Decision Tree

Entropy Calculation, Information Gain & Decision Tree Learning

Introduction:

Decision tree learning is a method for approximating discrete-valued target functions, in which the learned function is represented as sets of if-else/then rules to improve human readability. These learning methods are among the most popular of inductive inference algorithms and have been successfully applied to a broad range of tasks; from learning to diagnose medical cases, to assess credit risk of loan applicants. Most popular algorithm to build decision trees is ID3(Iterative Dichotomiser 3). Others are ASSISTANT and C4.5. These decision tree learning methods search a completely expressive hypothesis space (All possible hypotheses) and thus avoid the difficulties of restricted hypothesis spaces. Their inductive bias is a preference for small trees over longer tress.

When to use Decision Tree:

Remember, there are lots of classifiers to classify unseen instances based on the training examples. We have to understand by looking at the training examples which classifier will be the best for the dataset. Decision Tree is most effective if the problem characteristics look like the following points -

1) Instances can be described by attribute-value pairs.
2) Target function is discrete-valued.

Construction of Decision Tree:

Decision Trees classify instances by sorting them down the tree from root node to some leaf node. Each node specifies a test of some ***attribute*** of the instance, and each branch descending from that node corresponds to one of the possible values for this attribute.
Our basic algorithm ID3 learns decision trees by constructing them top-down, beginning with the question, "Which attribute should be tested at the root of the tree?" To answer this question, each attribute is evaluated using a statistical test to determine how well it alone classifies the training examples. The best attribute is selected as the root of the tree. Our next task is to find which node will be next after root. In this case, we would like to again choose the attribute which is most useful to classify training examples. Then repeat the process until we find leaf node.
Now the big question is, how do ID3 measures the most useful attributes. The answer is, ID3 uses a statistical property, called ***information gain*** that measures how well a given attribute separates the training examples according to their target classification. We will discuss in more detail about ***"information gain"*** once we get some knowledge about ***Entropy***} in section next section.

Day	Outlook	Temperature	Humidity	Wind	Play Tennis
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

Figure 1: Dataset of playing tennis, which will be used for training decision tree

Entropy:

To Define Information Gain precisely, we begin by defining a measure which is commonly used in information theory called Entropy. Entropy basically tells us how impure a collection of data is. The term impure here defines non-homogeneity. In other word we can say, "Entropy is the measurement of homogeneity. It returns us the information about an arbitrary dataset that how impure/non-homogeneous the data set is."
Given a collection of examples/dataset S, containing positive and negative examples of some target concept, the entropy of S relative to this boolean classification is-

$$Entropy(S) = -(P_{\oplus} log_2 P_{\oplus} + P_{\ominus} log_2 P_{\ominus}) \quad (1.1)$$

Where $P_{\oplus}$ is the portion of positive examples and $P_{\ominus}$ is the portion of negative examples in S.

To illustrate this equation, we will do an example that calculates the entropy of our data set in Fig: 1. The dataset has 9 positive instances and 5 negative instances, therefore-

$$Entropy([9+,5-]) = -(\frac{9}{14}log_2\frac{9}{14} + \frac{5}{14}log_2\frac{5}{14}) = 0.940 \qquad (1.2)$$

Which concludes, the dataset is 94% impure or 94% non-homogeneous.

Let's do some more calculations and try to understand the nature of *Entropy*.
What could be the Entropy of [7+,7-] & [14+,0-]?

$$Entropy[7+,7-] = -(\frac{7}{14}log_2\frac{7}{14} + \frac{7}{14}log_2\frac{7}{14}) = 1 \qquad (1.3)$$

And,

$$Entropy[14+,0-] = -(\frac{14}{14}log_2\frac{14}{14} + \frac{0}{14}log_2\frac{0}{14}) = 0 \qquad (1.4)$$

By observing closely on equations **1.2**, **1.3** and **1.4;** we can come to a conclusion that if the data set is completely homogeneous then the impurity is 0, therefore entropy is 0 (equation **1.4**), but if the data set can be equally divided into two classes, then it is completely non-homogeneous & impurity is 100%, therefore entropy is 1 (equation **1.3**).

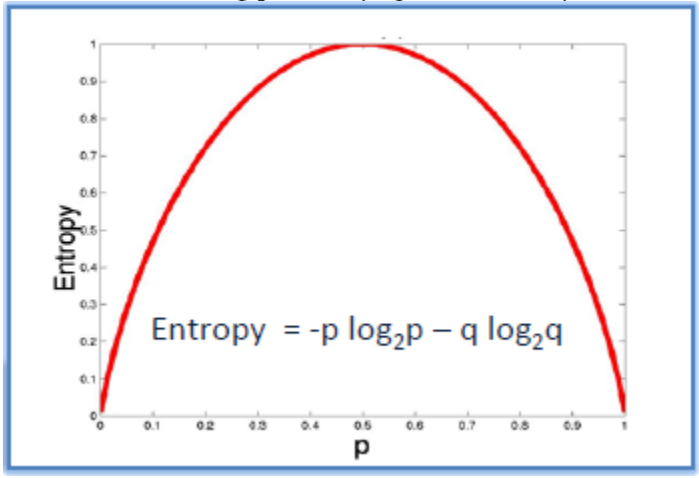

Figure 2: Entropy Graph

Now, if we try to plot the ***Entropy*** in a graph, it will look like Figure 2. It clearly shows that the Entropy is lowest when the data set is homogeneous and highest when the data set is completely non-homogeneous.

Information Gain:

Given Entropy is the measure of impurity in a collection of a dataset, now we can measure the effectiveness of an attribute in classifying the training set. The measure we will use called ***information gain***, is simply the expected reduction in ***entropy*** caused by partitioning the data set according to this attribute. The information gain ***(Gain(S,A)*** of an attribute A relative to a collection of data set S, is defined as-

$$Gain(S, A) = Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v) \qquad (1.5)$$

Where, $Values(A)$ is the all possible values for attribute A, and S_v is the subset of S for which attribute A has value v.

To become more clear, let's use this equation and measure the ***information gain*** of attribute ***Wind*** from the dataset of Figure 1. The dataset has 14 instances, so the sample space is 14 where the sample has 9 positive and 5 negative instances. The Attribute **Wind** can have the values **Weak** or **Strong**. Therefore,

Values(Wind) = Weak, Strong

$S = [9+, 5-]$
$S_{\text{weak}} = [6+, 2-]$
$S_{\text{strong}} = [3+, 3-]$
$Entropy(S) = 0.940$ from eqtn 1.2

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

$$Gain(S, Wind) = Entropy(S) - (\frac{8}{14} Entropy(S_{\text{weak}}) + \frac{6}{14} Entropy(S_{\text{strong}})) \tag{1.6}$$

$$Entropy(S_{\text{weak}}) = -(\frac{6}{8} log_2 \frac{6}{8} + \frac{2}{8} log_2 \frac{2}{8}) = 0.811 \tag{1.7}$$

$$Entropy(S_{\text{strong}}) = -(\frac{3}{3} log_2 \frac{3}{3} + \frac{3}{3} log_2 \frac{3}{3}) = 1.00 \tag{1.8}$$

Put the values of $Entropy(S_{\text{weak}})$ and $Entropy(S_{\text{strong}})$ in eqtn 1.6

$$\begin{aligned} Gain(S, Wind) &= Entropy(S) - (\frac{8}{14} 0.811 + \frac{6}{14} 1.00) \\ &= 0.940 - (0.463 + 0.429) \\ &= 0.048 \end{aligned} \tag{1.9}$$

So, the ***information gain*** by the **Wind** attribute is 0.048. Let's calculate the ***information gain*** by the **Outlook** attribute.

$$Values(Outlook) = Sunny, Overcast, Rain$$
$$S = [9+, 5-]$$
$$S_{\text{sunny}} = [2+, 3-]$$
$$S_{\text{overcast}} = [4+, 0-]$$
$$S_{\text{rain}} = [3+, 2-]$$

$$G(S, Outlook) = Entropy(S) - (\frac{5}{14}Entropy(S_{\text{sunny}}) + \frac{4}{14}Entropy(S_{\text{overcast}}) + \frac{5}{14}Entropy(S_{\text{rain}})) \tag{1.10}$$

$$Entropy(S_{\text{sunny}}) = -(\frac{2}{5}log_2\frac{2}{5} + \frac{3}{5}log_2\frac{3}{5}) = 0.971 \tag{1.11}$$

$$Entropy(S_{\text{overcast}}) = -(\frac{4}{4}log_2\frac{4}{4} + \frac{0}{4}log_2\frac{0}{4}) = 0 \tag{1.12}$$

$$Entropy(S_{\text{rain}}) = -(\frac{3}{5}log_2\frac{3}{5} + \frac{2}{5}log_2\frac{2}{5}) = 0.971 \tag{1.13}$$

Put the values of eqtn 1.11, 1.12, 1.13 in 1.10.

$$G(S, Outlook) = 0.940 - (\frac{5}{14}0.971 + \frac{4}{14}0 + \frac{5}{14}0.971) = 0.246 \tag{1.14}$$

These two examples should make us clear that how we can calculate ***information gain***. The information gain of the 4 attributes of Figure 1 dataset are:

$$Gain(S, Outlook) = 0.246$$
$$Gain(S, Humidity) = 0.151$$
$$Gain(S, Wind) = 0.048$$
$$Gain(S, Temperature) = 0.029$$

Remember, the main goal of measuring ***information gain*** is to find the attribute which is most useful to classify training set. Our ID3 algorithm will use the attribute as it's root to build the decision tree. Then it will again calculate information gain to find the next node. As far as we calculated, the most useful

attribute is “Outlook” as it is giving us more information than others. So, “Outlook” will be the root of our tree.

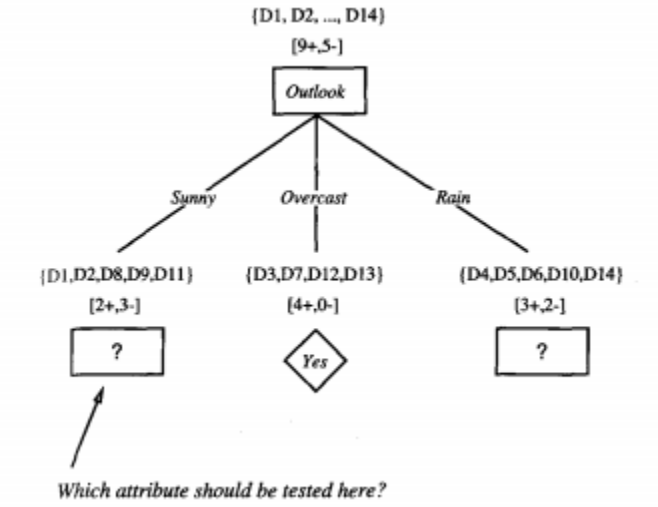

Figure 3: Partially learned Decision Tree from the first stage of ID3

Figure 3 visualizes our decision tree learned at the first stage of ID3. The training examples are sorted to the corresponding descendant nodes. The **Overcast** descendant has only positive instances and therefore becomes a leaf node with classification **Yes**. For other two nodes, the question again arises which attribute should be tested? These two nodes will be further expanded by selecting the attributes with the highest information gain **relative** to the new subset of examples. Let’s find the attribute that should be tested at the **Sunny** descendant.

The Dataset in Figure 1 has the value **Sunny** on Day1, Day2, Day8, Day9, Day11. So the Sample Space S=5 here.

$S_{\text{suuny}} = 5 = S$
$Humidity = High, Normal$
$Humidity_{\text{high}} = [0+, 3-]$
$Humidity_{\text{normal}} = [2+, 0-]$
$Gain(S, Humidity) = ?$

$$Gain(S_{\text{sunny}}, Humidity) = Entropy(S) - (\frac{3}{5}Entropy(Humidity_{\text{high}}) + \frac{2}{5}Humidity_{\text{normal}}) \tag{1.15}$$

$$Entropy(Humidity_{\text{high}}) = -(\frac{0}{3}log_2\frac{0}{3} + \frac{3}{3}log_2\frac{3}{3}) = 0 \tag{1.16}$$

$$Entropy(Humidity_{\text{normal}}) = -(\frac{2}{2}log_2\frac{2}{2} + \frac{0}{2}log_2\frac{0}{2}) = 0 \tag{1.17}$$

Put the values in eqtn 1.15

$$\begin{aligned} Gain(S_{\text{sunny}}, Humidity) &= Entropy(S) - (\frac{3}{5}0 + \frac{2}{5}0) \\ &= 0.970 - 0 \\ &= 0.970 \end{aligned} \tag{1.18}$$

We can now measure the information gain of Temperature and Wind by following the same way we measured **Gain(S, Humidity)**. Finally, we will get:

$Gain(S, Humidity) = 0.970$
$Gain(S, Temperature) = 0.570$
$Gain(S, Wind) = 0.019$

So Humidity gives us the most information at this stage. The node after **"Outlook"** at **Sunny** descendant will be **Humidity**. The **High** descendant has only negative examples and the **Normal** descendant has only positive examples. So both of them become the leaf node and can not be furthered expanded. If we expand the **Rain** descendant by the same procedure we will see that the **Wind** attribute is providing most information. I am leaving this portion for the readers to do the calculation on their own. Therefore our final decision tree looks like Figure 4:

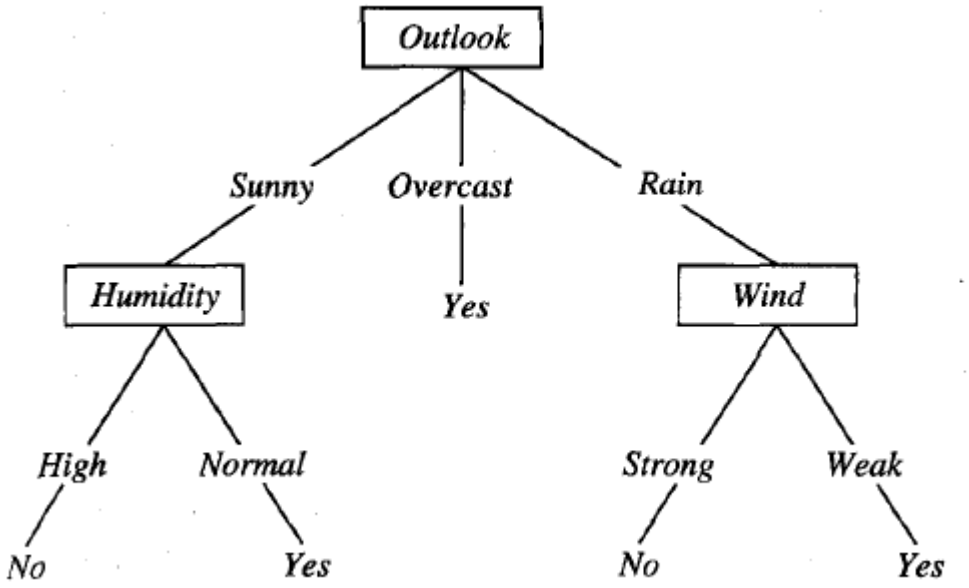

Figure 4: Fully learned Decision Tree by ID3 Algorithm