# Advanced Business Decision Support Systems

**Professor Deepu Philip**
**Department of Industrial Engineering and Management Engineering**
**Indian Institute of Technology, Kanpur**
**Professor Amandeep Singh**
**Imagineering Laboratory**
**Dr. Prabal Pratap Singh**
**Indian Institute of Technology, Kanpur**

**Lecture 13**

## Decision Tree Algorithm for Business Decision (Part 1 of 3)

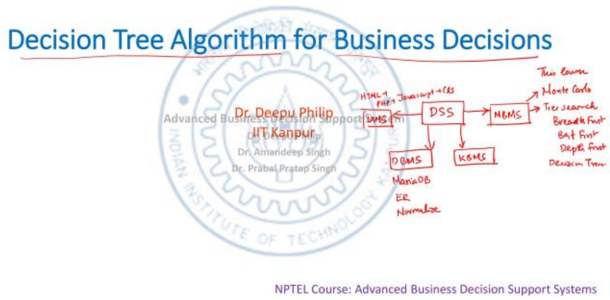
Good afternoon everyone. Welcome to the fourth week of the Business Decision Support System, the advanced course of the Web-Based Decision Support System under the NPTEL MOOC's program from IIT Kanpur. I am Dr. Deepu Philip and along with me Dr. Amandeep Singh Oberoi and Dr. Prabhal Pratap Singh are teaching this course.

And, so far, we have over viewed the what is a Decision Support System and we seen what is the Business Decision Process and how the Business Decision Process is different from the typical decision making process and what constitutes a Business Decision and the stakeholder and the evaluation, all those kind of things, we have already discussed.

And, we also seen, how the corporate decisions are typically made in a collective thing and the different tiers of corporate the lower level decisions that the strategic tactical operation decisions etcetera or day to day decisions. And we also then moved into the over view of the DSS web based DSS which was a precursor course of this one. And, then we started looking into various models, that are used for decision making.

And, because the previous course, we have covered mostly the Database Management System and the User Interface mostly and very few on Model and Knowledge Base. And, the advanced course, we will be spending more on Modeling and also on Knowledge Base. So far, we have seen what is Monte Carlo Decision Making Models and how can you use Monte Carlo to make decision models under uncertainty or randomness.

And, then after, we have seen the Tree Search approach especially for decision making using the single Machine Scheduling problem. We have already seen how to use Breadth First and best First Search and also, we introduced the concept of Branch and Bound into this. And, now finally, we have that time the class itself, we discussed that we will also look into what you call as a Decision Tree Algorithm.

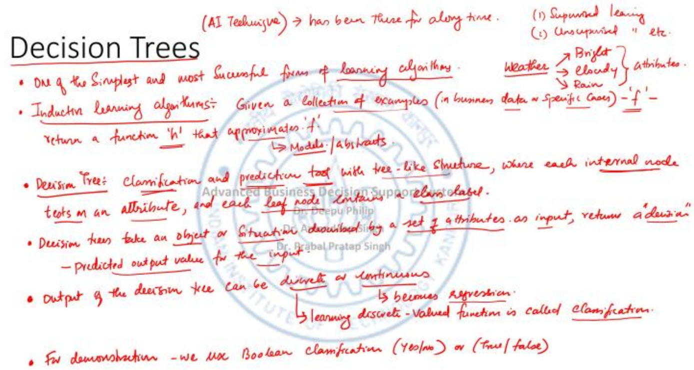



So, without much further delay, let us look into today's topic that is the Decision Tree Algorithm and how it is used for Business Decisions. Instead of discussing a business decision, I would be mostly discussing a example problem in this class. So, as I said earlier you have the Decision Support System DSS and the major components of it is one is the UIMS (User Interface Management System). And, we already seen this using HTML and PHP + JavaScript.

We already seen this plus CSS as part of the introductory course. Then, we had DBMS (Database Management System) we seen this under the MariaDB or MySQL then, we seen the ER diagram, the normalizations, all those aspects of database we have covered that to us in the previous course, the Web-Based Decision Support System. And, then we have discussed two more components of it is one is the MBMS (Model Based Management System) and we are discussing this. We already seen what we call as the Monte Carlo. So, this is course and then, the Tree Search.

In this, we have seen what is Breadth First, Best First is in Depth First. All these things in Tree Search. And, today we are going to do a learning algorithm which is called as Decision trees. And, the last component of this is the KBMS (Knowledge Based Management System). Once we cover the Model Based Management System, various decision models and etcetera.

Then, we will move to what we call as the Knowledge Based Management System. May be introduce you guys to that concept and then go from there. So, without further delay, let us look into what is a Decision Tree.




So, the Decision Trees is, lot of the people associate this with the currently people say, it is an AI technique or something, but it has been there for a long time, it is not just, it need not be. It is has been there for a long time.

It need not be a new AI thing that, we ask, people try to make it sound like been there. And, lot of it is used in Business Decisions. So, the first characteristic, it is one of the simplest and most successful forms of learning algorithms. So, you may have heard about many type of learning algorithms, you might have heard the term called Supervised Learning, Unsupervised Learning, Semi-Supervised Learning, etcetera. So, we are not getting into that, but just understand that the Decision Tree is a part of the learning algorithms.

So, in this case, it is you know, there is one category of algorithm that I would like to say Inductive Learning Algorithms. The AI algorithms, learning algorithms are something like out of this, we are focusing more on Decision Making. So, the Inductive learning algorithm. So, what it is that collection of examples in business, it is data or specific cases, So, given a collection of examples and let us denote this examples by 'f'. Then, return a function 'h' that approximates 'f'.

So, this approximates or in another words you can use models 'f'. Let us say remember or abstracts 'f' or approximates either words are fine. So, there is a class of algorithms Inductive learning algorithms when you have given a collection of examples in or in business, you call this data or a specific case based, whatever it is. So, we denote that by 'f' If you have that, then Inductive learning algorithms of the capability to return a function 'h' which can approximate which can model or which can abstract 'f'. Then, Decision Tree is a classification and prediction tool.

For forecasting tool, you can say prediction tool. We use the word prediction here with tree like structure. So, you can say, it is a classification prediction tool. I am using the word tool here, if you want, you can use a model or algorithm. It does not matters.

So, it is a classification and prediction tool model, that is having a with a tree like structure, where each internal node, we seen what a Root node what is an Internal node all in the previous class. So, refer to the lecture Internal node tests on an attribute. So, each Internal node tests on an attribute and each leaf is the terminal node. Each leaf node contains a class label. Class label is you can think about as decision.

So, the point is, it is a classification or a prediction tool, that has a tree-like structure, where each internal node test on an attribute of certain things. In this case, we can call it as, an independent variable. We will talk about that later on certain attribute and the leaf node, which is a terminal node, contains a class label. Mostly class label means an output or an outcome or a decision. So, Decision Trees take an object or situation described by a set of attributes.

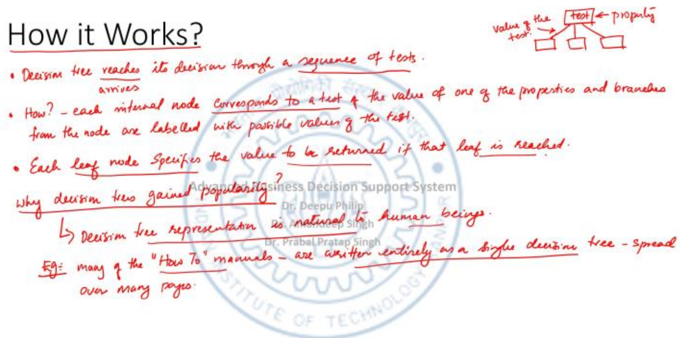
So, the attributes is either an object or a situation, it is described by a set of attributes. Remember, in the previous thing, we have already discussed in the database, what are attributes and how entities and attributes comes into picture? So, if I say weather as a situation. Then, I may classify the weather as bright cloudy and I will say it as rain. So, these three becomes the attributes of the weather, which is kind of, you know can be rainy weather, can be bright weather, can be cloudy weather.

That kind of a thing. So, it can be described by a set of attributes as input. So, the attributes become input and returns a decision putting the decision in double cards. What is the decision? Decision is the predicted output value for the input. So, Decision Tree takes an object or situation described by a set of attributes as the input. So, it takes that input by either an object or situation and along with attributes and returns a decision.

What is the decision? Decision is the predicted output value for that given input, whatever be the input. Now, other part is the output of the Decision Tree can be discrete or continuous. So, two types of outputs are possible. It can be the discrete output or can be continuous output. So, if it is discrete learning, discrete valued function is called Classification.

If you are learning discrete valued functions it becomes Classification. If it is a continued values, it becomes regression. So, the output can be discrete or continuous. If it is discrete valued output, then you are learning the discrete valued aspect is called as Classification. If it is the continuous one, it is called as Regression.

For demonstration, we use Boolean Classification which implies yes or no or true or false. So, it is much more easy to understand the mechanics of this process if we use this Boolean classification.




So, next is how the Decision Tree works? So, the in simplest sense, Decision Tree reaches decision or it arrives at this decision through a sequence of tests. So, it conducts a sequence of tests. So, how is it conducts the test? So, the question is how it conducts a sequence of test? The answer to that is, each internal node of whom the decision tree corresponds to a test.

So, the internal node of the tree corresponds to a test of the value of one of the properties. So, you can think about one of the properties of the attribute. So, each internal node corresponds to a test of the value of one of the properties. It tests on one of the value of the properties and branches from the node are labeled with possible values of the test. So, what happens is, the one way to think about is each node has a test.

This test is one of the property. This is where you test the property and it will have multiple branches and the each one of these branches will maybe give rise to other nodes, but these branches are the value of the test. So, whatever be the value of the property is, what will be there. So, if I use the weather as the test, then one of them will be bright cloudy and rain that will be the value of the test. Then, each leaf node specifies the value to be returned value if that leaf is reached. So, each leaf node specifies the value to be returned.

If that leaf node is reached, what is that value it need to be returned? Then, other thing is that, why Decision Trees gained popularity. This is one main thing. What makes it so popular? People say that, I have been using Decision tree. All these kind of things why is it very popular. The main answer to that is, Decision Tree representation is natural to human beings.

The Decision Tree representation is natural to human beings. So, classical example of this, many of the "How to" manuals, how to assemble something, how to do this, how to do that, all those kind of things. Manuals are written entirely as a single Decision Tree spread over many pages. So, it is very natural, the Decision Tree representation is extremely natural. It is second nature to us human beings.

So, it is the way we understand or we perceive things. So, that is one of the other reasons why Decision Tree is very popular. So, an example is, how to manual, how to assemble a chair, how to do this how to swim, all those kind of things that typically a Single Decision Tree return over large number of pages. So, we seen the working mechanism of the Decision Tree.


Illustrative Example

• Let's decide whether to go for fishing or not

| Sl. No | Weather | Temperature | Humidity | Wind | Fishing? |
|---|---|---|---|---|---|
| 1 | Bright | Hot | High | Calm | No |
| 2 | Bright | Hot | High | Gusty | No |
| 3 | Cloudy | Hot | High | Calm | Yes |
| 4 | Rain | Mild | High | Calm | Yes |
| 5 | Rain | Cool | Normal | Calm | Yes |
| 6 | Rain | Cool | Normal | Gusty | No |
| 7 | Cloudy | Cool | Normal | Gusty | Yes |
| 8 | Bright | Mild | High | Calm | No |
| 9 | Bright | Cool | Normal | Calm | Yes |
| 10 | Rain | Mild | Normal | Calm | Yes |
| 11 | Bright | Mild | Normal | Gusty | Yes |
| 12 | Cloudy | Mild | High | Gusty | Yes |
| 13 | Cloudy | Hot | Normal | Calm | Yes |
| 14 | Rain | Mild | High | Gusty | No |



So, then if that is the case, let us now move to an illustrative example and we will solve using this illustrative example.

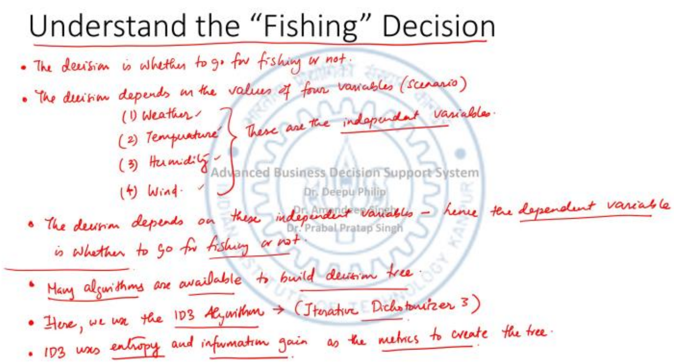
So, you will understand. So, the decision today we are going to take is a simple toy problem decision. we are going to take today is, whether to go for fishing or not. So, what you have given here is a data set. This whole thing is a data set. You have 14 data sets and in which you have various.

So, remember I have told you, there is like a set of attributes, an object or a situation described by a set of attributes. So, if you look into it, this is the situation. The situation is described by weather, temperature, humidity and wind. These 4 describe the situation together creates a situation and each one of these the bright cloudy rain.

These are the attributes of weather. Same way, hot, mild and cool. They are the attributes of temperature. So, then what is the decision. So, this is the decision.

Decision is no and yes it is binary. So, what happens here is, in the no means, if it is bright, if sun is shining bright and the temperature is hot, humidity is high and the wind is gusty, the decision is no. No means, do not go for fishing ok. If you look at the other way, if the weather is cloudy, the temperature is hot, humidity is high, but the wind is calm, then go for fishing. Yes means, go for fishing.

So, you can see that, there is 14 such scenarios. That is given to this data set gives you 14 different scenarios where some scenarios will result in going for fishing.   Some scenarios will result in not going for fishing.




So, then from this, the next one is what we are talking about is understanding the fishing decision.  Let us break it down, what is the "Fishing" decision?  So, the decision is, whether to go for fishing or not, that is the criteria.  The decision is whether you want to go for fishing or not.

The decision depends on the values of 4 variables or scenarios .You want to call it here, few  do not want to use the word variable. Four variables and which are those Four variables.  Number 1 is Weather, number 2 is Temperature, number 3 is Humidity, number 4 is Wind.  And, these four variables which determines, whether to go for fishing or not.

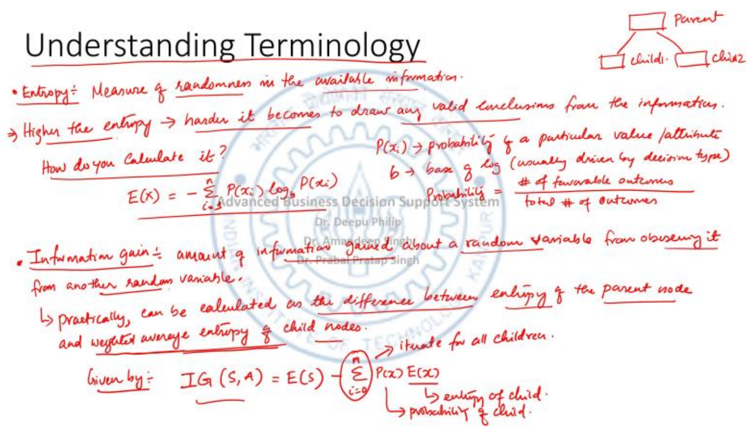
These are the independent variables.  So, they are independent because their values you have no control over and whatever be  these values of these four variables. It will determine whether you will go for fishing or not.  So, the decision depends upon these independent variables.  Hence, the dependent variable, what is the dependent variable here?  Dependent variable is whether to go for fishing or not.

So, that is the dependent variable.  So, this is the fishing scenario. That is the case, there are many algorithms are available to build Decision Tree.  So, please understand that, there are many algorithms available to build a Decision Tree.  There is no doubt about it.

There are so many of them available at this point.  So, what we use is here the ID3 algorithm when Dr. Prabal will teach you, he will teach you many other algorithms also. ID3 stands for Iterative Dichotomizer 3.  So, when say somebody says ID3, that is iterative dichotomizer 3 algorithm and ID3 uses entropy, this is a new word for you.

We will discuss what it is. Entropy and information gain is another new word for you, we will discuss in the next slide ask the matrix to create the tree. So, they use ID3 entropy and information gain ask the matrix to create the tree. It just does not create the tree randomly, it uses these 2 matrices to create the tree. So, you have seen that the there are four independent variables, here they are Weather, Temperature, Humidity and Wind and the dependent variable is whether to go for fishing or not. And, it is dependent because the decision whether to go for fishing or not depends upon the value of the independent variables.

And, then we are going to there is, since there are so many algorithms, but we are going to use the ID3 algorithm (Iterative Dichotomizer 3) which uses entropy and information gain to create the tree.




So, let us understand some of these terminologies So, first let us understand the terminology Entropy. I am not going to get into all the detailed definitions and other aspects of it. It is just a simple functional definition, what I am going to do it is a measure of randomness in the available information.

You are measuring the randomness in the available information. Everybody talks about randomness and entropy and that kind of a thing, the similar idea. So, the randomness means, if the entropy is 0, that means, you are definite about something but it is high. So, the idea is that, higher the entropy, what does that means? It becomes harder to draw any valid conclusions from the information. So, if the entropy is high or higher, it is larger value of entropy makes it harder to draw any conclusion from the information. So, you want the entropy to be minimal, you want the entropy to be close to 0.

So, how do you calculate it? How do you calculate the entropy? It is calculated by,

$$E(x) = -\sum_{i=1}^{n} P(x_i)\ log_b\ P(x_i)$$

Where,

*$P(x_i)$*- probability of a particular value/attribute.

*b*- base of *log* (usually driven by decision type).

So, I will show you how this equation gets applied and how the probability is and if you do not remember probability too much.

Probability is the simplest equation,

$$\frac{no.of\ favourable\ outcomes}{total\ no.of\ outcomes}$$

We will use this definition for the time being for this course or for in doing the Decision Tree. Now, comes the information gain. What is information gain? So, the information gain is the amount of information gained about a random variable from observing another random variable or observing it from. So, what it does is, how much of information you can gain about a particular random variable by observing it from another random variable.

So, from a view point of another random variable, you observe a different random variable and see how much information you can gain out of this. That is the theoretical definition. Practically can be calculated as the difference between entropy of the parent node and weighted average entropy of child nodes. So, we can be calculated as a difference between the entropy of the parent. So, parent node, if you think about it, here is a parent and this is the parent we talked about this as the child.

When we did the tree thing, so entropy of the parent node and the weighted average entropy of the child nodes. So, if there are 2 child nodes, child 1 and child 2, so then, the weighted average entropy of both the children and then the difference between the entropy of the parent will tell you, what it is.

So, given by the equation for this,

$$IG(S,A) = E(S) = -\sum_{i=0}^{n} P(x)\ E(x)$$

So, the weighted average, we said the word is weighted average. How do you weighted with the help of probability. $E(x)$ is the entropy of the child. $P(x)$ is the probability of the child. So, use the probability of the child to weigh the entropy and i equal to 0 to n is the iterate for all children.

So, that is the general idea. So, you iterate it you average it and sum it up and then subtract it from the parent, you will actually get the what you call as the Information Gain. So, we will speed up in the next one now, I hope that you understood the calculations and we will complete the tree in the next session. Thank you for your patience here.