Decision Tree

Lecture 11

18/08/2021

Decision Tree

- As the word suggests, a decision tree produces a tree-like structure of decisions and hence assists in visualizing the possible scenarios with their consequences.
- It is a way of representing algorithm or thought process in a graphical format for better decision making.
- Construction of a decision tree involves a collection of *decision nodes*, connected by *branches* extending downward from the *root node.*

Requirements for using decision trees

- Decision tree algorithm represent supervised learning and as such require pre-classified target variables.
- The training data set should be rich and varied providing the algorithm with a healthy cross section of the types of records.
- Classification and Regression Tree (CART) is an umbrella term, we will call it a classification tree if the outcome variable value is discrete and a regression tree if the outcome variable value is continuous.

How does one measure uniformity/heterogeneity?

- Decision tree provides classification assignments with the highest measure of confidence available.
- Two leading algorithms for constructing decision trees.

➢ Classification and Regression Trees (CART)

➢ C5.0 Algorithm

Classification and Regression Trees (CART)

- The CART method was suggested by Breiman et al.(1984). The decision trees produced by CART are strictly binary , containing exactly two branches for each decision node.
- It is also used to predict the value of an outcome-variable.
- The tree starts with a root node consisting of complete data and thereafter uses intelligent strategies to split the nodes(parent node) into multiple branches (thus creating children node).
- The original data is divided into subsets in this process.

CART: Gini Index

- Classification Tree uses various impurity measures such as Gini impurity Index and Entropy to split the nodes. Regression Tree on the other hand , split the node that minimizes the Sum of Squared Errors (SSE).
- We look at each node in the tree and we can actually calculate impurity in that node.

$$Gini(k) = \sum_{j=1}^{J} P(j|k)[1 - P(j|k)]$$

- Where J= No. of classes in the data set, $P(j|k)$ is the proportion of category j in node k.
- Smaller Gini Index implies less impurity. A gini score of 0 is the most pure score possible.
- (In a general sense “purity” can be thought of as how homogenized a group is)

CART: Gini Index (contd…)

- If there are only two classes,(say 0 and 1), consider the node label k with 10, 1s and 90, 0s. Then

$$Gini(k) = \sum_{j=1}^{2} P(j|k)[1 - P(j|k)]$$

$$\text{Gini(k)} = 2P(j|k)[1 - P(j|k)] = 2 * 0.1 * 0.9 = 0.18.$$

- Consider the node label k with 50, 1s and 50, 0s. Then

$$\text{Gini(k)} = 2P(j|k)[1 - P(j|k)] = 2 * 0.5 * 0.5 = 0.50.$$

- Minimum value will happen when one of those classes is zero in that particular node.
- So the Gini Index value will lie between zero to 0.5.

Entropy (Impurity Measure)

- Entropy is another impurity measure that is frequently used.
- Entropy at node k is given by

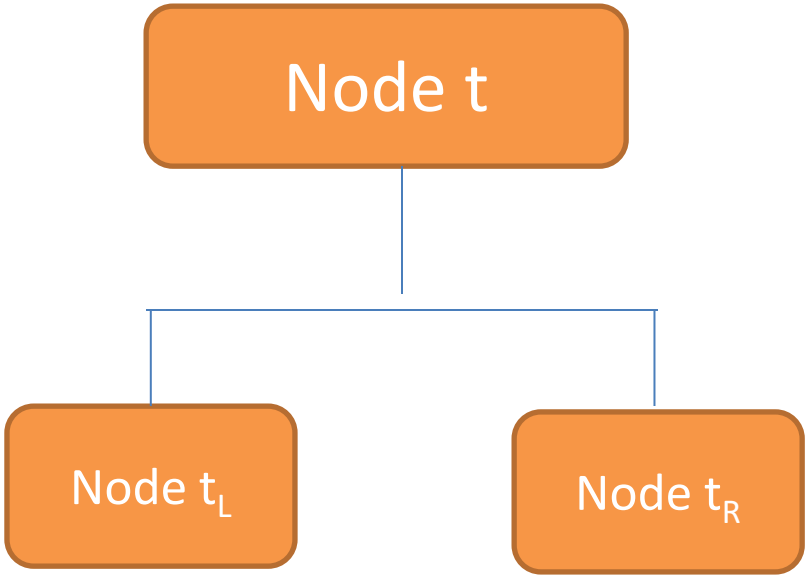
$$Entropy(k) = -\sum_{j=1}^{J} P(j|k) log_2 P(j|k).$$

- If there are only two classes,(say 0 and 1), consider the node label k with 10, 1s and 90, 0s.

$$Entropy(k) = -\sum_{j=1}^{2} P(j|k) log_2 P(j|k)$$

$$= -0.1 * log_2(0.1) - 0.9 * log_2(0.9) = 0.4689.$$

Classification Tree Logic
Calculation of change in Impurity



$$Max[i(t) - P_L * i(t_L) - P_R * i(t_R)].$$

Where $i(.) = \text{Impurity at node } (.)$

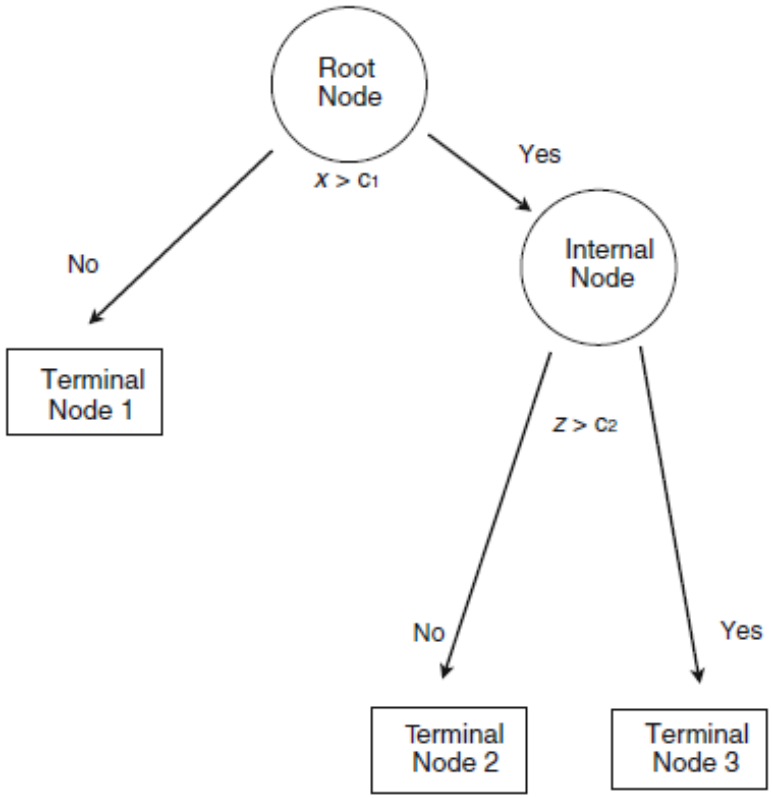
P_L : Proportion of observations in the left node

P_R : Proportion of observations in the right node.

A simple CART tree structure

[Source : Statistical learning from a regression perspective (Second Edn.) by Richard A Berk]

Fig. 3.2 A simple CART tree structure



C5.0 Algorithm

- C5.0 algorithm is an extension of C4.5 algorithm for generating decision tree.
- Difference between CART and C5.0 algorithm

➢ Unlike CART, C5.0 algorithm is not restricted to binary split.

➢ For categorical variables, C5.0 by default produces a separate branch for each value of the categorical attribute.

C5.0 Algorithm (contd…)

- It uses the concept of information gain or entropy reduction to select the optimal split.
- Suppose that we have a variable X whose k possible values have probabilities $p_1, p_2, \ldots p_k$.
- The entropy of X is defined as

$$H(X) = -\sum_j p_j \log_2 (p_j)$$

C5.0 Pruned

- If the tree is too big, the lower branches are modeling noise in the data ("overfitting").
- The usual paradigm is to grow the trees large and "prune" back unnecessary splits.
- One limitation of decision trees is that the division of input space is based on hard splits in which only one model is responsible for making predictions for any given value of the input variables