SNS COLLEGE OF ENGINEERING

Kurumbapalayam (Po), Coimbatore – 641 107

An Autonomous Institution

Accredited by NAAC – UGC with 'A' Grade

Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

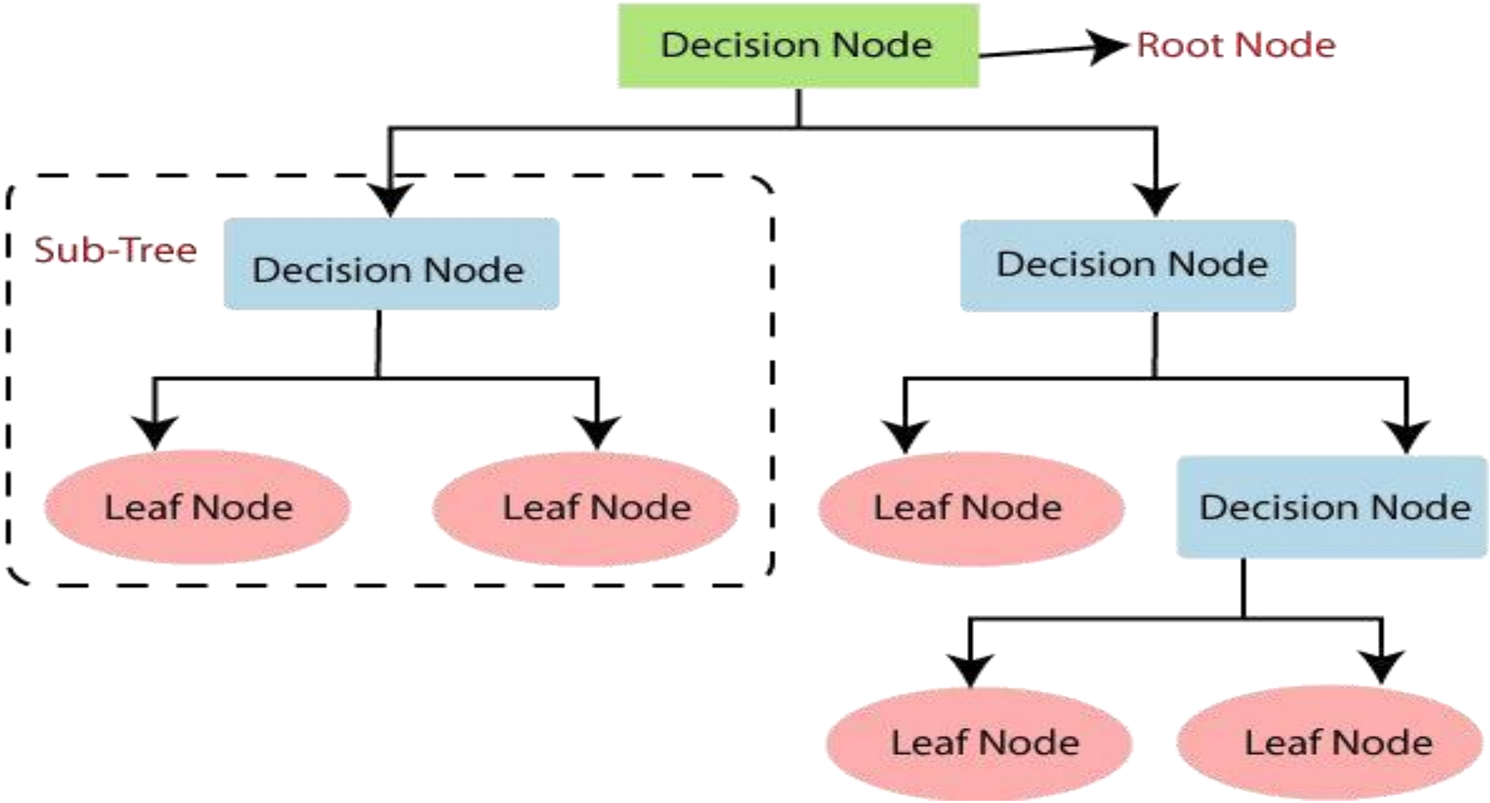
DEPARTMENT OF COMPUTER SCIENCE AND TECHNOLOGY

COURSE NAME : 19CS407-DATA ANALYTICS WITH R

II YEAR /IV SEMESTER

Unit II – Statistics and Prescriptive Analytics

Topic : Decision Tree


Decision Node
Root Node
Sub-Tree
Decision Node
Decision Node
Leaf Node
Leaf Node
Leaf Node
Decision Node
Leaf Node
Leaf Node

Why use Decision Trees?

There are various algorithms in Machine learning, so choosing the best algorithm for the given dataset and problem is the main point to remember while creating a machine learning model.

Below are the two reasons for using the Decision tree:

- Decision Trees usually mimic human thinking ability while making a decision, so it is easy to understand.
- The logic behind the decision tree can be easily understood because it shows a tree-like structure.

Decision Tree Terminologies

- **Root Node:** Root node is from where the decision tree starts. It represents the entire dataset, which further gets divided into two or more homogeneous sets.
- **Leaf Node:** Leaf nodes are the final output node, and the tree cannot be segregated further after getting a leaf node.
- **Splitting:** Splitting is the process of dividing the decision node/root node into sub-nodes according to the given conditions.
- **Branch/Sub Tree:** A tree formed by splitting the tree.
- **Pruning:** Pruning is the process of removing the unwanted branches from the tree.
- **Parent/Child node:** The root node of the tree is called the parent node, and other nodes are called the child nodes.

How does the Decision Tree algorithm Work?

In a decision tree, for predicting the class of the given dataset, the algorithm starts from the root node of the tree.

This algorithm compares the values of root attribute with the record (real dataset) attribute and, based on the comparison, follows the branch and jumps to the next node.

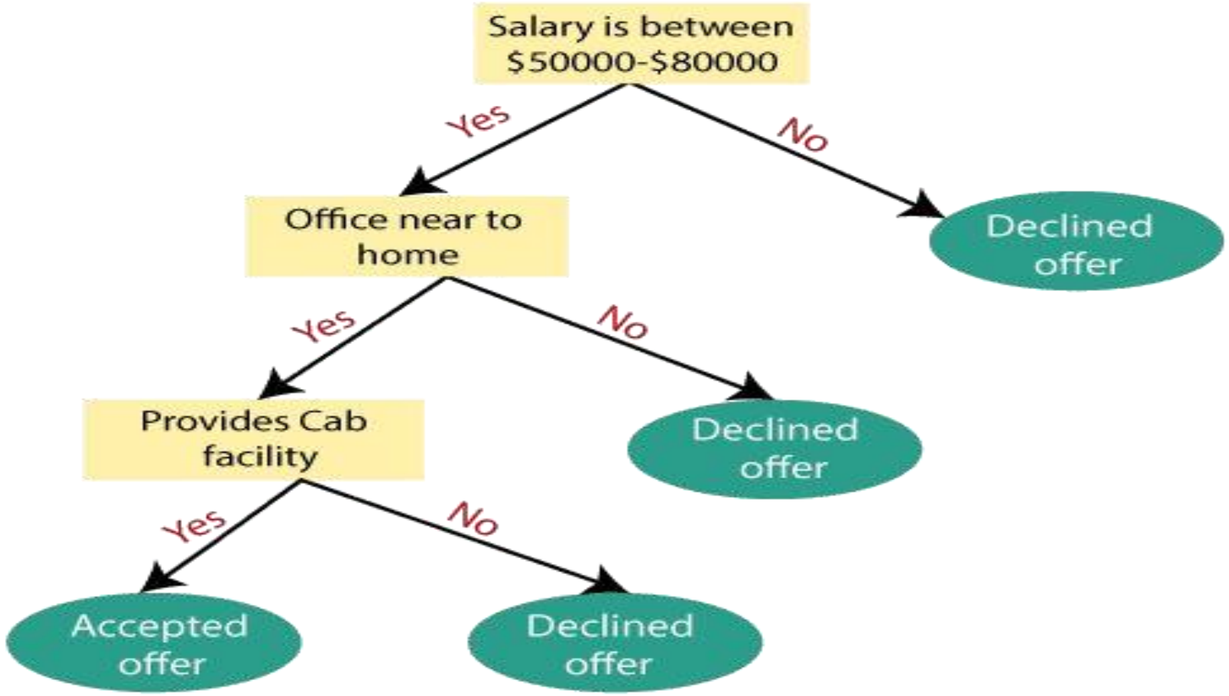
Decision Trees Algorithm

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.

Example

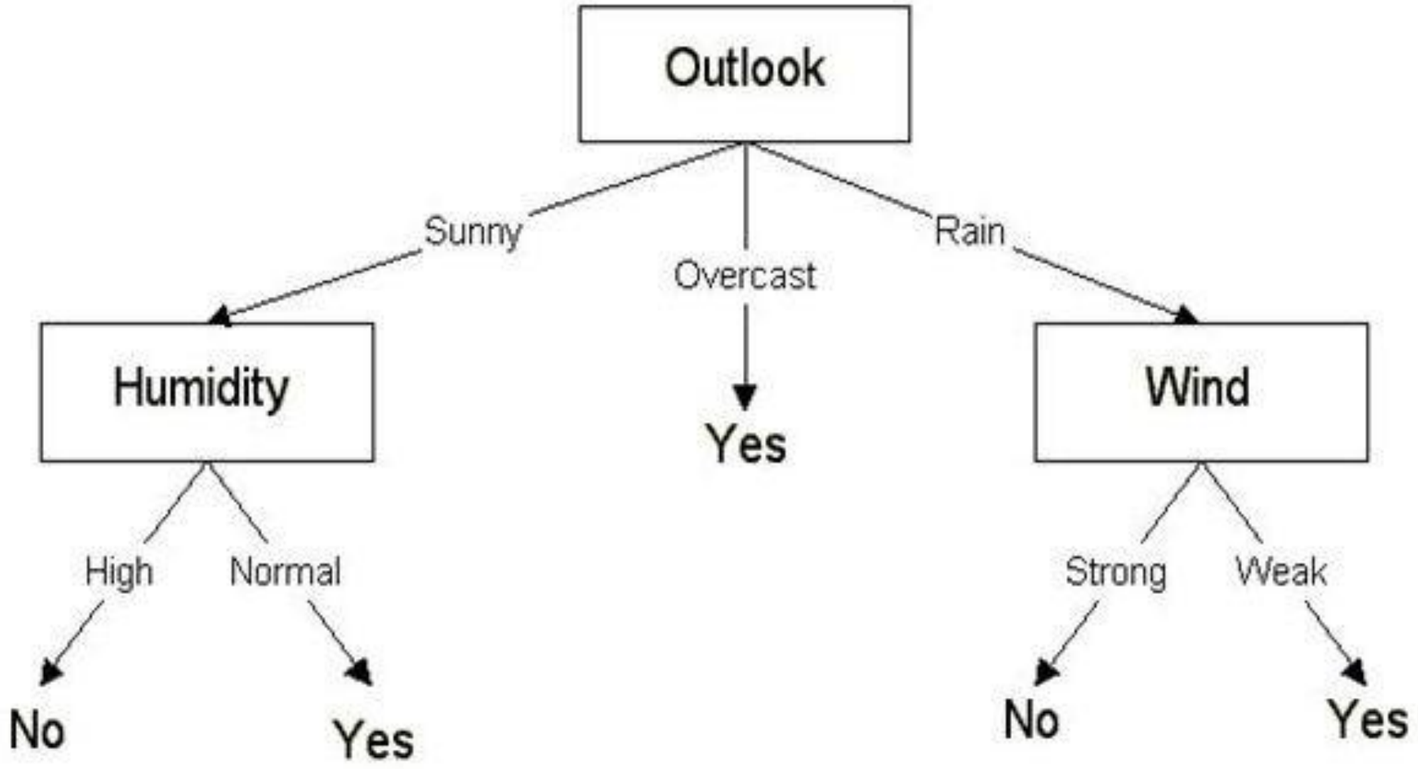
Example: Suppose there is a candidate who has a job offer and wants to decide whether he should accept the offer or Not. So, to solve this problem, the decision tree starts with the root node (Salary attribute by ASM). The root node splits further into the next decision node (distance from the office) 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 below diagram:



Consider whether a dataset based on which we will determine whether to play football or not.

Outlook	Temperature	Humidity	Wind	Played football(yes/no)
Sunny	Hot	High	Weak	No
Sunny	Hot	High	Strong	No
Overcast	Hot	High	Weak	Yes
Rain	Mild	High	Weak	Yes
Rain	Cool	Normal	Weak	Yes
Rain	Cool	Normal	Strong	No
Overcast	Cool	Normal	Strong	Yes
Sunny	Mild	High	Weak	No
Sunny	Cool	Normal	Weak	Yes
Rain	Mild	Normal	Weak	Yes
Sunny	Mild	Normal	Strong	Yes
Overcast	Mild	High	Strong	Yes
Overcast	Hot	Normal	Weak	Yes
Rain	Mild	High	Strong	No

Decision tree for the above data set?



Attribute Selection Measures

- While implementing a Decision tree, the main issue arises that how to select the best attribute for the root node and for sub-nodes. So, to solve such problems there is a technique which is called as **Attribute selection measure or ASM.**
- By this measurement, we can easily select the best attribute for the nodes of the tree.

There are two popular techniques for ASM, which are:

- **Information Gain**
- **Gini Index**

Information Gain:

- Information gain is the measurement of changes in entropy after the segmentation of a dataset based on an attribute.
- It calculates how much information a feature provides us about a class.
- According to the value of information gain, we split the node and build the decision tree.
- A decision tree algorithm always tries to maximize the value of information gain, and a node/attribute having the highest information gain is split first. It can be calculated using the below formula:
- **Information Gain= Entropy(S)- [(Weighted Avg) *Entropy(each feature)**

Entropy:

Entropy is a metric to measure the impurity in a given attribute. It specifies randomness in data. Entropy can be calculated as: **Entropy(s)= - P(yes)log2 P(yes) - P(no) log2 P(no)**

Where,

- S= Total number of samples
- P(yes)= probability of yes
- P(no)= probability of no

Gini Index:

- Gini index is a measure of impurity or purity used while creating a decision tree in the CART(Classification and Regression Tree) algorithm.
- An attribute with the low Gini index should be preferred as compared to the high Gini index.
- It only creates binary splits, and the CART algorithm uses the Gini index to create binary splits.
- Gini index can be calculated using the below formula:

$$\text{Gini Index} = 1 - \sum_j P_j^2$$

Pruning: Getting an Optimal Decision tree

Pruning is a process of deleting the unnecessary nodes from a tree in order to get the optimal decision tree.

A too-large tree increases the risk of overfitting, and a small tree may not capture all the important features of the dataset. Therefore, a technique that decreases the size of the learning tree without reducing accuracy is known as Pruning. There are mainly two types of tree **pruning** technology used:

- **Cost Complexity Pruning**
- **Reduced Error Pruning.**

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.

Problem on Decision Tree -Classification using the ID3 algorithm

Outlook	Temperature	Humidity	Wind	Played football(yes/no)
Sunny	Hot	High	Weak	No
Sunny	Hot	High	Strong	No
Overcast	Hot	High	Weak	Yes
Rain	Mild	High	Weak	Yes
Rain	Cool	Normal	Weak	Yes
Rain	Cool	Normal	Strong	No
Overcast	Cool	Normal	Strong	Yes
Sunny	Mild	High	Weak	No
Sunny	Cool	Normal	Weak	Yes
Rain	Mild	Normal	Weak	Yes
Sunny	Mild	Normal	Strong	Yes
Overcast	Mild	High	Strong	Yes
Overcast	Hot	Normal	Weak	Yes
Rain	Mild	High	Strong	No

Here typically we will take log to base 2. Here total there are 14 yes/no.

Out of which 9 yes and 5 no.

Based on it we calculated probability above.

Entropy(s)= -P(yes)log2 P(yes) - P(no) log2 P(no)

Find the entropy of the class variable..

E(S) = -[(9/14)log(9/14) + (5/14)log(5/14)] = 0.94

From the above data for outlook we can arrive at the following table easily

		play		
		yes	no	total
	sunny	3	2	5
Outlook	overcast	4	0	4
	rainy	2	3	5
				14

Now we have to calculate average weighted entropy. ie, we have found the total of weights of each feature multiplied by probabilities.

E(S, outlook) = (5/14)*E(3,2) + (4/14)*E(4,0) + (5/14)*E(2,3) =

(5/14)(-(3/5)log(3/5)-(2/5)log(2/5))+ (4/14)(0) +

(5/14)((2/5)log(2/5)-(3/5)log(3/5))

= 0.693

The next step is to find the information gain. It is the difference between parent entropy and average weighted entropy we found above.

IG(S, outlook) = 0.94 - 0.693 = 0.247

Similarly find Information gain for Temperature, Humidity, and Windy. IG(S, Temperature) = 0.940 - 0.911 = 0.029

IG(S, Humidity) = 0.940 - 0.788 = 0.152

IG(S, Windy) = 0.940 - 0.8932 = 0.048

Now select the feature having the largest entropy gain. Here it is Outlook. So it forms the first node(root node) of our decision tree.

Classification using CART algorithm

Classification using CART is similar to it. But instead of entropy, we use Gini impurity.

So as the first step we will find the root node of our decision tree. For that Calculate the Gini index of the class variable

$Gini(S) = 1 - [(9/14)^2 + (5/14)^2] = 0.4591$

As the next step, we will calculate the Gini gain. For that first, we will find the average weighted Gini impurity of Outlook, Temperature, Humidity, and Windy. First, consider case of Outlook

Classification using CART algorithm

Gini(S, outlook) = (5/14)gini(3,2) + (4/14)*gini(4,0)+ (5/14)*gini(2,3)
= (5/14)(1 - (3/5)² - (2/5)²) + (4/14)*0 + (5/14)(1 - (2/5)² - (3/5)²)

		play		
		yes	no	total
Outlook	sunny	3	2	5
	overcast	4	0	4
	rainy	2	3	5
				14

= 0.171+0+0.171 = 0.342

Gini gain (S, outlook) = 0.459 - 0.342 = 0.117
Gini gain(S, Temperature) = 0.459 - 0.4405 = 0.0185
Gini gain(S, Humidity) = 0.459 - 0.3674 = 0.0916
Gini gain(S, windy) = 0.459 - 0.4286 = 0.0304

Choose one that has a higher Gini gain. Gini gain is higher for outlook.
So we can choose it as our root node.

References

TEXT BOOKS

1.João Moreira, Andre Carvalho, Tomás Horvath – "A General Introduction to Data Analytics" – Wiley - 2018

2.An Introduction to R, Notes on R: A Programming Environment for Data Analysis and Graphics. W. N. Venables, D.M. Smith and the R Development Core Team. Version 3.0.1 (2013-05-16). URL: https://cran.r-project.org/doc/manuals/r-release/R-intro.pdf

References:

1**. Dean J, —***Big Data, Data Mining and Machine learning***,** Wiley publications, 2014.

2. Provost F and Fawcett T, —*Data Science for Business*, O'Reilly Media Inc, 2013.

3. Janert PK, —*Data Analysis with Open Source Tools*, O'Reilly Media Inc, 2011.

4. Weiss SM, Indurkhya N and Zhang T, —*Fundamentals of Predictive Text Mining,* Springer-Verlag London Limited, 2010.

5.**Marz N and Warren J**,- Big Data, Manning Publications,2015

Thank You