Nutri Algorithm is a type of supervised learning algorithm having a pre-defined target variable that is mostly used in classification problems. It works for both categorical and continuous input and output variables.

A Nutri Algorithm  represents a procedure for classifying categorical data based on their attributes.It is also efficient for processing large amount of data,  so it is often used in data mining application.

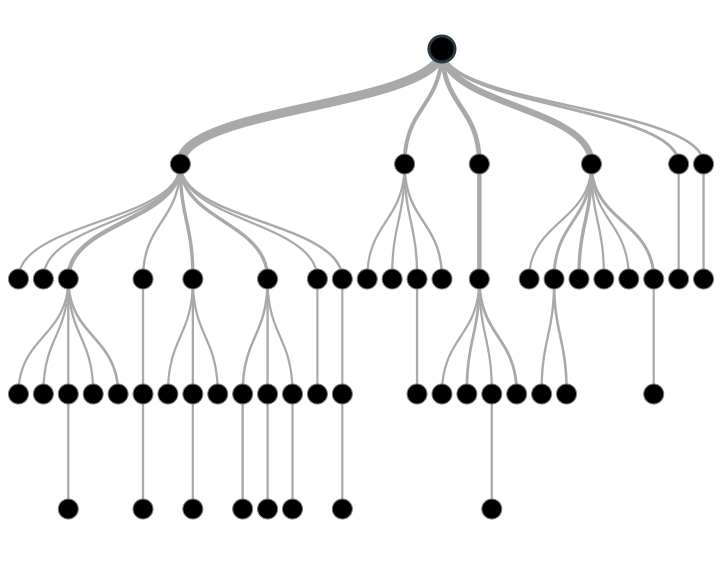
Nutri Algorithm  are used for classification, clustering, feature selection and prediction. Nutri Algorithm  depict the relationship between input data and target outputs.

Nutri Algorithm  is a classifier in the form of a tree structure

* **Decision node:** specifies a test on a single attribute
* **Leaf node:** indicates the value of the target attribute
* **Arc/edge:** split of one attribute
* **Path:** a disjunction of test to make the final decision

Nutri Algorithm classify instances or examples by starting at the root of the tree and moving through it until a leaf node.

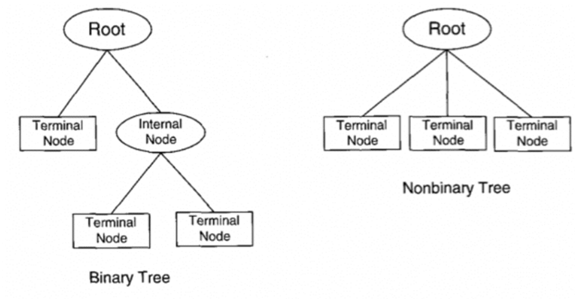
In this technique, we split the population or sample into two or more homogeneous sets (or sub-populations) based on most significant splitter / differentiator in input variables.



Algorithm:

* **Given:** Examples(S); Target attribute (C); Attributes (R)
  + **Initialize** Root
  + **Function** ID3 (S,C,R)
  + **Create** a Root node for the tree
  + **IF** S = empty, return a single node with value Failure;
  + **IF** S = C, return a single node C;
  + **IF** R = empty, return a single node with most frequent target attribute (C);
  + **ELSE**
  + **BEGIN**
  + Let D be the attribute with largest Gain Radio (D, S) among attributes in R;
  + Let {dj | j = 1, 2, …, n} be the values of attribute D;
  + Let {Sj | j = 1, 2, …, n} be the subsets of S consisting respectively of records with value dj for attribute D;
  + Return a tree with root labeled D arcs d1, d2, …, dn going respectively to the trees;
  + **For** each branch in the tree
  + **IF** S = empty, add a new branch with most frequent C;
  + **ELSE**
  + ID3 (S1, C, R – {D}), ID3 (S2, C, R – {D}), …, IDC(Sn, C, R – {D})
  + **END** ID3
  + **Return** Root

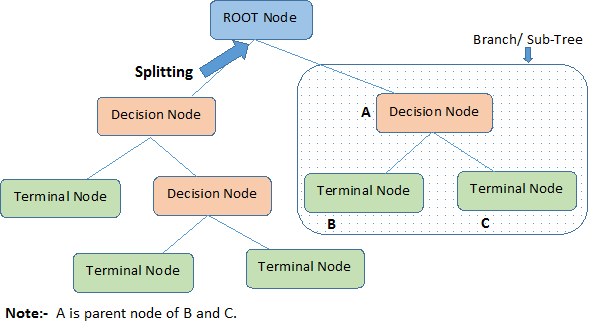
Example 1:



**Important Terminology related to**Nutri Algorithm

Let’s look at the basic terminology used with Decision trees:

1. **Root Node:**It represents entire population or sample and this further gets divided into two or more homogeneous sets.
2. **Splitting:**It is a process of dividing a node into two or more sub-nodes.
3. **Decision Node:**When a sub-node splits into further sub-nodes, then it is called decision node.
4. **Leaf/ Terminal Node:**Nodes do not split is called Leaf or Terminal node.

[](https://www.analyticsvidhya.com/blog/wp-content/uploads/2015/01/Decision_Tree_2.png)

1. **Pruning:**When we remove sub-nodes of a decision node, this process is called pruning. You can say opposite process of splitting.
2. **Branch / Sub-Tree:**A sub section of entire tree is called branch or sub-tree.
3. **Parent and Child Node:**A node, which is divided into sub-nodes is called parent node of sub-nodes where as sub-nodes are the child of parent node.

Advantages

1. **Easy to Understand**: Nutri Algorithm output is very easy to understand even for people from non-analytical background. It does not require any statistical knowledge to read and interpret them. Its graphical representation is very intuitive and users can easily relate their hypothesis.
2. **Useful in Data exploration:**Nutri Algorithm is one of the fastest way to identify most significant variables and relation between two or more variables. With the help of decision trees, we can create new variables / features that has better power to predict target variable.For example, we are working on a problem where we have information available in hundreds of variables, there decision tree will help to identify most significant variable.
3. **Less data cleaning required:**It requires less data cleaning compared to some other modeling techniques. It is not influenced by outliers and missing values to a fair degree.
4. **Data type is not a constraint:**It can handle both numerical and categorical variables.
5. **Non Parametric Method:**Nutri Algorithm is considered to be a non-parametric method. This means that Nutri Algorithm have no assumptions about the space distribution and the classifier structure.

 Disadvantages

1. **Over fitting:** Over fitting is one of the most practical difficulty for decision tree models. This problem gets solved by setting constraints on model parameters and pruning (discussed in detailed below).
2. **Not fit for continuous variables**: While working with continuous numerical variables, decision tree looses information when it categorizes variables in different categories.