J. Ind. Soc. Agril. Statist.
61(2), 2007 : 192-198

Prediction of Forest Cover using Decision Trees

B. Chandra and Pallath Paul V.[1]
Indian Institute of Technology, New Delhi

SUMMARY

Information regarding forest land is highly required for developing ecosystem management strategies which will facilitate the decision-making process. It is often difficult to get the relevant data for forest land that are outside the immediate jurisdiction of the concerned authorities. One of the approaches for obtaining this information is through the use of predictive models like Decision Trees and Neural Networks. (Blackard *et al.* 2000) have shown that Neural Network approach outperforms the traditional discriminant analysis method in predicting forest cover types. The accuracy achieved by Neural Network was 70.58%. Decision Trees algorithms have been proposed in the past for classification of numeric as well as categorical attributes. SLIQ algorithm was proposed (Mehta *et al.* 1996) as an improvement over ID3 and C4.5 algorithms (Quinlan 1993). Robust algorithm for Decision Tree Classification was proposed (Chandra *et al.* 2006) as improvement over SLIQ where the Decision Tree is built by examining reduced number of split points and maintaining the same classification accuracy. Prediction of forest cover types using Decision Trees is discussed in this paper. Maximum accuracy of about 84% is achieved using Decision Trees.

Key words : Classification, Prediction, Information gain, Gain Ratio, Gini Index.

1. INTRODUCTION

Several decision tree algorithms have been developed for classification. ID3 algorithm (Quinlan 1981) for classification uses information gain as a measure to select the best splitting attribute. The attribute with the highest information gain is selected as the splitting attribute. One of the main drawbacks of ID3 is that the measure Gain used tends to favor attributes with a large number of distinct values. This drawback was overcome to some extent in C4.5 (Quinlan 1993) by introducing a new measure called Gain Ratio.

SLIQ (Metha *et al.* 1996) is a decision tree classifier developed by the Quest team to handle both numeric and categorical attributes. SPRINT (Shaefer *et al.* 1996) was also developed by the Quest team that basically aims at parallelizing SLIQ. In SLIQ algorithm, while evaluating the best split for each numeric attribute having n values, the Gini index has number of split points to be evaluated at a node with m attributes is m*(n-1), m being the number of attributes. This makes SLIQ computationally complex for numeric attributes. The PUBLIC algorithm proposed by Rastogi *et al.* (1998) for tree generation is the same as SPRINT but Entropy is used as a measure for checking the goodness of split. Robust C4.5 algorithm (Zheng *et al.* 2005) is an improvement over C4.5. Elegant Decision Tree Algorithm (EDTA) developed by Chandra *et al.* (2002) was proposed as an improvement over SLIQ where the Gini index is computed not for every successive pair of values of an attribute but over different ranges of attribute values. This reduces the number of split points as well as the number of computations. It was shown that the classification accuracy was much better than SLIQ. In EDTA the number of split points evaluated is n / k (where n is the total number of different values the attribute can take and k is the total number of intervals or group size). In this paper an improvement over EDTA has been suggested to reduce the computational complexity. Robust Algorithm for Classification using Decision Trees (RDTA) developed by Chandra *et al.* (2006) is a further improvement over EDTA where Gini Index is evaluated for each attribute at displacements of sigma (standard

[1] *Institute of Technology, Nirma University, Ahmedabad*

deviation) from the minimum value and/or maximum value (in the direction of decreasing value) of the attribute. The other variations include evaluating Gini at additional split point at a displacement of two sigma from the minimum and maximum value.

The results for Decision Tree Classification of Forest Cover data using SLIQ and RDTA is discussed in the paper. The number of split points to be evaluated reduced considerably and the classification accuracy was at par to that of SLIQ.

2. OVERVIEW OF CLASSIFICATION ALGORITHMS USING DECISION TREES

This section presents overview of various decision tree algorithms developed so far.

A. ID3 Algorithm

ID3 algorithm (Quinlan 1981) uses information gain to decide the splitting attribute. Given a collection S of c outcomes, Entropy is defined as

$$\text{Entropy}(S) = \Sigma - p(I)\log_2 p(I) \qquad (1)$$

where p(I) is the proportion of S belonging to class I. Gain(S, A), the information gain of example set S on attribute A is defined as

$$\text{Gain}(S, A) = \text{Entropy}(S) - \Sigma\left(\left(|S_v|/|S|\right) * \text{Entropy}(S_v)\right) \qquad (2)$$

where S_v = subset of S for which attribute A has value v.

The attribute value that maximizes the information gain is chosen as the splitting attribute.

B. C4.5 Algorithm

C4.5 (Quinlan1993) is an extension of ID3 algorithm. Information Gain used in ID3 algorithm always tends to select attributes that have a large number of values since the gain of such an attribute would be maximal. To overcome this drawback Quinlan (1993) suggested the use of Gain Ratio as a measure to select the splitting attribute instead of Information Gain.

$$\text{Gain Ratio}(S, A) = \frac{\text{Gain}(S, A)}{\text{SplitInfo}(S, A)} \qquad (3)$$

where SplitInfo(S, A) is the information due to the split of S on the basis of the value of the attribute A.

$$\text{Splitinfo}(S, A) = I(|S_1|/|S|, |S_2|/|S|, \ldots, |S_m|/|S|) \qquad (4)$$

where $S_1, S_2, \ldots S_m$ are the partitions induced by attribute A in S.

C. SLIQ Algorithm

SLIQ (Supervised Learning in Quest, Mehta *et al.* 1996) was developed by the Quest team at IBM. Gini Index is used as a split measure. Gini Index is minimized at each split so that the tree becomes less diverse as we progress. The class histograms are built for each successive pairs of values of attributes. At any particular node, after obtaining all the histograms for all attributes, the Gini Index for each histogram is computed. Gini index for a sample histogram with two classes namely A and B is defined in Table 1.

In Table 1, P denotes the splitting value for an attribute, a1and a2 denote the number of attributes which are less than P and belong to class A and B respectively. b1 and b2 denote the number of attributes which are greater than P and belong to class A and B respectively.

$$\text{Gini Index} = \frac{a1 + a2}{n}\left[1 - \left(\frac{a1}{a1 + a2}\right)^2 - \left(\frac{a2}{a1 + a2}\right)^2\right] + \frac{b1 + b2}{n}\left[1 - \left(\frac{b1}{b1 + b2}\right)^2 - \left(\frac{b2}{b1 + b2}\right)^2\right] \qquad (5)$$

where n = Total number of records = a1+ a2 + b1+b2

Once the Gini Index for each histogram is known, the split attribute is chosen to be the one whose class histogram gives the least Gini Index, and the split value equals the splitting point P for that histogram.

Table 1. Class Histogram

Attribute Value < P	A	B
L	a1	a2
R	b1	b2

D. SPRINT and PUBLIC Algorithms

SPRINT (Shaefer *et al.* 1996) algorithm aims at parallelizing SLIQ. The splitting criterion used by SPRINT is based on the value of a single attribute. SPRINT avoids costly sorting at each node by presorting continuous attributes only once, at the beginning. Each continuous attribute is maintained in a sorted attribute list comprising of the attribute value, class label of the record, and its corresponding record identification. Best split point is determined in the same way as SLIQ.

In PUBLIC algorithm (Rastogi *et al.* 1998), the approach used for tree generation is same as SPRINT but the measure used for checking the goodness of split is Entropy. Pruning is carried out along with tree building. PUBLIC integrates the second "pruning" phase with the initial "building" phase. In PUBLIC, a node is not expanded during the building phase if it is determined that it will be pruned during the subsequent pruning phase. In order to make this determination for a node, before it is expanded, PUBLIC computes a lower bound on the minimum cost sub tree rooted at the node. This estimate is then used by PUBLIC to identify the nodes that are certain to be pruned, and for such nodes, not expand effort on splitting them.

E. Elegant Decision Tree Algorithm

Chandra *et al.* (2002) proposed an algorithm for improving the performance of SLIQ. In this algorithm, the Gini Index is computed not for every successive pair of values of an attribute like in SLIQ but over different ranges of attribute values. It was also observed that the number of split points at which Gini was computed was much less compared to that of SLIQ and also the classification accuracy was better than that of SLIQ. In this approach the total computations at a node is the product of the number of attributes (m) and the number of groups formed (g = n/k where k is the interval/group size) i.e. m*n/k. This reduces the number of split points as well as the number of computations.

F. Robust Algorithm for Classification using Decision Trees

Chandra *et al.* (2006) proposed an algorithm for improving the performance of EDTA. In this algorithm, standard deviation was computed for each numerical attribute in the Dataset (further referred to as PV1 – Proposed variation 1). The split point was taken as

$$\text{Splitpo int} = \text{L(i)} + \text{stdev(i)} \tag{6}$$

where L(i) denotes the least value and stdev(i) denotes the standard deviation of the ith attribute. Gini Index was computed at the corresponding split points for each of the attributes. The attribute that has the minimum Gini Index was chosen as the splitting attribute and the dataset was partitioned at the corresponding split value of the chosen attribute. This process was repeated recursively until all the examples in each partition belong to one class. The algorithm is given as follows:

Other variations of the method which are possible are

(a) Computing GINI at 2 split points at attribute value equal to L(A) + stdev(A) and L(A)+2*stdev(A). (Further referred as PV2 – Proposed Variation 2)

(b) Computing GINI at 1 split point at attribute value equal to U(A) – stdev(A) where U(A) is the maximum value of the attribute A. (Further referred as PV3 – Proposed Variation 3)

(c) Computing GINI at 2 split points at attribute value equal to U(A) - stdev(A) and U(A)-2*stdev(A). (Further referred as PV4 – Proposed Variation 4)

(d) Computing GINI at 2 split points at attribute value equal to L(A) + stdev(A) and U(A)-stdev(A). (Further referred as PV5 – Proposed Variation 5)

The following section discusses the performance of SLIQ and RDTA on forest cover dataset.

3. ILLUSTRATION

The advantage of Robust Decision Tree Algorithm is illustrated using the following example. Dataset used for illustration as given in Table 2 contains 4 attributes namely Slope in degrees, Horizontal distance to the nearest roadway, Hill shade index at 3pm summer solstice and Horizontal distance to nearest wildfire ignition points and 10 records. The data is labeled into two categories based on type of the forest. Class 1 denotes if the forest is Spruce/Fir and Class 2 denotes if forest is Lodgepole Pine.

Decision Tree was built using both SLIQ and the RDTA (PV1) algorithms. Table 3 shows the split point values of all the four attributes and the corresponding Gini Index values using SLIQ algorithm. The minimum Gini Index value among all the split points is shown as underlined bold. Attribute 2 has the minimum Gini Index

Table 2. Dataset

Sr. No.	Attribute 1 Slope in Degrees	Attribute 2 Horz Dist to nearest roadway	Attribute 3 Hillshade index at 3pm, summer solstice	Attribute 4 Horz Dist to nearest wildfire ignition points	Type of Forest
1	2	331	158	5745	2
2	3	1116	148	5091	2
3	1	1266	158	5079	2
4	3	2096	159	6853	1
5	12	2244	124	2958	1
6	11	3060	121	5654	1
7	5	4002	164	3460	1
8	10	4286	131	3248	1
9	9	4858	151	4548	2
10	12	5757	155	4017	2

Table 3. Gini Index values for various attribute splits (sliq) for the complete dataset

Sr. No.	Attribute No. 1		Attribute No. 2		Attribute No. 3		Attribute No. 4	
	SV	GI	SV	GI	SV	GI	SV	GI
1	1.5	0.44	723.5	0.44	122.5	0.44	3103.0	0.44
2	2.5	0.37	1191	0.38	127.5	0.38	3354.0	0.38
3	4.0	0.42	**1681**	**0.29**	139.5	0.29	3738.5	0.29
4	7.0	0.48	2170	0.42	149.5	0.42	4282.5	0.42
5	9.5	0.42	2652	0.48	153.0	0.48	4813.5	0.48
6	11.0	0.48	3531	0.50	156.5	0.50	5085.0	0.50
7	12.0	0.50	4144	0.48	158.5	0.38	5372.5	0.48
8			4572	0.38	161.5	0.44	5699.5	0.50
9			5307.5	0.44			6299.0	0.44

SV – Split point value, GI – Gini Index

value at split point value 1681. This is chosen as the root node and the dataset is partitioned (at Attribute 2 value equal to 1681) into two subsets. All the records in the subset where the values of second attribute are less than 1681 belong to class 2. The SLIQ algorithm is applied to the subset containing records where the second attribute is greater than 1681 as it consists of mixture of both the classes (i.e. as it is a impure dataset). The Split point values and the corresponding Gini Index values are shown in Table 4. Attribute 2 at value 4572 has the minimum Gini Index value and hence chosen as the splitting attribute. Both the subsets generated due to this split result into pure class as shown in Fig. 1.

Table 4. Gini Index values for various attribute splits (sliq) for the dataset having Attribute 2 value > 1681

Sr. No.	Attribute No. 1		Attribute No. 2		Attribute No. 3		Attribute No. 4	
	SV	GI	SV	GI	SV	GI	SV	GI
1	4	0.38	2170	0.38	122.5	0.38	3103	0.38
2	7	0.34	2652	0.34	127.5	0.34	3354	0.34
3	9.5	0.40	3531	0.29	141	0.29	3738.5	0.29
4	11	0.40	4144	0.19	153	0.40	4282.5	0.40
5	12	0.37	**4572**	**0.00**	157	0.34	5101	0.34
6			5307.5	0.24	161.5	0.38	6253.5	0.38

SV – Split point value, GI – Gini Index

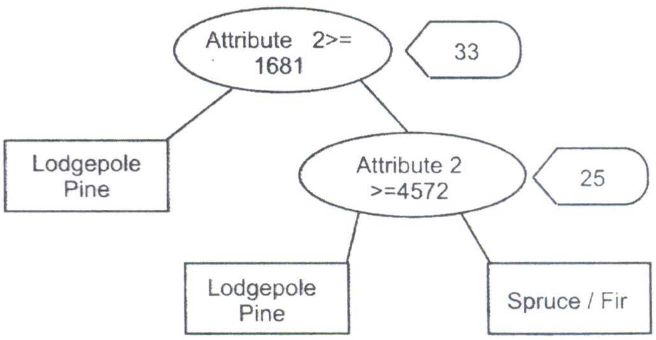

Fig. 1. Decision Tree generated using SLIQ algorithm

Table 5 shows the split points values and the Gini Index values when the decision tree is built using the RDTA algorithm. The Attribute 2 at value 2024.7 has the minimum Gini Index and hence chosen as the splitting attribute. The records where the Attribute 2 values are less than 2024.7 belongs to class 2, however, the records where Attribute 2 values are greater than or equal to 2024.7 is impure. RDTA algorithm is applied for the impure dataset. Table 6 shows the split point values and the Gini Index for the impure dataset where the values of the second attribute is greater than 2024.7. Attribute 2

Table 5. Gini Index values for various attribute splits (PV1) for the complete dataset

Sr. No.	Attribute No. 1		Attribute No. 2		Attribute No. 3		Attribute No. 4	
	SV	GI	SV	GI	SV	GI	SV	GI
1	5.2	0.48	**2024.7**	**0.29**	135.9	0.29	4145.1	0.42

SV – Split point value, GI – Gini Index

Table 6 . Gini Index values for various attribute splits (PV1) for the dataset having Attribute 2 value > 2024.7

Sr. No.	Attribute No. 1		Attribute No. 2		Attribute No. 3		Attribute No. 4	
	SV	GI	SV	GI	SV	GI	SV	GI
1	6.3	0.34	**3354.5**	**0.29**	**137.4**	**0.29**	4269.9	0.40

SV – Split point value, GI – Gini Index

is chosen as splitting attribute as it has the minimum Gini Index. The Attribute 2 is split at value 3354.5. The records having value of Attribute 2 less than 3354.5 belong to class 1 and the records having Attribute 2 value greater than or equal to 3354.5 is impure. RDTA algorithm is again applied to this dataset. The splitpoint values and the corresponding Gini Index values are shown in Table 7. Attribute 2 at splitpoint value 4672.5 has Gini Index equal to zero. Hence both the partitions resulting due to this are pure.

Table 7. Gini Index values for various attribute splits (PV1) for the dataset having Attribute 2 value > 3354.5

Sr. No.	Attribute No. 1		Attribute No. 2		Attribute No. 3		Attribute No. 4	
	SV	GI	SV	GI	SV	GI	SV	GI
1	7.5	0.33	**4672.5**	**0.00**	143.1	0.33	**3754.4**	**0.00**

SV – Split point value, GI – Gini Index

The Decision Tree generated using both the algorithms is shown in Figs. 1 and 2. The number of split points evaluated for each non leaf node is shown adjacent to each node. It is clearly visible that SLIQ evaluates 58 split points whereas RDTA evaluates only 12 split points to build the Decision Tree.

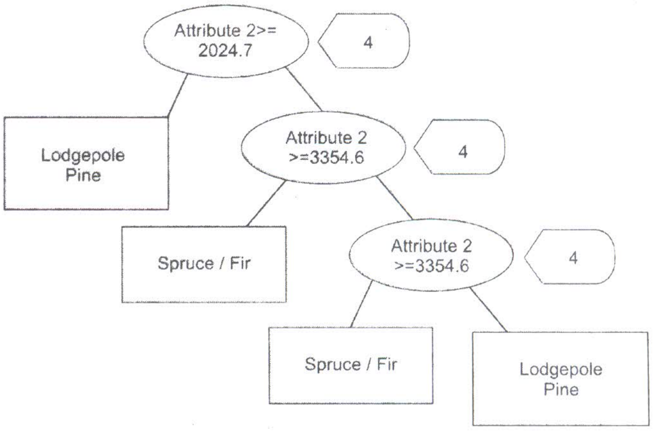

Fig. 2. Decision Tree generated using RDTA

Table 8. Description of attribute for forest cover dataset

Attribute Name	Data Type	Description
Elevation	quantitative	Elevation in meters
Aspect	quantitative	Aspect in degrees azimuth
Slope	quantitative	Slope in degrees
Horizontal_Distance_To_Hydrology	quantitative	Horizontal distance to nearest surface water features
Vertical_Distance_To_Hydrology	quantitative	Vertical distance to nearest surface water features
Horizontal_Distance_To_Roadways	quantitative	Horizontal distance to nearest roadway
Hillshade_9am	quantitative	Hillshade index at 9am, summer solstice
Hillshade_Noon	quantitative	Hillshade index at noon, summer soltice
Hillshade_3pm	quantitative	Hillshade index at 3pm, summer solstice
Horizontal_Distance_To_Fire_Points	quantitative	Horizontal distance to nearest wildfire ignition points
Wilderness_Area (4 binary columns)	qualitative	Wilderness area designation
Soil_Type (40 binary columns)	qualitative	Soil type designation

4. RESULTS

The performance of the RDTA algorithm was compared with SLIQ using Forest Cover dataset. The description of the forest cover dataset is shown in Table 8 and Table 9. A total of twelve cartographic measures were utilized as independent variables in the predictive models while seven major forest cover types were used as dependent variables.

The dataset includes four wilderness areas located in the Roosevelt National Forest of northern Colorado. These areas represent forests with minimal human-caused disturbances so that existing forest cover types are more a result of ecological processes rather than forest management practices. Some background information for these four wilderness areas: Neota (area 2) probably has the highest mean elevational value of the 4 wilderness areas. Rawah (area 1) and Comanche Peak (area 3) would have a lower mean elevational value while Cache la Poudre (area 4) would have the lowest mean elevational value. As for primary major tree species in these areas, Neota would have spruce/fir (type 1) while Rawah and Comanche Peak would probably have lodgepole pine (type 2) as their primary species followed by spruce/fir and aspen (type 5). Cache la Poudre would tend to have Ponderosa pine (type 3), Douglas-fir (type 6) and cottonwood/willow (type 4).

Table 9. Description of class for forest cover dataset

Class No.	Description
1	Spruce/Fir
2	Lodgepole Pine
3	Ponderosa Pine
4	Cottonwood/Willow
5	Aspen
6	Douglas
7	Krummholz

The dataset was split into two parts for training and testing. The Decision Tree was built using the training dataset and the prediction was carried out using the testing. The average classification accuracies (after ten fold cross validation) achieved after prediction is discussed below.

Among the proposed variations, PV1, PV2 and PV3 gave better classification accuracy compared to that of other variations as shown in Table 10.

Table 10. Testing accuracy for different datasets

Dataset name	Decision Tree Algorithms					
	SLIQ	PV1	PV2	PV3	PV4	PV5
Forest Cover	87.42	87.714	87.857	88.143	87.00	85.571

The number of split points evaluated while building the Decision Tree is shown in Table 11. It is observed that the number of split points is significantly reduced for the variations of RDTA compared to that of SLIQ.

Table 11. Number of split points evaluated for different algorithms

Dataset name	Decision Tree Algorithms					
	SLIQ	PV1	PV2	PV3	PV4	PV5
Forest Cover	25192	4428	4877	5130	5440	5687

The time taken by SLIQ and variations of RDTA algorithm is shown in Fig. 3. It is observed that all variations of RDTA algorithm takes less time for building the Decision Tree compared to that of SLIQ.

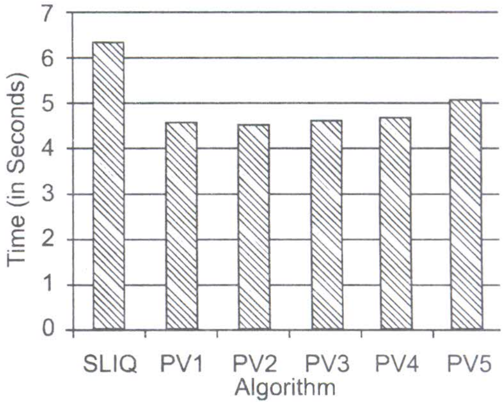

Fig. 3. Time taken by each algorithm

5. CONCLUSION

The forest cover data of the Roosevelt National Forest of northern Colorado was used to evaluate the performance of Decision Trees. The Decision Tree algorithm has achieved maximum classification accuracy 88.143% as compared to that of 70.58% Neural Network (Blackard *et al.* 2000). The variations of RDTA algorithm have shown considerable reduction in the number of split points to be evaluated compared to that of SLIQ algorithm and the classification accuracy of RDTA variation PV1, PV2 and PV3 is better than that of SLIQ.

REFERENCES

Blackard, Jock A. and Denis, J. Dean (2000). Comparative accuracies of artificial neural networks and discriminant analysis in predicting forest cover types from cartographic variables. *Comp. Elect. Agri.*, **24(3)**, 131-151.

Chandra, B., Mazumdar, S., Arena, V., Parimi, N. (2002). Elegant Decision Tree Algorithms for classification in data mining. *Proc. of the Third International Conference on Information Systems Engineering (Workshops) – (2002). IEEE CS*, 160 – 169.

Chandra, B. and Paul, Pallath (2006). Robust Algorithm for classification using Decision Trees. *Proc. of IEEE International Conference CIS-RAM 2006,* IEEE, 608-612.

Mehta , M., Agrawal, R. and Riassnen, J. (1996). SLIQ: A fast scalable classifier for data mining. In: *Extending Database Technology*, Avignon, Springer, France, 18-32.

Quinlan, J.R. (1986). Introduction of decision tree. *Machine Learning,* **1**, 81-106.

Quinlan, J.R. (1993). *C4.5: Programs for Machine Learning.* Morgan Kaufmann, San Mates, California.

Rastogi, R. and Shim, K. (1998). Public: A decision tree classifier that integrates building and pruning. In: *Proc. of 24th International Conference on Very Large Data Bases,* August 1998, 404–415, New York.

Ruggieri, S. (2002). Efficient C4.5 [classification algorithm] *IEEE Transactions on Knowledge and Data Engineering,* **14(2),** 438 – 444.

Shafer, J.C., Agarwal, R. and Mehta, M., (1996). SPRINT : A scalable parallel classifier for data mining. *Proc. of the 24th International Conference on Very Large Databases,* Mumbai.

Yao, Zheng, Liu, Peng, Lei, Lei and Yin, Junijie (2005). R- C4.5 Decision Tree model and its application to health core dataset. *Proc. of ICSSSM, IEEE,* **2(13-15)**, 1099-1103.