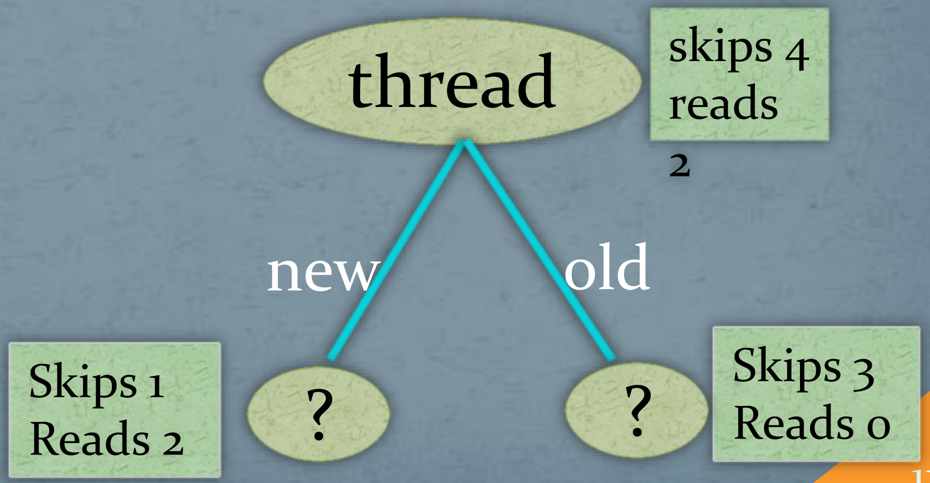
Training Examples:

	Action	Author	Thread	Length	Where
e1	skips	known	new	long	Home
e2	reads	unknown	new	short	Work
e3	skips	unknown	old	long	Work
e4	skips	known	old	long	home
e5	reads	known	new	short	home
e6	skips	known	old	long	work

New Examples:

e7	???	known	new	short	work
e8	???	unknown	new	short	work



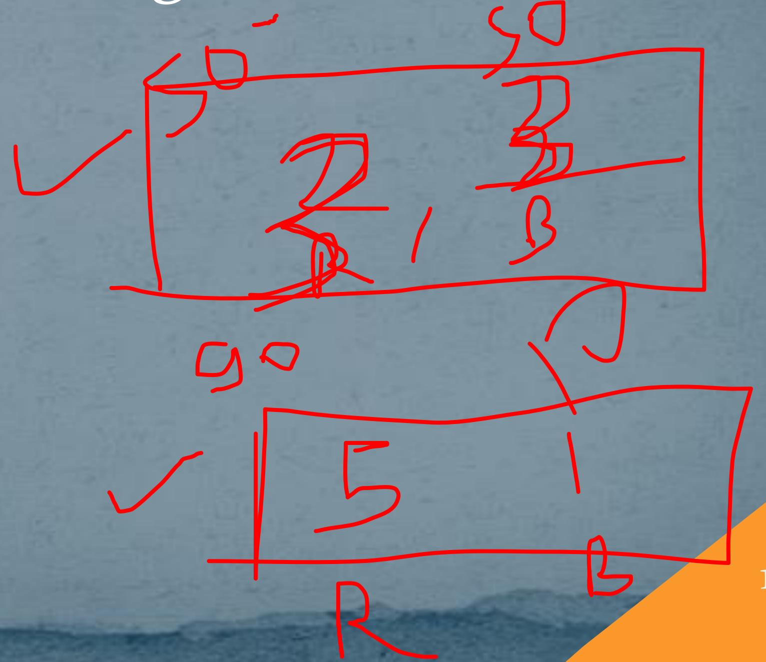


How to choose attribute for splitting?

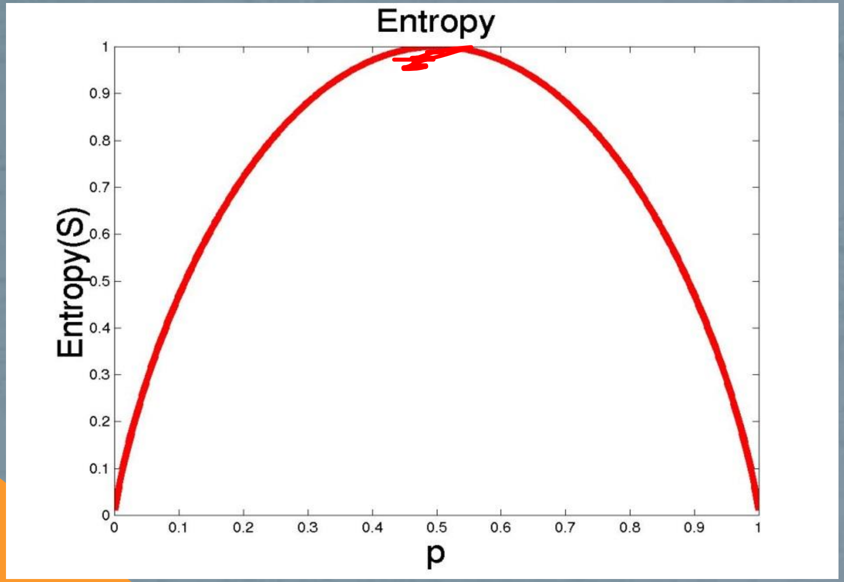
- Attribute selection measure is a heuristic for selecting the splitting criterion that partition data into the best possible manner.
- Attribute with best score of ASM will be selected as a splitting attribute .
- Most popular selection measures are
  - Information Gain (ID3 Algorithm)
  - Gini Index (CART Algorithm)

Information Gain

- Entropy referred as the randomness or the impurity in the system. In information theory, it refers to the impurity in a group of examples.
- Less Entropy means high information gain.
- Information gain computes the difference between entropy before split and average entropy after split of the dataset based on given attribute values



Entropy



- The entropy is 0 if the outcome is ``certain”.
- The entropy is maximum if we have no knowledge of the system (or any outcome is equally possible).

Entropy

- If S is set of training instances
- P_1 is the proportion of class 1 instances
- P_2 is the proportion of class 2 instances
- Entropy measures the impurity of S, using the formula

  $Entropy(S) = -P_1 \log_2 P_1 - P_2 \log_2 P_2$
- Similarly for m classes,
  - $Entropy(S) = - \sum_{i=1}^{m} P_i \log_2 P_i$

Information Gain

- Information Gain of an attribute A in a set S is given by

$$Gain(S,A)=Entropy(S) - \sum_{v \in values(A)} |S_v|/|S| \; Entropy(S_v)$$

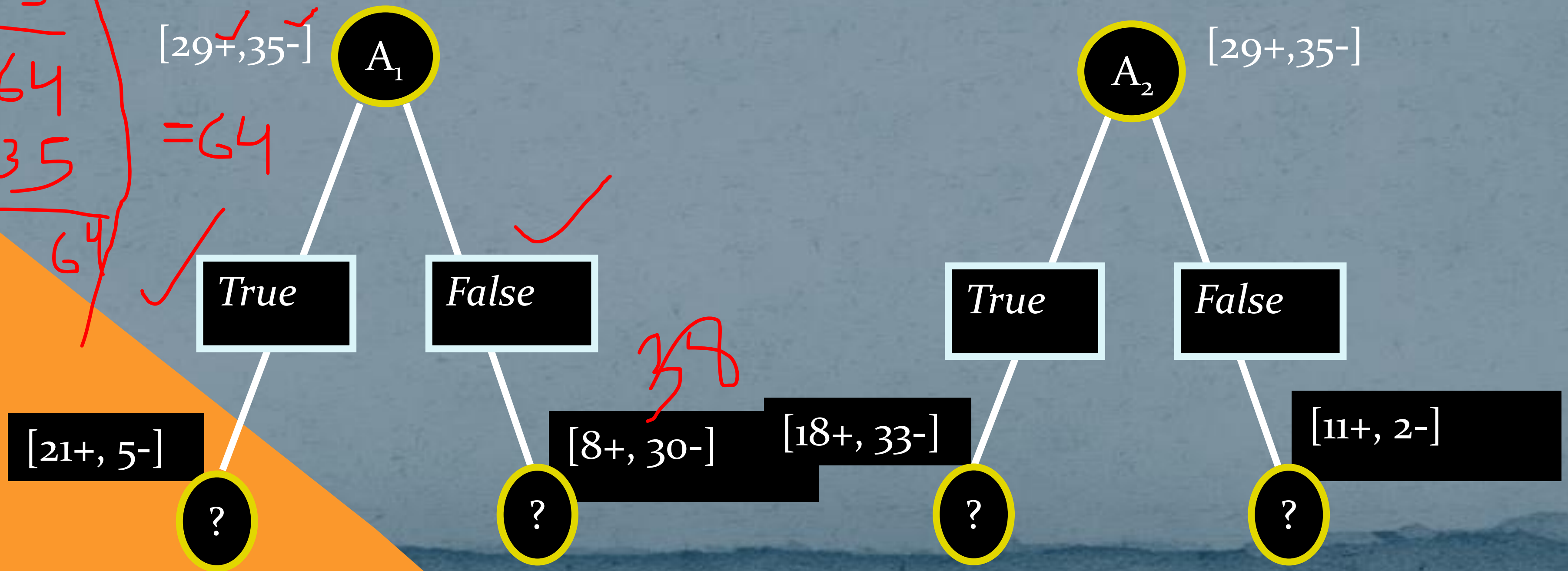
- where v - Values of attribute A

  $|S_v|$- Number of instances in the set S having value v for attribute A

Example

- Let us consider a case where target output has two possible classes positive (+) and negative(-). Total number of training instances is 64.
- Out of two attributes A_1 and A_2 , which will be considered as best are to be chosen

$$-\left(\frac{29}{64}\log_2\frac{29}{64} + \frac{35}{64}\log_2\frac{35}{64}\right)$$



Example: Find Information Gain

- Entropy(S) = Entropy([29+,35-])
  $= -29/64 \log_2 29/64 - 35/64 \log_2 35/64$
  $= 0.99$

A1

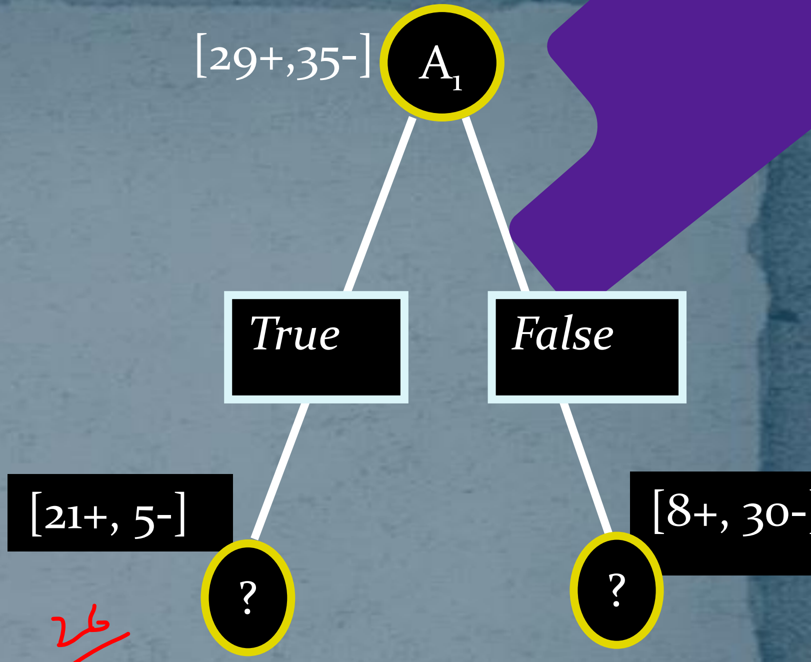
Entropy([21+,5-]) = 0.71
Entropy([8+,30-]) = 0.74
Gain(S,A_1)=Entropy(S)
-26/64*Entropy([21+,5-])
-38/64*Entropy([8+,30-])
=0.27

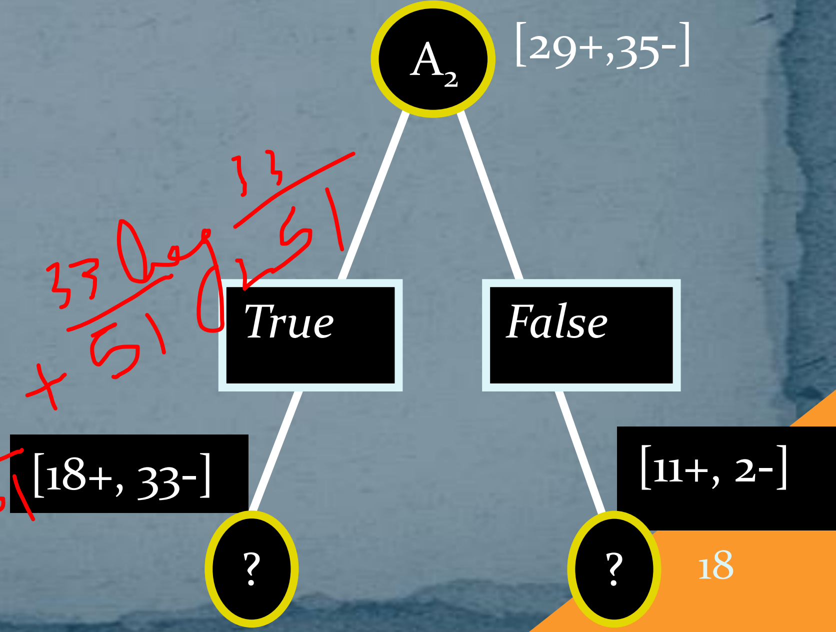
A2

Entropy([18+,33-]) = 0.94
Entropy([8+,30-]) = 0.62
Gain(S,A_2)=Entropy(S)
-51/64*Entropy([18+,33-])
-13/64*Entropy([11+,2-])
=0.12

$\frac{26}{64}$

$\frac{18}{51} \log \frac{18}{51} + \frac{33}{51} \log \frac{33}{51}$





Gain(S,A)=Entropy(S) $-\sum_{v \in values(A)} |S_v|/|S|$ Entropy(S_v)

Another Measure : GINI Index

$$GINI_{node}(Node) = 1 - \sum_{c \in classes} [p(c)]^2$$

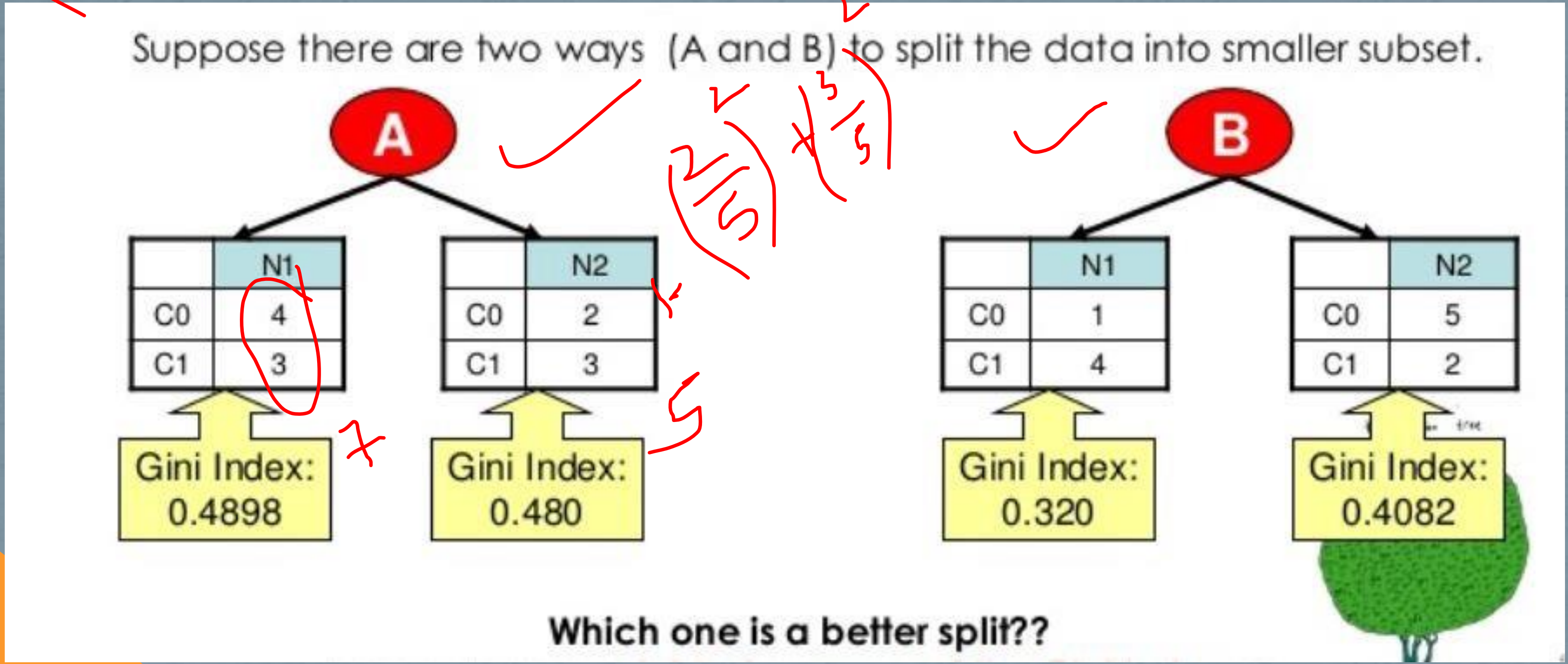
$$GINI_{split}(A) = \sum_{v \in Values(A)} \frac{|S_v|}{|S|} GINI(N_v)$$

- where

p(c) - proportion of class c instances in the node

$GINI(N_v)$ - represent the GINI index for node with value v of attribute A

$1-\left(\left(\frac{4}{7}\right)^2+\left(\frac{3}{7}\right)^2\right)$



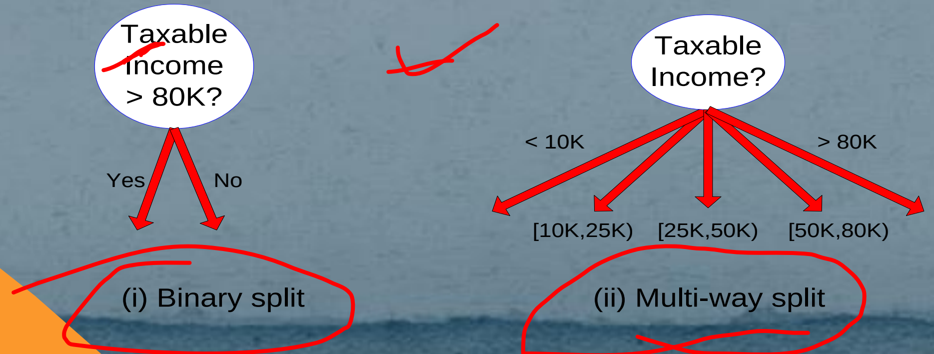
- $GINI_{split}(A)=(7/12)*0.4898+(5/12)*0.480=0.4857$
- $GINI_{split}(B)=(5/12)*0.320+(7/12)*0.4082=0.3715$

Generate decision tree
(Target Feature is Tennis?)

Day	Outlook	Temp	Humidity	Wind	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

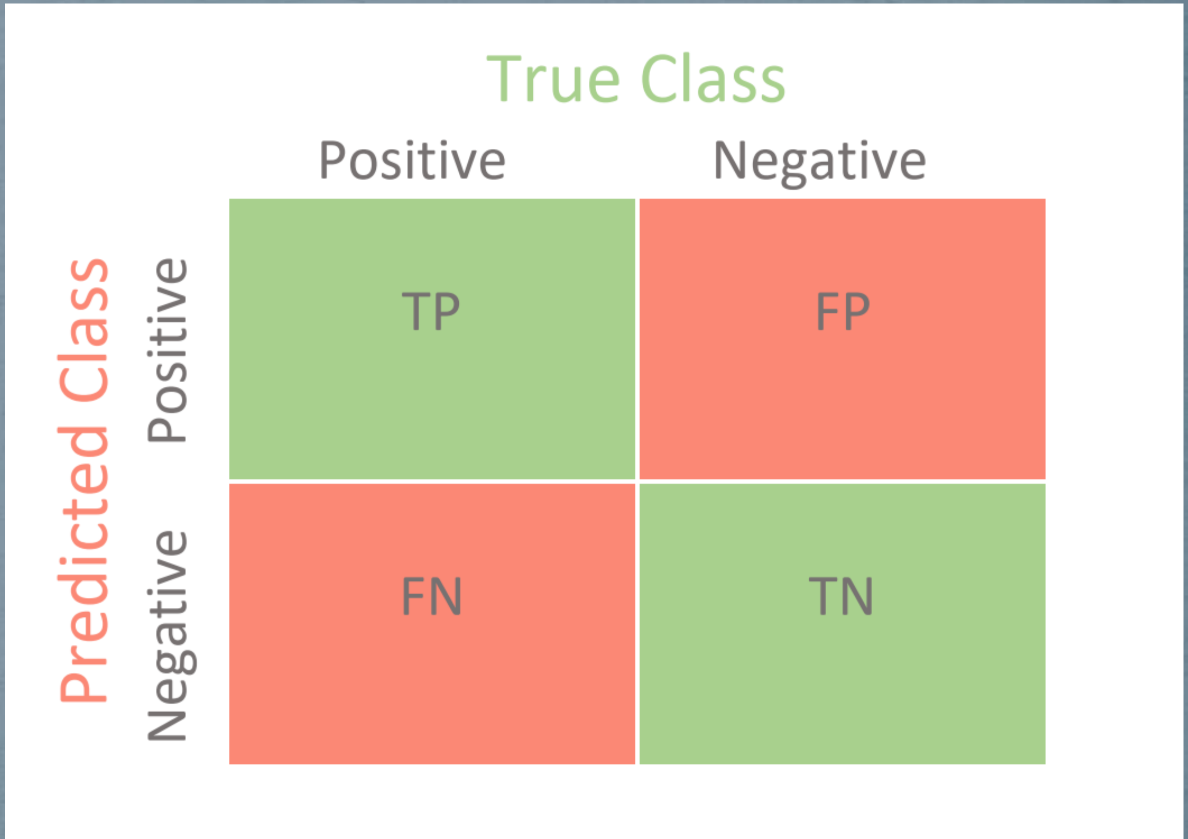
Limitations

- A small change in the data can cause a large change in the structure of the decision tree causing instability.
- For a Decision tree sometimes calculation can go far more complex compared to other algorithms.
- Decision tree often involves higher time to train the model.
- Handling missing values could be challenge.
- The Decision Tree algorithm is inadequate for applying regression and predicting continuous values but can be handled as follows



Metrics for Classifiers

- A confusion matrix is a table that is often used to describe the performance of a classification model (or "classifier") on a set of test data for which the true values are known
- It is extremely useful for measuring
  - Recall
  - Precision
  - Accuracy
  - F-Score



Metrics for Classifiers

$$Recall = \frac{TP}{TP + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Accuracy = \frac{TP+TN}{Total}$$

$$F\text{-}measure = \frac{2*Recall*Precision}{Recall + Precision}$$

THANK YOU!!!!