Pruning Strategies for Decision Trees

[1]Avinash S. Jagtap , [2]Jaya L. Limbore, [3]Neeta K. Dhane
[1]Associate Professor, [2]Assistant Professor, [3]Assistant Professor
[1]Department of Statistics, Tuljaram Chaturchand College, Baramati, Dist. Pune (MS), India
[2]Department of Statistics, Tuljaram Chaturchand College, Baramati, Dist. Pune (MS), India
[3]Department of Statistics, Tuljaram Chaturchand College, Baramati, Dist. Pune (MS), India

Abstract : Decision theory involves various analytical techniques that are designed with a view to help the decision maker in making optimal decisions. These decisions are to be made keeping in view the available alternatives and their potential consequences. It is a common practice to represent complex or complicated decision rules in the form of decision trees. This automatically generates a sequence of steps, where every step uses one feature or characteristic. The succession of steps indicates declining levels of importance or utility of the corresponding features. Decision trees are optimal in the sense that they minimize the misclassification error, which may be measured or represented in various different ways. It often happens that decision tree becomes less accurate or precise only because the risk function does not reduce as a result of branching. In such cases it is recommended to prune the tree by cutting branches that result in an increase in the risk function. This paper compares different pruning strategies that take different criteria into consideration. The comparisons are mainly based on complexity and leaf-node purity of each pruning strategy.

Keywords: Decision Theory, Decision Tree, Decision Tree Induction, Pruning of Decision Trees

1. INTRODUCTION

Decision theory, also known as the theory of choice, involves analyzing available choices before selecting one of them. There are three main branches of decision theory, namely normative, descriptive, and prescriptive. Normative decision theory is concerned with making the best decisions when a set of values and a set of uncertain beliefs are specified or given. As a result, normative decision theory is more concerned with the reasoning behind making decisions rather than actually making decisions. Descriptive decision theory, in contrast with normative decision theory, analyses how active (or existing) decision makers actually make their decisions so that some guidelines can be developed from this analysis. Finally, prescriptive decision theory attempts to guide the decision theory attempts to guide the decision maker by developing a procedure on what should be done in order to make the best decision according to the normative theory. It is important to note that the normative theory is concerned more with rational and consistent decisions, whereas the descriptive theory can accommodate some irrational decisions. Prescriptive theory combines the two approaches and develops procedures for decision making that are supported by acting decision makers sufficiently strongly, if not totally and at the same time, satisfy the requirements of normative theory.

Decision theory is closely related to game theory. However, it is important to note that decision theory is concerned with the choices available to the decision maker in the given circumstances. On the contrary, game theory is concerned with interactions between adversaries and hence involves decision making under changing or dynamic conditions. Nevertheless, decision theory is an interdisciplinary area of study the covers subjects like statistics, economics, psychology, biology, political and other social sciences, philosophy and computer science. The core of decision theory is based in choice under uncertainty. The famous mathematician Blaise Pascal introduced the idea of expected value in the 17th century. When there are several possible actions, and when each of these actions can result in two or more possible outcomes (that is, consequence) with different probabilities, then the reasonable procedure is to list all possible outcomes, determine their probabilities as well as values. This procedure further multiples two for every possible outcome to derive the expected value that is the expected value as the consequence of the corresponding action. The action that has the highest expected value is chosen as the best action. Daniel Bernoulli showed during the 18th century, that the expected value theory is necessarily wrong from the normative point of view. He then defined a utility function and proposed to use the expected utility instead of the expected value for making the decision. Abraham Wald pointed out in 1939 that the statistical procedures of parameter estimation and hypothesis testing are special cases of the general decision problem. It was Wald's paper that synthesized some statistical concepts like loss functions, risk functions, admissibility of a decision rule, antecedent distributions, minimax procedures, and Bayesian procedures.

Most of the areas of research that have developed data exploration techniques aim at generating rules in the form of decision trees automatically. A large number of studies focused on generation and application of decision trees are found in disciplines like statistics, decision theory, engineering, and machine learning (artificial intelligence). The origins of decision trees in statistics can be traced to research that builds binary segmentation trees with a view to understand the relationship between the input variable(s) and the target (or response) variable. Survey data analysis is the application that has motivated development of different algorithms for this purpose. Notable examples are AID (Sonquist et al., 1971) MAID (Gillo, 1972) THAID (Morgan and Messenger, 1973) and CHAID (Kass, 1980). Quinlan (1975) created a model for performing decision tree analysis and called it Iterative Dichotomizer 3 (ID3). This algorithm creates the smallest and most efficient decision tree. This algorithm was used to develop the popular C4.5 algorithm and its improvement in the form of C5.0 algorithm. A SAS publication state that "Decision trees are a multiple variable (or multiple effect) analyses. All forms of multiple variable analyses enable prediction, explanation description, or classification of an outcome or response or target."

2. DECISION TREES

Decision trees represent a non parametric method because they do not involve any parametric model. Decision trees are used mainly in problems of regression and classification. Their structures are hierarchical and they belong to the class of supervised learning algorithm. Decision trees split the data space into regions according to the response variable so that prediction is possible for unobserved instances.

A decision tree can be represented by a graph G = (V, E), where V is a non-empty finite set of vertices (also called nodes) and E is a set of edges (also nodes links). This graph satisfies the following conditions.

- An edge is an ordered pair of vertices, say (v, w) implying that the graph is a directed network.
- The graph is acyclic. That is, the graph contains no cycle.
- Since all edges are directed, every edge is a departing edge or an entering edge.
- There is exactly one node that has no entering edge. This node is called the root.
- Except for the root, every other node has exactly one entering edge.
- A path is a sequence of edges of the form (V_1, V_2), (V_2, V_3), …, (V_{n-1}, V_n) from node V_1 to node V_n. In particular when V_1 is the root node, there is a unique path from the root to every other node in the graph.
- For a path from node v to node w where $v \neq w$, the node v is said to be a proper ancestor of w and w is said to be a proper descendent of v. A node that has no proper descendent is called a leaf node or a terminal node. All other nodes, excepting the root node, are called intermediate nodes.

The root and every internal node use an attribute and a test for splitting data, while every edge represents one of the possible outcomes of the test. Breadth and depth are two important concepts regarding decision trees. The number of edges from the root to the farthest leaf node is called depth of the tree. The number of nodes at any particular level is called breadth of the tree. A binary tree, for example cannot have its breadth exceed 2^n at the n^{th} level. Breadth and depth of a tree are used as indicators of complexity of the tree. The higher their values are, the higher is the complexity of the decision tree.

There is no unique way of growing a decision tree. It is therefore not easy to grow an optimal decision tree. As a matter of fact, generating a minimal binary tree has been found to be an NP- complete problem The same complexity is established even in case and optimal decision tree is to be built form a decision table.

3. DECISION TREE INDUCTION

This paper considers the Concept Learning System framework (CLS) for top-down induction of a decision tree. The tree induction algorithm can be defined only by two steps of a recursion. For this, the set of observations in node t is denoted by D(t) and $y = \{y_1, y_2, \ldots, y_k\}$ denotes the set of k class labels. The two recursive steps of the algorithm are as follows.

Setp 1. If all observation in D(t) have the same label y_t then the node is made a leaf node and is labeled as y_t.

Setp 2. If observations in D(t) belong to two or more classes, an attribute is selected and a test is developed for the selected attribute to partition D(t) into smaller subsets. Each outcome of the test condition creates a branch and observations in D(t) are distributed among child nodes.

These two steps are recursively applied to each child node.

What is important to note in this algorithm is that the stopping rule, as specified in step 1, requires every leaf node to be pure that is every leaf node to contain all observations belonging to only one class. Practically this stringent stopping rule grows enormous decision trees and often leads to situation of overfitting. One solution to this problem is to stop the tree growth prematurely after the impurity in leaf nodes is within a specified limit. The other way is to grow the tree to the complete logical termination stage and then prune it as long as impurities stay within the specified limits.

4. PRUNING STRATEGIES

The focus of this paper being pruning strategies, this section describes six different pruning strategies available to the decision maker. Five of these six strategies are based on some consideration for error, while only one is based on the challenge of balancing between cost and complexity.

Before proceeding to a detailed discussion of different pruning strategies, it is necessary to introduce some notation related to pruning of decision trees. If t is an internal node (that is, it is neither the root nor a leaf node) then the set D(t) of observations is split by an attribute test. Suppose the attribute test has s possible outcomes then D(t) is divided into $D(t_1)$, $D(t_2)$, ..., $D(t_s)$ where the subsets are mutually exclusive and exhaustive. In other words, D(t) is partitioned. Each of these nodes is either a leaf note or an internal node. In the former case there is no more growth of the tree, while the trees grow further in the latter case. The nodes that succeed node t, directly or indirectly, also form a tree with node t as its root. It is therefore called a sub-tree of the original entire tree. Pruning removes a subree and makes the root of the removed sub tree a leaf node. Pruning can also be achieved by grafting, that is putting a succeeding node with its sub tree in place of an existing internal node. Grafting implies cutting out branches other than the existing node and the succeeding node that takes its place.

It is a common practice to divide the given data D in two parts, namely the training set D_{tr} and the test set D_{ts}. Let H denote the hypothesis space and h, h' two hypotheses in H. That is, let h, h' ∈ H. There are two errors related to hypothesis h. These are as follows.

1. Training error: $error_{D_{tr}}(h)$
2. Overall error: $error_D(h)$

Hypothesis h ∈ H is said to overfit training data if there is another hypothesis h' ∈ H satisfying $error_{D_{tr}}(h) < error_{D_{tr}}(h')$ and $error_D(h) > error_D(h')$

Similarly, the hypothesis h ∈ H is said to overfit D if there exists another hypothesis h' such that $error_D(h) < error_D(h')$ and $error^*(h) > error^*(h')$

where $error^*(h)$ denotes the true misclassification rate of h.

When D is divided in two subsets D_{tr} and $D_{ts,}$ then the hypothesis h ∈ H is said to overfit D if there exists another hypothesis h' ∈ H such that

$error_{D_{tr}}(h) > error_{D_{tr}}(h')$ and $error_{D_{ts}}(h) > error_{D_{ts}}(h')$

It is interesting to note that $error_{D_{tr}}(h)$ decreases monotonically as the size of tree increases.

Pruning is one of the ways of countering overfit. Pruning can be implemented through one of the strategies described in the remaining paper.

4.1 Error-based Pruning

Error-based pruning is a simple method and does not require a validation set. It is, nevertheless, criticized for not pruning a decision tree enough that is it underprunes a decision tree. Error-based pruning (EBP) uses the error at a node of the tree on the training data for estimating the error on test set at that node. The error rate is assumed to follow a binomial distribution, and pruning is controlled by the certainty factor (CF) parameter. This CF is also used for estimating an upper bound for the probability of error the population at a leaf node. This is done by using the CF as a confidence limit for the binomial distribution. Of course, this requires an assumption that the errors at a specified node form a sequence of independent trials and this question is questionable. This estimate is used for predicting the number (or proportion) of errors that would really occur at specified node. A smaller value of CF means to predict that more errors will occur than occurred in training data, leading to more pruning because the error rate at the leaf node is over-estimated. On the other hand, the CF value of 100 indicates no pruning because no error is predicted at the leaf node. As the value of CF decreases the chance and extent of pruning increases because the error rate is progressively predicted to be increasing, even when the training data size remains unchanged. The popular decision tree induction algorithm C4.5 uses this pruning method.

4.2 Minimum Error Pruning

Cestnik and Bratko (1991) developed this method, which is a bottom-up approach that attempts to minimize the expected rate of error on an independent dataset. If the prediction is that all future observations will belong to class C, the expected rate of error at node t is estimated by the following formula.

$$E_{rr}(t) = \frac{n_t - n_{t,c} + k - 1}{n_t + k}$$

where k is the number of classes ,

n_t is the size of node t, and

$n_{t,c}$ is the number of observations in node t that belong to class c.

Pruning is then decided according the following steps.

- At every internal node, the expected error is calculated if its subtree is pruned.
- At the same node, the expected rate of error is calculated if its subtree is not pruned.
- If the expected rate of error is smaller when the subtree is pruned, then the subtree is pruned. Otherwise, the subtree is not pruned.

4.3 Reduced Error Pruning

Quinlan (1987) proposed this simple and easily understood method of pruning decision trees. In this method, every decision node is a candidate for pruning. Pruning at a node removes the subtree rooted at that node and makes it a leaf node. The data set is divided into three parts namely, training set, validation set, and test set. Validation set is used for pruning, while test set is used for obtaining an unbiased estimate of accuracy for future observations. For every internal node T, consider its subtree S and find the change in the error rate (over the test set) if S is preplaced by the best possible leaf node in S. If the error rate does not increase and if no subtree of S has this property, then S is replaced by the leaf node. This process in continued until the error rate begins to increase as the result of pruning.

This method of pruning generates a sequence of trees. The major disadvantage of this method is that it requires a separate test set. Also if some parts of the original tree are not represented in the test set, then these parts may get pruned.

4.4 Pessimistic Error Pruning

Suppose the training set has N observations and the tree T is used for classifying these N observations. Consider a leaf of size K, where J observations are misclassified. In this case, the ratio J/K cannot be taken as a reliable estimate of the error rate for classifying unseen (or new) data. This is because the tree has been optimized for the training data. It is therefore suggested that a more practical estimate may be found by applying the continuity correction for the binomial distribution. In other words, we use J + 1/2 in place of J.

Now consider a subtree S of the tree T. Let L(s) denote the set of all leaf nodes of T and let ∑ K and ∑ J be totals of leaf sizes and numbers of misclassified observations. The pessimistic estimate of misclassification states that the number of misclassifications would be ∑ J + L(s) / 2 among ∑ K observations. Applying the same reasoning to a leaf node, where E cases are misclassified. The pessimistic pruning rule is to replace the subtree S with the best leaf is E + 1/2 is within one standard error of ∑ J + L(s) / 2. Subtrees of all interior nodes are examined to decide whether they should be pruned. It should be obvious that sub-subtrees of pruned subtrees are not required to be examined. It is necessary to note here that the standard error of estimate of the error rate is given by

$$SE = \sqrt{\frac{J(K-J)}{K}}$$

4.5 Critical Value Pruning

Mingers 1987 first proposed this method of pruning. This method requires a threshold to be set for estimating the strength of a node. This threshold is also called the critical value. A node is pruned if it does not reach the critical value. A node is not pruned if it satisfies the pruning condition but its child nodes do not all meet the condition because it contains relevant nodes. In such a case it should be obvious that the child nodes not reaching the critical value would be pruned, resulting in a smaller tree. If the critical value is set high, then pruning may be drastic and can cause trees to be small.

Critical value pruning involves two steps as described below.

Setp 1. Prune subtree in order to increase the critical value.

Setp 2. Measure the significance of every pruned tree so that the best tree can be selected.

In practice, the critical value is gradually increased and corresponding pruned tree is obtained. Then a comparison is made among these trees for their significance and predictive ability. The drawback of this pruning method is its tendency to under prune, resulting in selection of trees that have relatively low predictive accuracy.

4.6 Cost-Complexity Pruning

Breiman et al. (1984) first proposed this method of pruning a decision tree. This method is sensitive to the tree complexity as well as the error rate. Tree complexity is represented by the tree size. This method can be described to comprise of the following two steps.

Setp 1. A family of subtrees is selected according to some heuristic argument. This family is denoted by the set $\{T_0, T_1, T_2, ..., T_L\}$ of trees related to one another as described below. T_0 is the original decision tree. For i = 0, 1, ..., L - 1, the tree T_{i+1} is obtained from T_i by pruning branches showing the smallest increase in the rate of error per pruned leaf. This method, applied to successive subtrees, finally stops when T_L is just a leaf.

Setp 2. Select the tree in the family described in step 1 according to the estimated error rates of the trees in the family.

More specifically consider a subtree T used for classifying N observations in the training set and suppose E is the number of misclassified observations. Further, let N_T denote the number of leaves in the subtree T. The total cost-complexity of subtree T is defined as follows.

Cost-Complexity (T) = $\frac{E}{N} + \propto . N_t$

where α is the per-leaf cost in the sense that it is the reduction in error rate per leaf.

If the subtree T is pruned let the number of misclassified observations in the new tree be denoted by M. The new tree however, contains N_T - 1 fewer leaves than T. The cost would be unchanged when

$$\propto = \frac{M}{N(N_T - 1)}$$

The equation above allows calculation of α for every subtree so that the subtree (s) having the smallest value of α can be selected for pruning. This process is continued till a single-leaf subtree is obtained. The standard error of the misclassification rate is given by

$$SE = \frac{R(100 - R)}{N}$$

where R is misclassification rate of the pruned tree, and N is the test data set size.

The smallest tree whose observed number of misclassifications in the test data set does not exceed R + SE is selected.

This method uses a pruning set that is different from the training set. The limitation of this method is that it allows selection of a subtree only within the family $\{T_0, T_1, ..., T_L\}$ obtained in step 1, and not from all possible subtrees.

5. ILLUSTRATIVE EXAMPLE

The example for illustrating the performance and making comparisons of different pruning strategies is taken from Mingers (1989). Five different datasets are considered in this example. A brief description of each dataset will help the reader understand the level of complexity in the problem.

5.1 Profiles of Student in B.A. Business Studies (Babs).

This data set has 186 observations and seven attributes. The seven attributes are age in years, entry qualification (A - level, BTEC, Ordinary Diploma or Some Other), gender, number of O-levels, number of points (0-20) at A-level, mathematics grade (A, B, C, Fail) and full-time employment (Y/N) before the course. Degree has four possible classes, namely first, higher second, lower second, and third. The residual variation is high because many other factors that could affect the results have not been (or could not be) measured.

5.2 Recurrence of Breast Cancer (Cancer)

The data contains 286 observations on nine attributes namely age, size of tumor, number of nodes, malignancy (Y/N), radiation treatment (Y/N) affected area of breast (left, right, top, bottom, center) and the two classes are recur and not recur. Missing data as well as residual variation are present.

5.3 Classification of Iris (Iris)

The total sample size is 150 on three varieties of iris, each variety having 50 observations. Four numerical attributes are petal length, petal width, sepal length, and sepal width. The three varieties of Iris are setosa, versicolor, and virginica.

5.4 Recognizing Digits in LCD Display

A digit in LCD has seven line segments, where each line segment may be on or off. There are ten classes identifying the ten digits and seven attributes all of Y/N type. Three hundred randomly generated cases have been used in the example.

5.5 Soccer Result Prediction (Footb)

The dataset contains results of 346 soccer matches in British league. The observations are on five attributes that measure past performance of the team and the result can be win, lose, or draw.

5.6 Evaluation Criteria

Two criteria are important for evaluating a decision tree. These criteria are described briefly in this section.

5.6.1 Size of Tree

A generally accepted principle is that better models have fewer terms. In the case of statistical models, increased complexity improves the explanatory power of a model, but causes a decrease in its predictive ability. The size of a decision tree is given by the number of nodes, either all or only leaf nodes. Since these are related to each other, either one can be used as the criterion.

Table 1

Average size of pruned tree (number of leaves).

		Pruning Method					
Domain	Measure	Critical	Min-err	Err-comp	Pessim	Reduce	Size Unpruned
Babs							
	G-stat	5.4	6.6	2.7	9	8.3	60.4
	Marsh	10	4.3	2.4	7	6	
	Prob	10.8	9.1	2.4	6.3	5.1	
	G-R	7.7	9.2	2.6	4.8	5.8	
Cancer							
	G-stat	2.3	44.1	3.3	13.4	7.3	52.8
	Marsh	4.3	43.9	2.7	7.8	7.3	
	Prob	4.2	43.2	2.6	12.2	8.1	
	G-R	4.1	40.4	3	13.7	11.9	
Iris							
	G-stat	3.3	4.7	3.2	3.9	3	6.9
	Marsh	3.9	4.8	3.2	3.9	3.6	
	Prob	6.7	9.3	6.3	6.8	6.7	
	G-R	3.3	4.7	3.2	3.8	3	
Digits							
	G-stat	14.6	13.7	12.7	12.8	18.3	56.6
	Marsh	14.8	12.9	13.3	13.3	17.6	
	Prob	17.1	20.4	14.6	17.3	19.9	
	G-R	11.8	13.7	10.9	12.3	15.2	
Footb							
	G-stat	2.8	58.4	2.2	38	38	85.7
	Marsh	6.6	58.6	2.1	34.6	34.6	
	Prob	4	58.7	3	36.6	36.6	
	G-R	5.1	51.9	2.8	33.4	33.4	
Total		142.8	512.6	99.2	290.9	184.6	

5.6.2 Accuracy

Accuracy of a decision tree refers to its predictive ability. It is measured by the error rate. What is important is to note that accuracy can be low in small data sets.

Table 2

Average error rate for pruned trees(% miss-classified)

		Pruning Method				
Domain	Measure	Critical	Min-err	Err-comp	Pessim	Reduce
Babs						
	G-stat	42.8	41.9	42.5	44.4	40.7
	Marsh	43.4	43.6	42.1	43.9	41.8
	Prob	44.2	43.7	42.1	40.1	40.2
	G-R	45.8	44.4	42.3	42.3	42.1
Cancer						
	G-stat	30.4	31.7	28	27.5	27.1
	Marsh	27.7	31.3	27.7	28.1	26.9
	Prob	29	32.6	28.5	28.1	27.2
	G-R	30.7	33.2	30.3	28.4	29.2
Iris						
	G-stat	7.2	7.2	7.2	7.2	7.2
	Marsh	8.9	7.8	7.6	8.8	7.6
	Prob	8.7	6.7	9.5	7.6	10.7
	G-R	7.5	6.8	7.5	7.5	7.5
Digits						
	G-stat	30.2	30.9	29.5	30.1	28.8
	Marsh	29.6	30.8	29.3	29.9	29.2
	Prob	31.2	29.8	30.6	29.9	29.9
	G-R	31.6	31.6	30.9	31.3	30.5
Footb						
	G-stat	49.5	54.4	47.3	55.5	48.6
	Marsh	49.1	52.9	46.9	53.7	47.5
	Prob	48.4	54.5	48.2	55.2	48.3
	G-R	48.7	57.3	48.7	56.2	50.9
Total		664.6	673.1	626.7	655.7	621.9

6. CONCLUSIONS

It should be apparent from the illustrative examples that the different pruning methods differ significantly in their results. More specifically, minimum-error pruning is very sensitive to the number of distinct classes in the data, indicating that it is the least accurate method. On the other hand, pessimistic pruning is the crudest but also the quickest of all. Since its results are bad for certain data sets, it must be used with caution. Other methods produce good results, but require a separate test data set.

The second important conclusion of the current study is that there is no evidence of any interaction between the measures used in tree induction and tree pruning. This indicates that the choices of induction method and pruning method can be made independently of each other.

The comparisons made in the illustrative examples involve quantitative assessment. There is a need for assessment of the quality of rules that result from trees pruned according to different tree pruning methods. This is an open challenge for researchers in decision theory who are interested in decision tree induction and pruning.

REFERENCES

[1] Gillo, M. W. MAID: A Honeywell 600 program for an automatised survey analysis. Behavioral Science 17 (1972): 251-252.

[2] Kass, G. V. (1980). An exploratory technique for investigating large quantities of categorical data. Applied Statistics 29 (2), 119–127.

[3] Mingers, J. (1989). An Empirical Comparison of Pruning Methods for Decision Tree Induction. Machine Learning, vol. 4, pp. 227 - 243.
[4] Morgan, J. N. and R. C. Messenger (1973). THAID a sequential analysis program for analysis of nominal scale dependent variables. Survey Research Center, Institute for Social Research, University of Michigan, Ann Arbor.
[5] Pascal Blaise (1670) Pensees Bernoulli, Daniel (1738). "Exposition of a New Theory on the Measurement of Risk". Econometrica, vol. 22, No. 1, pp. 23-36.
[6] Quinlan, J. R. (1993). C4.5: Programs for Machine Learning. San Mateo: Morgan Kaufmann.
[7] Sonquist, J. A., E. L. Baker, and J. N. Morgan (1971). Searching for structure (Alias–AID–III). Survey Research Center, Institute for Social Research, University of Michigan, Ann Arbor.
[8] Wald Abraham (1939). "Contribution to the Theory of Statistical Estimation and Testing Hypothesis". Annals of Mathematical Statistics, Vol.10, No.4, pp. 299-326.