

An Approach to Decision Tree Induction for Classification

Prof. Shabana S.Pathan

Assistant Professor
Dept. of IT

Article History: Received: 11 January 2021; Revised: 12 February 2021; Accepted: 27 March 2021; Published online: 23 May 2021

Abstract: One of the most easiest to understand learning algorithms is Decision tree Induction. For both classification and regression forms of problems, the decision tree may be used. It is possible to use these algorithms to create models that learn information from previous data. To solve problems related to classification, regression, clustering and optimization, algorithms like Decision Trees (ID3, C4.5, C5.0, and CART), Support Vector Machine (SVM), Neural Networks, Linear Regression, K-Nearest Neighbor (KNN) and so on can be used. The accuracy of the forecast is very critical. All the algorithms are explained one by one in this paper and are compared on the basis of efficiency & precision. The various parameters of Decision tree Induction and prediction efficiency metrics are discussed.

Keywords: Decision tree, Classification, Induction

1. Introduction

Machine learning focuses on features that learn from experience over time and strengthen their decision-making or predictive accuracy. A machine learning technique used to group related data objects together is the classification of data objects. Many classification algorithms are present in the literature, but, due to its simplicity of usage, the decision tree is most usually utilized and more clearly contrasted with other classification algorithms.

In order to predict future data patterns, Classification and Prediction are used to analysis of data that can generate critical data models. Such an analysis will help to give us a clearer understanding of the entire data. Classification predicts categorical names like discrete, unordered whereas prediction models predicts continuous names.

In machine learning, pattern recognition and statistics, researchers have proposed many classification and inference methods. Most algorithms, usually assuming a small data size, are memory-resident. The cases are placed into distinct categories in the classification. For preparation & testing of test cases classification create rules.

For classification-based machine learning, a range of algorithms that include a decision tree, k-Nearest Neighbor, Bayesian and Neural-Net dependent classifiers have been developed [3].

Among all, the decision tree is easier and wide employed in data discovery and also in pattern recognition. It is created by splitting criterion in accordance with a training dataset that is recursively divided into two or more root node. This algorithmic procedure is repeatedly done until a termination condition is met. Similar to classification accuracy in many cases, decision trees have several advantages and so, decision trees are still progressively used for varied tasks, such as privacy protection, biology, intrusion detection, drugs and healthcare systems [5].

Several decision tree induction algorithms vary in how the most critical characteristics for construction of decision tree are represented, and how they deal with numerical or, symbolic characteristics, also noise can be reduced using pruning techniques [9].

2. Literature Review

Three existing algorithms in decision tree (ID3, C4.5, and CART) have been applied to the educational data to predict the performance of the student in the test, proposed by Anuja Priyam (2013). To predict their progress all algorithms are applied to the internal assessment of students that enabled the teacher to understand and develop the performance of poor students [1].

Bouchra Lamrini proposed the study criteria and pruning algorithms to control complexity and maximize the efficiency of the decision tree [2]. Mohd Mahmood Ali (2009) suggested a simple approach to classification using Decision Tree Induction showing how the algorithm generalizes the hierarchies using attribute-oriented induction algorithm from the raw training datasets [7].

In a decision tree induction algorithm, Nittaya Kerdprasop (2011) provides a technique to handle the noise. Noise handling function is a different step than induction of the tree. Prior to implementing the tree induction module, all corrupted and uncorrupted data is clustered and heuristically selected [9].

Neha Patel proposed a classification algorithm for learning decision tree classifiers from data with less complexity and less time to construct the decision tree using dissimilarities functions [8].

Chakaravarthy et al. (2011) stated the substance of building decision trees for recognition of the entity using comprehensive setup where multi-value attributes and the possibilities relate to the entities. A natural greedy algorithm considered in this paper to prove an estimate ratio about Ramsey numbers, the problems involving arbitrary input tables and possibility circulations are studied over the entity collection [13].

Barros et al. (2012) [11] proposed an automated type of decision-tree induction algorithms for binding information sets for drug-enzyme. He also examines the efficiency of the latest methodology conformance of varied drug candidates to InhA, and analyzes results for precision, comprehensibility, and biotic validity of the decision tree.

Liu and Tsang (2017) [14] introduced for the perceptron decision tree (PDT) a single data-dependent generalization error, which provides the theoretical basis for telling and pruning a thin linear hyperplane in each call node. To control the generalization error of decision trees, a popular strategy is the combination of bagging and building ensembles of trees. Through integrating predictions of several different trees such as random forest selection (RFFS) and Gradient boosted feature selection (GBFS) to achieve improved accuracy.

Petra Perner (2010) assessed the feature preselection on the prediction of C4.5 using a real-world data set. The precision of the C4.5 classifier could be improved with an appropriate feature preselection for the learning algorithm [10].

3. Decision Tree Induction

3.1 Decision Tree

Decision tree is widely used supervised learning method in many real-world implementations and can handle higher precision numerical, categorical, and continuous data. The goal of decision tree is to identify the class labels of new/unknown instances with known input values, but unknown classes accurately. A decision tree algorithm iteratively divides the specified training dataset into smaller sub-datasets until all instances in a subset satisfy only one class or the terminating level of the tree. Noisy datasets can be handled by a decision tree also unusual interconnections between input characteristics do not affect the tree's output. [6].

The decision tree consists of nodes with no incoming edges with a node that forms a rooted tree called the root. An internal node is the outgoing edges while other nodes are leaves. Each internal node divides the datasets space into two or more subspaces based on some unique feature of the input attribute values. Instances are categorized according to the results of the tests along the route by navigating them down to a leaf from the root of the tree.[3]

Decision and Leaf Node are two nodes in the Decision Tree where Decision node have several branches used to make any decisions, although no other branches are used by the Leaf nodes for the production of those decisions. Decisions or analyses are taken on the basis of the data collection features[16].

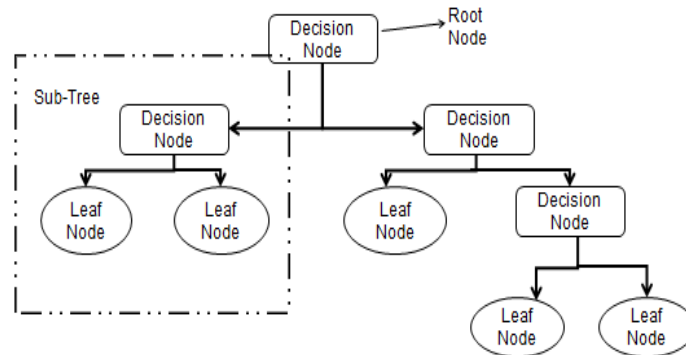


Figure 1. Structure of Decision Tree

3.2 Attribute Selection Measures

Attribute selection measure is used to identify the best attribute for the root node and sub-nodes. We can easily select the best attribute for the tree nodes through this calculation. The techniques for Attribute Selection Measure, are:

Information Gain

Gini Index

Information Gain: It is the calculation of changes in entropy after segmenting a dataset based on an attribute. It describes how much information a function gives us about a class. We split the node and build the decision tree, according to the significance of data benefit. A decision tree algorithm also seeks to optimize the benefit of knowledge acquisition and first splits the node/attribute with the highest gain of information. It can be calculated using the following formula:

$$\text{Information Gain} = \text{Entropy}(S) - [(\text{Weighted Avg}) * \text{Entropy}(\text{each feature})]$$

Entropy: For a given attribute, entropy is a metric for measuring impurity. It also determines randomness in the results. You are able to calculate entropy as:

$$\text{Entropy}(s) = -P(\text{yes}) \log_2 P(\text{yes}) - P(\text{no}) \log_2 P(\text{no})$$

Where,

S= Total number of samples

P(yes)= probability of yes

P(no)= probability of no

The Gini index is an impurity or purity metric while constructing a decision tree. In contrast to the high Gini index, a low Gini index attribute should be preferred. You can calculate the Gini index using the formula below:

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

3.3 Working of Decision Tree

In a decision tree, in order to predict the class for the given dataset, the set of rules will start advancing from the root hub of the tree. It starts from the branch and moves to the node by comparing the root attribute values with the reference attribute. Set of rules compares the value of the attribute once again with the opposite sub-nodes for the following node and forwards them. It preserves the phase until it exceeds the tree's leaf node. Performance can be the full methodology understood by the use of the measures below:

1. Use the root node to start the tree containing the entire dataset.
2. Identify a quality attribute within the dataset
3. Split root node into different subsets to integrate the values of attribute.
4. The attribute of quality is created.
5. Recursively use the subsets of the dataset created in step-3 to construct new decision trees. Continue this method until you achieve a stage where it is miles of difficulty to further classify the nodes and mark a leaf node as the very last node.

Table 1. Dataset for Buys_laptop

Example	age	income	person	Loan_rating	Buys_laptop
1	young	high	no	fair	no
2	young	high	no	good	no
3	middle_age	high	no	fair	yes
4	senior	medium	no	fair	yes
5	senior	low	yes	fair	yes
6	senior	low	yes	good	no
7	middle_age	low	yes	good	yes
8	young	medium	no	fair	no
9	young	low	yes	fair	yes
10	senior	medium	yes	fair	yes
11	young	medium	yes	good	yes
12	middle_age	medium	no	good	yes
13	middle_age	high	yes	fair	yes

14	senior	mediu m	no	good	no
----	--------	------------	----	------	----

Start with the attribute age

$$\begin{aligned} \text{Gain (age, D)} &= \text{Entropy}(D) - \sum_{(young, Middle_age, senior)} \frac{|Sv|}{|S|} \text{Entropy}(Sv) \\ &= \text{Entropy}(D) - \frac{5}{14} \text{Entropy}(S_{young}) - \frac{4}{14} \text{Entropy}(S_{middle_age}) - \frac{4}{14} \text{Entropy}(S_{middle_age}) \\ &= 0.246 \end{aligned}$$

Gain (Income, D) = 0.029

Gain (person, D) = 0.151

Gain (loan_rating, D) = 0.048

Step by step Calculations:

Step 1: “example” set D

The set D of 14 examples with 9 yes and 5 no then

$$\text{Entropy}(D) = - (9/14) \log_2 (9 /14) - (5/14) \log_2 (5/14) = 0.940$$

Step 2: Attribute Income

Income value can be high, medium, and low

Income = high is of occurrence 4

Income = medium is of occurrence 6

Income = low is of occurrence 4

Income = high, 2 of the examples are “yes” and 2 are “no”

Income = medium, 4 of the examples are “yes” and 2 are “no”

Income = low, 3 of the examples are “yes” and 1 are “no”

$$\text{Entropy}(D_{high}) = - (2/4) \times \log_2 (2/4) - (2/4) \times \log_2 (2/4) = 1$$

$$\text{Entropy}(D_{medium}) = - (4/6) \times \log_2 (4/6) - (2/6) \times \log_2 (2/6) = 0.9173$$

$$\text{Entropy}(D_{low}) = - (3/4) \times \log_2 (3/4) - (1/4) \times \log_2 (1/4) = 0.8112$$

$$\text{Gain (Income,D)} = \text{Entropy}(D) - (4/14) \times \text{Entropy}(D_{high}) - (6/14) \times \text{Entropy}(D_{medium}) - (4/14) \times \text{Entropy}(D_{low})$$

$$= 0.940 - (4/14) \times 1 - (6/14) \times 0.9173 - (4/14) \times 0.8112$$

$$= 0.0293$$

Similarly Gain (person, D) and Gain (loan_rating, D) can be calculated.

Fig. 2 is the representation of the decision tree from the Buys_laptop training dataset.

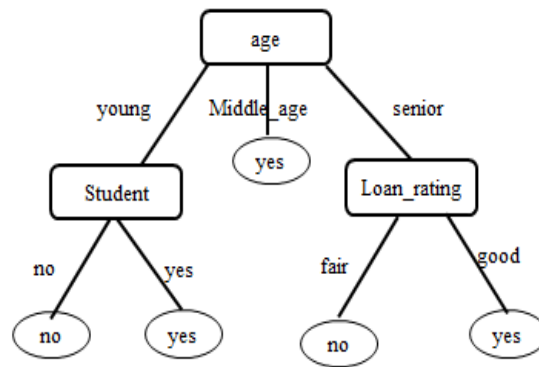


Figure 2. Decision tree for concept buys_laptop

3.4 Confusion Matrix

A matrix of uncertainty is with two dimensions “Actual” and ”Predicted” in addition, "True Positives (TP)", "True Negatives (TN)", "False Positives (FP)" and "False Negatives (FN)" have both dimensions as given in Fig 3.

		Actual Class	
		True	False
Predicted Class	Positive	TP	FP
	Negative	TN	FN

Figure 3. Confusion Matrix

$$\begin{aligned}
 \text{Accuracy} &= \frac{\text{Number of correct predictions}}{\text{Total number of Predictions}} \\
 &= \frac{TP+TN}{TP+TN+FP+FN}
 \end{aligned}$$

$$\begin{aligned}
 \text{Misclassification} &= \frac{\text{Number of wrong predictions}}{\text{Total number of Predictions}} \\
 &= \frac{FP+FN}{TP+TN+FP+FN}
 \end{aligned}$$

$$\begin{aligned}
 \text{Precision} &= \frac{\text{True Positive predictions}}{\text{Predicted Positive}} \\
 &= \frac{TP}{TP+FP}
 \end{aligned}$$

$$\begin{aligned}
 \text{Recall} &= \frac{\text{True Positive predictions}}{\text{Actual Positive}} \\
 &= \frac{TP}{TP+FN}
 \end{aligned}$$

$$\begin{aligned}
 \text{Specificity} &= \frac{\text{True Negative predictions}}{\text{Actual Negative}} \\
 &= \frac{TN}{TN+FP}
 \end{aligned}$$

$$f_{measure} = 2 \times \frac{precision \times recall}{precision + recall}$$

3.5 Decision Tree Algorithms

Over a time span, researchers have built different decision tree algorithms with performance improvement and the ability to handle different dataset. Some algorithms are mentioned below:

CART: Breiman's suggested Classification and Regression Tree applies gini index for attribute selection as an impurity indicator. The attribute containing less impurity is been used for dividing the node's results. It also combines knowledge of continuous or categorical values and thus deals with missing values. CART has enhanced features and capabilities that overcome CART's limitations, resulting in a high classification and more precision for prediction.

ID3:Quinlan Ross designed the decision tree algorithm for ID3 (Iterative Dichotomiser 3). The data gain approach is typically used in the decision tree process to decide the appropriate property for each node of the decision tree created. To find an optimal way to define a learning set some feature that calculate to provide the most balanced division is required called the information gain metric. ID3 uses the calculation of information gain to pick the splitting attribute. In constructing a tree model, it only accepts categorical attributes. When there is noise, it does not provide precise results and it is applied serially. Thus, before constructing a decision tree model with ID3, extensive pre-processing of data is carried out.

C4.5: C4.5 is an enhancement of the earlier ID3 algorithm by Quinlan. The algorithm C4.5 uses data gain as a splitting criterion. Data with categorical or discrete values may be accepted. It produces thresholds values greater or equal to threshold, to retain continuous values. Due to the missing attribute not used by C4.5, it can handle missing values easily.

C5.0: The C5.0 algorithm, which is also an ID3 extension, is a C4.5 algorithm extension. Based on the largest data gain area, the C5.0 model would break up samples. It will then divide the sample subset created from the former split. Until it is not possible to split the sample subset, the method will proceed. Finally, looking at the lowest level split, all sample subsets that do not make a major contribution to the model will be rejected[8].

Table 2. Parameter Comparison of Decision tree algorithm

Algorithms	Type of Data	Speed	Pruning	Missing Values	Measure	Procedure
ID3	Categorical	Low	No	Can't deal with	Entropy info-gain	Top-down decision tree construction
C4.5	Continuous and Categorical	Faster than ID3	Pre-Pruning	Can't deal with	Entropy info-gain	Top-down decision tree construction
C5.0	Continuous and Categorical, dates, times, timestamps	High	Pre-Pruning	Can deal with	Entropy info-gain	Top-down decision tree construction
CART	Continuous and nominal attributes data	Average	Post pruning	Can deal with	Gini diversity index	Constructs binary Decision tree

Table 3. Comparison of different Decision tree algorithm[18]

Algorithms	Author	Pruning Techniques	Methods to select node	Tree coverage approach	Drawbacks
ID3	Ross Quinlan in 1986	Simple pruning	Information gain	Depth first/divide and conquer	-unable to handle numeric problems -low accuracy of classification for large datasets -not scalable
C4.5	Ross Quinlan	Error based pruning	Gain ratio	Depth first/divide and conquer	-requires skills for better understanding -lack the means to partial automatic learning -memory dependant and not successful for large data sets
CART	Breiman in 1984	cost complexity pruning	Single variable and multivariable	Depth first/divide and conquer	-memory resident and not suitable for large datasets -perform sorting at each and every node
SLIQ	Riassnen et al	MDL principle	Gini index	Breadth first/divide and conquer	-attribute list is memory resident -successful for serial implementation only and not applied on node parallel machines.

3.5 Feature Selection

When designing a machine learning solution, selecting characteristics is an extremely significant task. Depending on the field expertise and machine learning approach, the selection of features can be done manually or by using automated tests to assess and pick the most appropriate model. In many cases, using the most common value or median value in the training set, a classical imputation method is used to identify missing values in datasets. When we substitute missing values using knowledge for other analytical purposes modification is required in the given problem .Luckily, the decision tree requires less user preprocessing of data and generally used with missing data, and there is no requirement for normalization of features. However, in the manner in which we define the categorical data, we must be careful [7].Fig 4 & Fig.5 shows two methods of Feature selection.

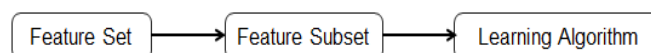


Figure 4. Filter Approach for Feature Selection

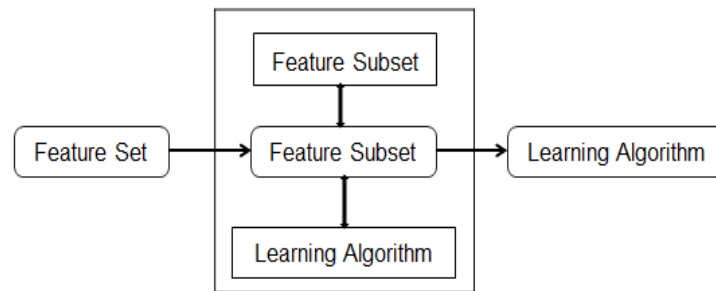


Figure 5. Wrapper Approach for Feature Selection

Conclusion

For scientists, research into machine learning algorithms has tremendous possibilities. It is an important method used in large data sets to identify new patterns and relationships. The most used algorithms for classification have been decision tree algorithms such as ID3 and C4.5 over the years. To this end, many researchers have tried to improve their performance in order to get better predictions and to keep up to date with constantly changing data. Depending on accuracy and time taken, the efficiency of different decision tree algorithms can be evaluated. The efficiency of algorithms also highly depends on entropy, data gain and data set characteristics. Also it is important to find a better algorithm for improving prediction performance.

References

1. Anuja Priyam , Abhijeet , Rahul Gupta , Anju Rathee & Saurabh Srivastava (2013). Comparative Analysis of Decision Tree Classification Algorithms. *International Journal of Current Engineering and Technology*, Vol.3(2), June, Available: [http:// inpressco .com/ category/ ijcet](http://inpressco.com/category/ijcet).
2. Bouchra Lamrini (2020). *Contribution to Decision Tree Induction with Python: A review*. Data Mining – Methods, Applications and Systems,(pp.1-21), intechopen.com.
3. Brijain R Patel & Kushik K Rana (2014).A Survey on Decision Tree Algorithm For classification”, *International Journal of Engineering Development and Research*, Vol 2(1).
4. Ibomoiye Domor Mienye , Yanxia Sun & Zenghui Wang (2019). Prediction performance of improved decision tree-based algorithms: a review. *Procedia Manufacturing* 35: 698–703, ELSEVIER.
5. J.Yan, Z.Zhang, L.Xie & Z. Zhu (2019). A Unified Framework for Decision Tree on Continuous Attributes. *IEEE Access*, Vol.7.
6. Lior Rokach & Oded Maimon (2005). Top-Down Induction of Decision Trees Classifiers-A Survey. *IEEE Transactions on Systems, Man, and Cybernetics-Part C: Applications and Reviews*, Vol.35, No.4, November
7. Mohd Mahmood Ali & Lakshmi Rajamani (2014). Decision Tree Induction: Data Classification using Height-Balanced Tree, *International Conference of Information and Knowledge Engineering*, January.
8. Neha Patel & Divakar Singh (2015).An Algorithm to Construct Decision Tree for Machine Learning based on Similarity Factor, *International Journal of Computer Applications* (0975 – 8887),Vol.111,No.10, February.
9. Nittaya Kerdprasop & Kittisak Kerdprasop (2011). A Heuristic-Based Decision Tree Induction Method for Noisy Data, 1–10, *Springer-Verlag Berlin Heidelberg*.
10. Petra Perner (2010). Improving the accuracy of decision tree induction by feature preselection, *Applied Artificial Intelligence: An International Journal*, 15:747-760, November.
11. R.Barros et.al (2012). Automatic design of decision-tree induction algorithms tailored to flexible-receptor docking data, *BMC Bioinformatics*.

12. Shabnam Sabah, Sara Zumerrah Binte Anwar, Sadia Afroze, Md. Abulkalam Azad, Swakkhar Shatabda, & Dewan Md. Farid (2019). Big Data with Decision Tree Induction, *13th International Conference on Software, Knowledge, Information Management and Applications, August*.
13. V. Chakaravarty, S.Roy, P.Awasthi & M.Mohania (2011). Decision Trees for Entity Identification: Approximation Algorithms and Hardness Results, *ACM Transactions on Algorithms, Vol.17, No.2, March*.
14. W.Liu & I.Tsang (2017). Making Decision Trees Feasible in Ultrahigh Feature and Label Dimensions, *Journal of Machine Learning Research, Vol.18, 1-36*.
15. Mehrdad Jeihouni, Ara Toomanian & Ali Mansourian (2020). Decision Tree-Based Data Mining and Rule Induction for Identifying High Quality Groundwater Zones to Water Supply Management: a Novel Hybrid Use of Data Mining and GIS, *Water Resources Management*, <https://doi.org/10.1007/s11269-019-02447-w>.
16. Priyanka & Dharmender Kumar (2020). Decision tree classifier: a detailed survey, *International Journal Information and Decision Sciences, Vol.12, No.3*.
17. Kotsiantis, S. B (2011). Decision trees: a recent overview, *Artificial Intelligence Review, 39(4), 261–283*. doi:10.1007/s10462-011-9272-4.
18. Anuradha & Dr.Gaurav Gupta (2014). A Self Explanatory Review of Decision Tree Classifiers, *IEEE International Conference on Recent Advances and Innovations in Engineering (ICRAIE-2014), May*.