A Decision Tree Based Classifier for Classification & Prediction of Diseases

Bhupendra Patidar¹ Prof. Gajendra Singh²
¹,²SSSIST, Sehore (MP)

Abstract-- In this paper, we are proposing a modified algorithm for classification. This algorithm is based on the concept of the decision trees. The proposed algorithm is better then the previous algorithms. It provides more accurate results. We have tested the proposed method on the example of patient data set. Our proposed methodology uses greedy approach to select the best attribute. To do so the information gain is used. The attribute with highest information gain is selected. If information gain is not good then again divide attributes values into groups. These steps are done until we get good classification/misclassification ratio. The proposed algorithms classify the data sets more accurately and efficiently.

I. INTRODUCTION

Decision trees: - The well-known machine learning techniques, A decision tree is composed of three basic elements:

1. A decision node specifying a test attributes.
2. An edge or a branch corresponding to the one of the possible attribute values which means one of the test attribute outcomes.
3. A leaf which is also named an answer node contains the class to which the object belongs.

In decision trees, two major phases should be ensured:

1. Building the tree: Based on a given training set, a decision tree is built. It consists of selecting for each decision node the ‘Appropriate ‘test attribute and also to define the class labelling each leaf.
2. Classification: In order to classify a new instance, we start by the root of the decision tree, then we test the attribute specified by this node. The result of this test allows moving down the tree branch relative to the attribute value of the given instance. This process will be repeated until a leaf is encountered. The instance is then being classified in the same class as the one characterizing the reached leaf.

Decision trees have also been used for intrusion detection [3]. The decision trees select the best features for each decision node during the construction of the tree based on some well-defined criteria. One such criterion is to use the information gain ratio. [2]

II. RELATED WORK

Naïve bays classifier is also a very good and accurate method for the data classification. Naïve Bayes classifier [17] is a probabilistic classifier based on the Bayes theorem, considering strong (Naïve) independence assumption. Thus, a Naïve Bayes classifier believes that all attributes (features) independently contribute to the probability of a certain decision. Considering the characteristics of the underlying probability model, the Naïve Bayes classifier can be trained very efficiently in a supervised learning setting. This could yield much better results in many complex real-world situations, especially in the field of computer-aided diagnosis [16] [17]. Here it is assumed that all variables are independent. Hence only the variances of the variables for each class need to be determined and not the entire covariance matrix.

The RnD tree is a modern method for data classification. Accurate results provided by this method are also attracting so many researchers to this method. The RnD tree [18] algorithm can be applied to both classification and regression problems. Random trees are a collection or assembly of tree predictors that is called forest [18]. The classification works as follows: the random trees classifier takes the input feature vector, classifies it with every tree in the forest, and outputs the class label that received the majority of “votes”. In the case of regression the classifier response is the average of the responses over all the trees in the forest.

A recursive Bayesian classifier is introduced in [7]. Lots of improvement is already done on decision tree induction method for 100 % accuracy and many of them achieved the goal also but main problem on these improved methods is that they required lots of time and complex extracted rules. The main idea is to split the data recursively into partitions where the conditional independence assumption holds. A decision tree is a mapping from observations about an item to conclusions about its target value [9, 10, 11,12 and 13]. Decision trees are commonly used in operations research, specifically in decision analysis, to help identify a strategy most likely to reach a goal. Another use of decision trees is as a descriptive means for calculating conditional probabilities. A decision tree (or tree diagram) is a decision support tool that uses a tree-like graph or model of decisions and their possible consequences, including chance event outcomes, resource costs, and utility [14]. Decision tree Induction Method has been successfully used in expert systems in capturing knowledge. Decision tree induction Method is good for multiple attribute Data sets.

III. PROPOSED SOLUTION

[18]Classification is a form of data analysis that extracts models describing important data classes. These models also called as classifiers are used to predict categorical (discrete, unordered)class labels. This analysis can help us for better understanding of large data. Classification has numerous applications, including fraud detection, target marketing,
performance prediction, manufacturing, credit risk and medical diagnosis. Data Classification is a two-step process. They are: Learning Step and Classification Step

A. Learning Step:
In this step classification model is constructed. A classifier is built describing a predetermined set of data classes or concepts. In learning step or training phase, where classification algorithm builds the classifier by analyzing or “learning from” a training set made up of database tuples and their associated class labels.

This step is also known as supervised learning as the class label of each training tuple is provided. This learning of the classifier is “supervised” by telling to which class each training tuple belongs. In unsupervised learning or clustering, the class label of each training tuple is not known, and the number or set of classes to be learned may not be known in advance.

B. Classification Step:
In this step, the model is used to predict class labels for given data and it is used for classification. First, the predictive accuracy of the classifier is estimated. To measure the classifiers accuracy, if we use the training set it would be optimistic, because the classifier tends to over fit the data i.e., during learning it may incorporate some particular anomalies of the training data that are not present in the general data set. Therefore, a test set is used, made up of the test tuples and their associated class labels., They are independent of the training tuples, from which the classifier cannot be constructed. The accuracy of a classifier on a given test set is the percentage of test tuples that are correctly classified by the classifier. The associated class label of each test tuple is compared with the learned classifier’s class prediction for the tuple. If the accuracy of the model or classifier is considered acceptable, the model can be used to classify future data tuples or objects for which the class label is not known.

C. Decision Tree Induction:
A decision tree is a flow-chart-like tree structure, where each internal node denotes a test on an attribute, each branch represents an outcome of the test, and leaf nodes represent classes or class distributions. The topmost node in a tree is the root node.

Given a tuple K, for which the associated class label is unknown, the attribute values of the tuple are tested against the decision tree. A path is traced from the root to a leaf node, which holds the class predicate for that tuple. Decision trees are easily converted to classification rules. The construction of decision does not require any domain knowledge or parameter setting. It can handle high dimension data. The learning and classification steps are simple and fast. It has good accuracy. Decision tree Induction algorithm can be used in many applications like medicine, manufacturing and production etc.

IV. PROPOSED METHOD:
DTC (in T: table; C: classification attribute) return decision tree

{ N: = a new node; if (there are no predictive attributes in T) /* Base case 1 */

Then label N with most common value of C in T (deterministic tree) or with frequencies of C in T (probabilistic tree)

else if (all instances in T have the same value V of C) /* Base case 2 */

then label N, “X.C=V with probability 1”

else { for each attribute A in T compute AVG ENTROPY(A,C,T); AS := the attribute for which AVG ENTROPY(AS,C,T) is minimal; if (AVG ENTROPY(AS,C,T) is not substantially smaller than ENTROPY(C,T)) /* Base case 3 */

then label N with most common value of C in T (deterministic tree) or with frequencies of C in T (probabilistic tree).

Else {label N with AS; for each value V of AS do {

return N;

N1:= DTC (SUBTABLE (T,A,V),C) /* Recursive call */ if (N1 != null) then make an arc from N to N1 labeled V; }

SUBTABLE (in T : table; A : predictive attribute; V : value) return table; { T1 := the set of instance X in T such that X.A = V; }

T1 := delete column A from T1; return T1 /* Note: in the textbook this is called I(p(y = v . . . p(y)) /* By convention, we consider 0 · log 0 to be 0. */ */

ENTROPY (in C : classification attribute; T : table) return real number; { for each value V of C, let p(V) := FREQUENCY(C,V,T);

Return Vp(V) log2k (p(V)) /* By convention, we consider 0 · log 0 to be 0. */ */

FREQUENCY (C,V,T) */

ENTROPY(A,C,T) is not substantially smaller than

ENTROPY(C,T) */

ADD (in AS: attribute; V: value; AS1 : set of attribute values; AS2 : set of attribute values) return (AS1 union AS2)

Return real number; { for each value V of AS, let p(V) := FREQUENCY(AS,V,T);

Return Vp(V) log2k (p(V)) /* By convention, we consider 0 · log 0 to be 0. */ */

FREQUENCY (in AS : set of attribute values; V : value; T : table) return real number; { return #! X in T [X.A=V ] / size(T); }

V. CONCLUSION
In this paper, we have proposed a modified algorithm for classification. This algorithm is based on the concept of the decision trees. The proposed algorithm is better then the previous algorithms. It provides more accurate results. We have tested the proposed method on the example of patient data set.

REFERENCES