From: FLAIRS-00 Proceedings. Copyright © 2000, AAAI (www.aaai.org). All rights reserved.

A Fuzzy Algorithm For The Efficient Utilisation of Information In Decision Trees

Keeley A Crockett[1], Zuhair Bandar[1], Akeel Al-Attar[2]
[1]The Intelligent Systems Group, The Manchester Metropolitan University, Chester Street, Manchester, UK, M15 GD.
[2]Attar Software Limited, Newlands House, Newlands Road, Leigh, Lancashire, WN7 4HN, UK
Email: K.Crockett, Z.Bandar@doc.mmu.ac.uk

Abstract

Highly optimized decision trees which have been created from ID3-type algorithms are often recognized as being one of the best methods for partitioning a given domain in terms of both classification accuracy and for the formulation of small rule sets. However, these trees are highly optimized and potential information in lower branches of the tree is lost through pruning. This paper presents a novel algorithm which regains this information by relaxing sharp decision boundaries and by "re-growing" the decision tree by relaxing the pruning criteria. The algorithm produces more robust decision trees by allowing information in lower branches to contribute in the decision making process without causing significant overfit. Classification accuracy is also improved by regaining information from the low level branches.

1. Introduction

A difficulty associated with inducing trees is knowing when to stop growing the tree. What constitutes a "right" sized tree? This could depend on the classification problem, the generalizational ability or the classification accuracy. The most common technique which tackles this issue is known as pruning. It is a measure which examines the performance of a particular branch within the tree to decided whether or not to stop the growth down that specific branch (Quinlan 1993). Pruning removes those branches from a decision tree, which fail to contribute significantly to the resulting decision tree model after training has taken place. In order to produce a generalized model of the domain the trees are often highly pruned, resulting in potential information being lost in the lower levels of the tree. This coupled with the sharp decision boundaries at each node can have a substantial impact of the accuracy of the tree. Is a highly optimized tree therefore the best representation of a given domain?

In the original ID3 algorithm (Quinlan 1996), no pruning strategy was used. The induction process was simply continued until each node within the tree was pure. i.e. all examples at a node belong only to one class. This created very large trees which tended to fit the Training Set which resulted in the tree loosing it's ability to generalize. As a consummation of this a low percentage of unseen cases were correctly classified. A pruning strategy can be used to produce a generalized decision tree which does not suffer from over fit. Different levels of pruning can be applied depending on the amount of statistical significance which is required. A tree created from a representative sample of the domain which has been pruned will increase the error rate of the Training set, but will provide a more generalized model for future unseen cases.

The statistical backward pruning algorithm (Quinlan 1990) (Quinlan 1993) can be used to remove all attribute branches of an induced tree which are not statistically significant. Significance is measured by the Chi-square test of independence and can be set to a number of levels. As the significance levels decrease, the pruning criteria is relaxed and the crisp tree will possibly utilize more attributes resulting in extra branches being created. ID3-type algorithms select only a proportion of attributes for tree construction. Certain attributes are disregarded as they fail to contribute significantly to the decision process and are assumed to be noisy (often referred to as overfit). However, branches generated from these attributes are likely to contain useful information, which could contribute towards the classification process which ID3-type algorithms fail to utilize due to the limitations implicated by the creation of sharp decision boundaries.

This paper presents a new algorithm which firstly involves the application of Fuzzy Logic to crisp decision trees and secondly grows the tree from it's highly optimized state to various levels of significance. The algorithm applies principles of fuzzy set theory and Genetic Algorithms to relax the sharp decision boundaries at each continuous node. The resulting tree is more robust as it utilizes the potential information in the low level branches without causing significant overfit. Classification accuracy is also improved by regaining information from the low level branches.

2. Pruning Strategies

A number of pruning strategies have been developed over the years specifically for the ID3 family of decision trees. In early models, Quinlan used a strategy known as forward pruning which was concerned with looking one branch

Copyright © 2000, American Association for Artificial Intelligence (www.aaai.org). All rights reserved

ahead in order to determine whether the expansion of the branch would be beneficial or not (Quinlan 1993). The technique involves introducing a stopping criterion, which is examined before a further branch is grown in order to stop the continual growing of a node. Quinlan used a stopping criteria based upon the chi-square test of statistical significance. In certain domains the results obtained using forward pruning were satisfactory but in others there was an unevenness. Backward or post-pruning is a more recent pruning strategy which is used in variants of ID3. Quinlan's post-pruning technique (Quinlan 1993) uses an optimization criteria that offsets the complexity of the tree against it's observed classification accuracy on the training examples. Additional computation is required to initially grow the tree but is compensated for by a more substantial exploration of possible partitions.

C4.5, a more recent algorithm developed from ID3 uses a type of post-pruning known as pessimistic pruning where only the information in the training set is used to prune the tree. This particular strategy has been further improved by using Yates correction (Quinlan 1993) to estimate the reliability of classification when a leaf is impure. When dealing with uncertain and imprecise attribute values it is possible to estimate the probability of each outcome from cases in the training set. The relative probabilities are then combined for each of these assumptions.

3. Softening Decision Boundaries

In (Crockett, Bandar, Al-Attar 1997,1998) a new Fuzzy Inference Algorithm (FIA) was introduced which was shown to improve the classification accuracy of crisp decision trees by introducing fuzzification onto the branches of the tree and by combining membership grades using fuzzy inference. FIA first requires a tree to be created using a C4.5 type algorithm.

Once the crisp decision trees have been created, a statistical backward pruning algorithm is used to remove all attribute branches of the induced tree which are not statistically significant. This is done prior to the application of the FIA algorithm. Significance is measured by the Chi-square test of independence and can be set to a number of levels. Branches which contain less records than the Lower Branching Limit will also be removed.

Each path from root to leaf is then converted into a series of fuzzy IF-THEN production rules. A case passing through the tree will result in all branches in the tree firing to some degree, which is determined by each specific attributes degree of membership in the corresponding fuzzy region. To determine the classification outcome of a case passing through the tree, the membership grades at all branches are combined using a pre-selected fuzzy inference technique.

The application of FIA consists of three distinct processes: fuzzification, inference and optimisation which will be discussed in the forthcoming sections.

3.1 Fuzzification

Nodes within the crisply generated tree were fuzzified by creating fuzzy regions around each tree node in order to soften the sharp decision thresholds. A fuzzy region is defined using a pair of linear membership functions for each decision node. This is illustrated in Fig. 1. for a tree node with a decision threshold of 3, where the darker circles indicate a more intense membership function. Each linear membership function is defined by upper and lower bounds *dm* and *dn,* about the decision threshold (*dt*) of the attribute which is determined by the tree induction algorithm.

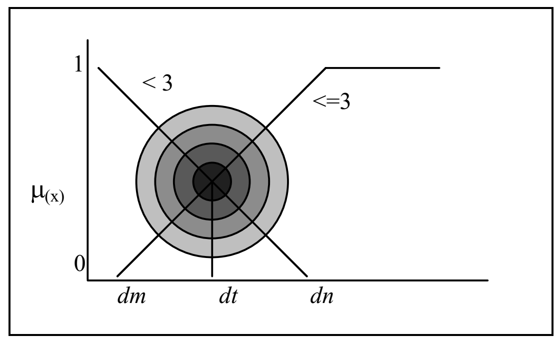

Figure 1. Fuzzy region around *dt*

The domain of a membership function at branch *i* can hence be defined as

$$dm_i = dt_i - n_j\sigma_i \quad \text{and} \quad dn_i = dt_i + n_{j+1}\sigma_i \qquad (1)$$

where σ_i is the standard deviation of attribute *I*, *n* is a real number $n \rightarrow [0,\infty]$, and *dm* and *dn* are the lower and upper bounds respectively of membership function *i*.

3. 2 Inference

A set of data ***S*** will consist of ***n***-attributes $\{A_1, A_2 .... A_n\}$ of domain D which are used to describe a single object. Applying an inference technique onto an existing tree of m-branches involves the combination of ***V*** membership function values $\{\mathbf{v_1}, \mathbf{v_2}, ... \mathbf{v_n}\}$ of all root to leaf node paths. Let ***T*** be a set of all possible outcomes $\{t_1, t_2 .. t_n\}$ defined from an existing crisp tree. An inference mechanism, **IM** which consists of an intersection function ***f1***, will take in ***V*** and produces a set of minimum outcomes **Min** $\{Min_1, Min_2 .... Min_j\}$ where *j* is the number of leaf nodes, and a union function $f2$, which combines output $f1$ to produce a maximum membership grade **O**.

Let $f1, f2, O \in \{0,1\}$ consisting of real numbers, $\Re$.

- Applying the fuzzy intersection function, $f1$

$$f1(\{v_1,v_2...v_n\}) \rightarrow \text{Min}\ \{\text{Min}_1,\text{Min}_2...\text{Min}_j\} \qquad (2)$$

- Applying leaf probabilities

The leaf probability represents the probability that an example reaching a leaf node will have the same outcome as the leaf. The probability of the dominant outcome is defined as

$$P = \frac{C_d . W_d . NF_d}{\sum_{i=1}^{n} C_i . W_i . NF_i} \qquad (3)$$

where C_i, W_i and NF_i are the frequency, weight and normalisation factor respectively of outcome i. C_d, W_d and Nf_d are the frequency, weight and normalisation factor respectively of the displayed outcome.

Let P be a set of leaf probabilities $\{p_1,p_2...p_j\}$ then

$$f1(\{v_1,v_2...v_n\}) \rightarrow \text{Min}\ \{(\text{Min}_1{*}p_1),(\text{Min}_2{*}p_2)..(\text{Min}_j{*}p_j)\} \qquad (4)$$

Each leaf probability is applied to the corresponding membership grade at each leaf node, after the intersection operation.

- Applying fuzzy union function $f2$

$$f2(\{(\text{Min}_1{*}p_1),(\text{Min}_2{*}p_2)......(\text{Min}_j{*}p_j)\}) \rightarrow \mathbf{O} \qquad (5)$$

O is the fuzzy singleton used to determine the success of correct classification having taken place for S.

Zadeh's min-max fuzzy inference technique (Zadeh 1965, 1992) will be used to combine grades of membership generated by the linear membership functions for each attribute down all paths within the tree. Although this technique is sometimes criticised by not allowing interaction of membership grades, it is still used as the standard benchmark inference technique in many fuzzy systems. Previous work showing a comparison of a number of inference techniques can be found in (Crockett, Bandar, Al-Attar 1998).

3.3 Optimization Using A Genetic Algorithm

When fuzzifying a tree, it is essential to obtain a balance of fuzziness. Too much fuzzification leads to additional uncertainty in the tree, whilst too little has insignificant impact on the performance. To determine sufficient fuzziness for a given tree a Genetic Algorithm (GA) is used (Grefenstette 1993) The membership functions are encoded onto a chromosome where each gene will represent a real value n, used in the determination of one domain delimiter (dm_i or dn_i). This is illustrated in figure 2.

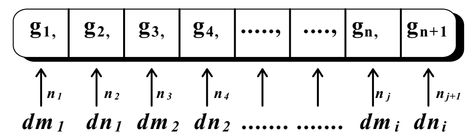

Figure 2. Chromosome Representation

The GA used to optimise FIA was provided by the software package XpertRule (Al-Attar 1998). Each chromosome was passed as a real number array. Once a population of chromosomes had been generated by XpertRule, each chromosome was passed into FIA in order for it's fitness to be evaluated. This involved the process of mapping each gene to a membership function domain delimiter and then generating one run to obtain the cost function to be returned to XpertRule for further optimisation.

3. 4 Reducing the Search Space

In order to focus the search, constraints can be applied in order to restrict the range of values a gene may take. In (Crockett, Bandar, Al-Attar 1999) it was shown that the best performance of FIA was achieved when the fuzzy regions created around each tree node were no more than two standard deviations around the decision threshold of a given attribute. It was found that this restriction on the amount of fuzziness could be used to constrain the genes, therefore reducing the search space and hence the time required to seek a optimal or near optimal amount of fuzzification. The average percentage classification accuracy of the training set will be used as a measure of the fitness of the GA. This will be achieved by using membership function domains generated by the GA during it's evolutionary cycle to define a fuzzy set at each decision tree branch. The aim is to obtain a optimal or near optimal set of parameters from a training set and apply them independently within the fuzzy algorithm to measure the effect on unseen test cases. This automated approach to membership determination relinquishes the need for a expert to define a series of fuzzy sets for each specific domain.

4. Experiments

4.1 Data sets and Methodology

Two real world data sets known as *Mortgage* and *Diabetes in Pima Indians* were used to create all decision trees. The Mortgage data set investigates the possibility of a person acquiring a mortgage and comprises of 8611 records featuring 25 discrete and continuous attributes. (4306 representing a Good Risk and 4305 depicting a Bad Risk). The second set known as Diabetes in Pima Indians investigates whether Pima Indian patients show signs of diabetes and comprises of 768 records featuring 9 continuos attributes (500 Class 1, indicating that a person has diabetes, 268 Class 2 which represents a person who shows no signs of the disease).

Each data set was first partitioned into two sets of randomly selected examples referred to as the Training and Testing sets. The Training set contains an equal number of class 1 and class 2 examples. It has been previously shown [8] that binary trees produce a higher classification accuracy and therefore 5 binary ID3 tree was created for each data set using statistical backward pruning (as described in section) each with significance levels of 0.1%, 0.5% and 1%. This produced a number of different sized trees from the same training sets for the purposes of comparison. A GA was then applied to optimize the membership functions assigned to each tree branch and the parameters of inference operators. Zadeh's non-parameterised inference technique was used to combine membership grades (Zadeh 1965, 1992). Table 1 shows the GA parameters used.

Data Sets	Mortgage Diabetes
Significance Level of trees	0.1%
Inference Technique	Min-Max
Domain delimiters dm and dn	Gene constrained {0,2}
Number of Generations	50 - 300 (varied)
Number of Individuals	50
Crossover Probability	0.5
Mutation Probability	0.05

Table 1. GA Parameters

5. Results

This section examines the results obtained for trees created from the Diabetes and Mortgage data sets. Each table shows the classification results obtained for the crisp tree pruned to some significance level and those obtained by application of the fuzzy inference algorithm. Table 2 shows the results obtained for both the crisp trees and the fuzzified tress when chi square is set to 0.1%. This is typically used as the highest significance level for creating highly optimized trees.

	Diabetes		Mortgage	
(Test)	Crisp Tree	FIA	Crisp Tree	FIA
% AVG	70	75	67	69.5
% Class 1	89	83	70	75
% Class 2	52	67	64	64

Table 2. Chi square 0.1%

5.2 Increasing the Tree Size

To increase the tree size, the significance level was relaxed. The fuzzification of the additional branches is expected to create a more generalized tree. This will be achieved by utilizing information drawn from the additional branches generated by relaxing the pruning criteria. To enable a comparison to be made between fuzzified trees of various significance, the exact same Training and Test sets will be used.

	Diabetes		Mortgage	
(Test)	Crisp Tree	FIA	Crisp Tree	FIA
% AVG	68	76	65	70
% Class 1	55	74	69	73
% Class 2	80	78	61	68

Table 3. Chi square 0.5%

It can clearly be seen in Table 3 that typically, the performance of crisp trees on unseen test cases deteriorates when the significance is relaxed as a result of the tree overfitting the Training set. This has occurred on both data sets and is clearly shown in Table 3 when compared with Table 2. However, FIA has utilized valuable information present in these additional branches, which is shown by the improved performance.

Diabetes

By relaxing the pruning criteria from 0.1% to 0.5%, 3 extra branches were created all of which had been previously used within the tree. The results for this data set show clearly that by increasing the significance the performance of the 0.5% fuzzified tree improves by 8% on the crisp tree (Table 3) compared with 5% improvement on the 0.1% crisp tree (Table 2). The Diabetes data set consists entirely of continuous attributes and thus this substantial improvement lies with the sensitivity of continuous attributes to fuzzification. In this instance FIA has efficiently utilized the three additional branches, which have been created.

Mortgage
The 0.5% tree contained 19 additional branches with a distribution of 10 discrete and 9 continuous attributes, 4 of which had not been previously selected. This resulted in the performance of the crisp tree declining by 2%. Table 3 shows that FIA achieved a 5% increase over the crisp tree to yield a 70% average, thus giving the same overall performance as that obtained from fuzzifying the smaller tree (Table 2). The fact that the performance of the 0.1% fuzzified tree was not exceeded could be attributed to a combination of the quantity of discrete attributes in the tree and that a proportion of the additional branches were too noisy and could not be compensated for by fuzzification.

5.3 Growing the crisp tree further

A further set of experiments were undertaken to examine the effects of FIA when significance levels were further decreased from 99.5% to 99% i.e. the pruning criteria was relaxed to 1% (Table 4). The objective was to determine if FIA could continue to utilize the information present in additional branches.

	Diabetes		Mortgage	
(Test)	Crisp Tree	FIA	Crisp Tree	FIA
% AVG	68	76	65	69
% Class 1	55	74	70	69
% Class 2	80	78	60	69

Table 4. Chi square 1%

Diabetes
By decreasing the significance of the Diabetes tree, one extra tree node was created. The attribute had been used previously within the tree. It was established that this node had an extremely low fuzzy threshold i.e. $n^L_k \leq 0.1$. Therefore, the performance remained the same as that obtained with the 0.5% fuzzy tree. (Table 3)

Mortgage
The 1% Mortgage tree consisted of 11 additional tree nodes, 4 discrete and 7 continuous. Two attributes had not been previously selected. The results in Table 3 show that by increasing the pruning level from 0.5% to 1%, FIA obtained a performance of 69% which was evenly distributed between the two outcome classes. This was a 4% improvement on the crisp tree. Compared with both the 0.1% and 0.5% fuzzy trees, the performance has declined by 1%, possibly caused by inherent noise within the additional branches selected by ID3. Additionally, the performance has been affected by the high proportion of discrete attributes used within the tree. The impact of fuzzifying discrete nodes on the overall performance is minimal.

6. Conclusion

This set of experiments has shown that FIA can utilize additional branches created from relaxing the pruning criteria by the process of fuzzification, thus transforming excess branches into potentially useful information. The resulting tree is more robust and can deal more effectively with noise. The size of the tree and the amount of pruning applied becomes less relevant. It has also been shown that there is a limit on how much fuzzification can improve the performance. The impact on the performance was dependent on two factors. Firstly the proportion of additional attributes which were continuous and secondly the degree of noise present in the extra branches selected by ID3. The best improvement came from the Diabetes data set which clearly illustrated that the addition of continuous branches can lead to an improved performance up to a certain extent. The lower the significance level the more difficult it becomes to extract useful information from the often noisy branches selected by ID3 and signs of overfitting become more apparent.

References

Al-Attar, A. 1998. XpertRule Analyser Software Package. Attar Software Ltd. Newlands House, Newlands Road, Leigh, Lancashire, WN7 4HN, England.

Crockett, K. Bandar, Z. Al-Attar, A. 1997 Fuzzy Rule Induction From Data Sets. *In Proceedings of The 10th International Florida Artificial Intelligence Research Symposium*: 332-336.

Crockett, K. Bandar, Z. Al-Attar, A. 1998 A Fuzzy Inference Framework For Induced Decision Trees. In *ECAI 98. 13th European Conference on Artificial Intelligence, Brighton, UK.* John Wiley & Sons: 425-429.

Crockett, K. Bandar, Z. Al-Attar, A. 1999. Optimising Decision Classifications Using Genetic Algorithms. In *ICANNGA 99, Artificial Neural Networks and Genetic Algorithms*. Springer Wein New York : 191-195.

Grefenstette, J. 1993. Genetic Algorithms For Machine Learning, A Special Issue of Machine Learning. Machine Learning Vol. 13, Nos. 2-3. Kluwer Academic Publishers.

Quinlan, J, R. 1986 Induction of Decision Trees, Machine Learning 1: 81-106.

Quinlan, J, R. 1990 Probabilistic Decision Trees. Machine Learning Volume 3: An AI Approach. Eds Kockatoft, Y. Michalshi, R:140-152.

Quinlan, J, R. 1993. C4.5 : Programs for Machine Learning. Morgan Kaufmann Publishers.

Zadeh, L. 1965. Fuzzy Sets. Information and Control 8 : 228-353.

Zadeh, L. 1992. Knowledge Representation In Fuzzy Logic. An Introduction To Fuzzy Logic Applications In Intelligent systems. Kluwer Academic Publishers.