Pre-pruning Classification Trees to Reduce Overfitting in Noisy Domains

Max Bramer

Faculty of Technology, University of Portsmouth, UK
Max.Bramer@bcs.org.uk
http://www.btinternet.com/~Max.Bramer

Abstract. The automatic induction of classification rules from examples in the form of a classification tree is an important technique used in data mining. One of the problems encountered is the overfitting of rules to training data. In some cases this can lead to an excessively large number of rules, many of which have very little predictive value for unseen data. This paper describes a means of reducing overfitting known as *J-pruning*, based on the *J-measure*, an information theoretic means of quantifying the information content of a rule. It is demonstrated that using J-pruning generally leads to a substantial reduction in the number of rules generated and an increase in predictive accuracy. The advantage gained becomes more pronounced as the proportion of noise increases.

1 Introduction

The growing commercial importance of knowledge discovery and data mining techniques has stimulated new interest in the automatic induction of classification rules from examples, a field in which research can be traced back at least as far as the mid-1960s [1].

A problem that arises with all methods of generating classification rules is that of *overfitting* to the training data. In some cases this can result in excessively large rule sets and/or rules with very low predictive power for previously unseen data. A smaller number of more general rules may have greater predictive accuracy on unseen data, at the expense of no longer correctly classifying some of the instances in the original training set. Alternatively, a similar level of accuracy may be achieved with a more compact set of rules.

Most work in this field to date has concentrated on generating classification rules in the intermediate form of a decision tree using variants of the TDIDT (Top-Down Induction of Decision Trees) algorithm [2].

A method for reducing overfitting in classification rules known as *J*-Pruning has previously been reported [3]. The method makes use of the value of the *J-measure*, an information theoretic means of quantifying the information content of a rule. The rules are *pre-pruned*, i.e. pruned as they are being generated.

In this paper the robustness of this technique in the presence of noise is examined.

A comparison is made between the results obtained from the unpruned and J-pruned versions of TDIDT for varying levels of noise added in a systematic fashion to three datasets from the UCI Repository of Machine Learning Datasets [4].

2 Overfitting of Classification Rules to Data

One approach to reducing overfitting, known as *post-pruning*, which is often used in association with decision tree generation, is to generate the whole set of classification rules and then remove a (possibly substantial) number of rules and terms, by the use of statistical tests or otherwise. An empirical comparison of a number of such methods is given in [5]. An important practical objection to post-pruning methods is that there is a large computational overhead involved in generating rules only then to delete a high proportion of them, especially if the training sets are large.

Pre-pruning a classification tree involves truncating some of the branches prematurely as they are being generated. Each incomplete branch (rule) such as

IF x = 1 AND z = yes AND q > 63.5 …. THEN …

corresponds to a subset of instances currently 'under investigation'. If not all the instances have the same classification the TDIDT algorithm would normally extend the branch to form a subtree by selecting an attribute to split on.

When following a pre-pruning strategy the subset is first tested to determine whether or not a termination condition applies. If it does not, a 'splitting attribute' is selected as usual. If it does, the branch (rule) is *pruned*, i.e. it is treated as if no further attributes were available and the branch is labeled with the most frequently occurring classification for the instances in the corresponding subset.

Reference [6] reports on experiments with four possible termination conditions for pre-pruning rules as they are generated by TDIDT, e.g. truncate each rule as soon as it reaches 4 terms in length. The results obtained clearly show that pre-pruning can substantially reduce the number of terms generated and in some cases can also increase the predictive accuracy, in all cases with a considerable reduction in computation time compared with generating complete trees. Although the results also show that the choice of pre-pruning method is important, it is not clear that (say) the same length limit should be applied to each rule, far less which of the termination conditions is the best one to use or why. There is a need to find a more principled choice of termination condition to use with pre-pruning, if possible one which can be applied completely automatically without the need for the user to select any 'threshold value' (such as the maximum number of terms for any rule). The *J-measure* described in [6] provides the basis for a more principled approach to pre-pruning.

3 Using the J-measure for Pre-pruning Classification Trees

The *J-measure* was introduced into the rule induction literature by Smyth and

Goodman [7] who give a strong justification of its use as an information theoretic means of quantifying the information content of a rule that is soundly based on theory.

Given a rule of the form **If Y=y, then X=x**, the (average) information content of the rule, measured in bits of information, is denoted by J(X;Y=y). The value of this quantity is the product of two terms:

- p(y) The probability that the hypothesis (antecedent of the rule) will occur - a measure of *hypothesis simplicity*
- j(X;Y=y) The *cross-entropy* - a measure of the *goodness-of-fit* of a given rule.

In what follows, it will be taken as a working hypothesis that a rule with a high J value (i.e. high information content) is also likely to have a high level of predictive accuracy for previously unseen instances.

There are several ways in which J values can be used to aid classification tree generation. One method, which will be called *J-pruning*, is to prune a branch as soon as a node is generated at which the J value is less than that at its parent.

Thus for example consider an incomplete rule

IF attrib1 = a AND attrib2 = b …. (with J-value 0.4)

which is expanded by splitting on categorical attribute *attrib3* into the three rules

IF attrib1 = a AND attrib2 = b AND attrib3 = c1 …. (with J-value 0.38)
IF attrib1 = a AND attrib2 = b AND attrib3 = c2 …. (with J-value 0.45)
IF attrib1 = a AND attrib2 = b AND attrib3 = c3 …. (with J-value 0.03)

Assuming that none of the new rules is complete (i.e. corresponds to a subset of instances with only one classification) all three would be considered as candidates for J-pruning. As the J-values of the first and third are lower than that of the original (incomplete) rule each rule would be truncated, with all the corresponding instances classified as belonging to the class to which the largest number belong. For example, the first new rule might become

IF attrib1 = a AND attrib2 = b AND attrib3 = c1 THEN Class = 5

The second new rule has a larger J-value than the original rule and in this case the TDIDT algorithm would continue by splitting on an attribute as usual.

The difficulty in implementing this method is to know which classification to use when calculating the J-value of an incomplete rule. If there are only two classes the value of J is the same whichever is taken. When there are more than two classes an effective heuristic is to generate the J-value for each of the possible classes in turn and then to use the largest of the resulting values.

Reference [3] compares the results obtained using the TDIDT algorithm both with and without J-pruning for 12 datasets, mainly taken from the UCI Repository [4]. The results were calculated using 10-fold cross-validation in each case. TDIDT was used with the Information Gain attribute selection criterion throughout.

For many of the datasets a considerable reduction in the number of rules was obtained using J-Pruning (e.g. from 357.4 unpruned to 25.9 J-pruned for *genetics* and from 106.9 unpruned to 29.6 J-pruned for *soybean*). Averaged over the 12 datasets the number of rules was reduced from 68.5 to only 19.1. The effect on the predictive accuracy of the generated rulesets varied considerably from one dataset to another, with J-pruning giving a result that was better for 5 of the datasets, worse for 6 and unchanged for one, the average being slightly lower with J-Pruning than without. Although these results were very promising, an important criterion, not discussed in [3], for evaluating any classification rule generation algorithm is its *robustness*, particularly when noise is present in the data. This forms the topic of the next section.

4 Experiments with Noisy Datasets

Many (perhaps most) real-world datasets suffer from the problem of *noise*, i.e. inaccurately recorded attribute or classification values. Although the user of a rule generation algorithm will generally be unaware that noise is present in a particular dataset, far less the proportion of values that are affected, the presence of noise is likely to lead to an excessively large number of rules and/or a reduction in classification accuracy compared with the same data in noise-free form.

The robustness of the unpruned and J-pruned versions of the TDIDT algorithm to noise was investigated using the *vote* dataset from the UCI Repository [4]. The dataset comprises information about the votes of each of the members of the US House of Representatives on 16 key measures during 1984. The dataset has 300 instances, each relating the values of 16 categorical attributes to one of two possible classifications: *republican* or *democrat*. It seems reasonable to suppose that the members' votes will have been recorded with few (if any) errors, so for the purpose of these experiments the *vote* dataset in its original form will be considered noise-free.

From this dataset further datasets were created by contaminating the attribute values with progressively higher levels of noise. There were eight such datasets, named *vote_10*, *vote_20*, …, *vote_80*, with the numerical suffix indicating the percentage of contaminated values.

The methodology adopted in the case of say *vote_30* was to consider the possibility of contaminating each attribute value in each instance in turn. For each value a random number from 0 to 1 was generated. If the value was less than or equal to 0.30 the attribute value was replaced by another of the valid possible values of the same attribute, selected with equal probability. The original classification was left unchanged in all cases. As the level of noise contamination increases from zero (the original dataset), through 10%, 20%, … up to 80%, it is to be expected that (with any method) the predictive accuracy of any ruleset generated will decline, possibly severely.

Figure 1 shows the number of rules generated using the TDIDT algorithm (with the 'Information Gain' attribute selection criterion) in its standard 'unpruned' form and with J-pruning for each of the datasets *vote_10*, *vote_20*, … *vote_80*. Figure 2 shows the corresponding levels of predictive accuracy for the two forms of the algorithm for the nine versions of the *vote* dataset. All results were calculated using 10-fold cross-validation. The J-pruned algorithm clearly produces substantially fewer rules with at least as good predictive accuracy as the unpruned version.

This experiment was repeated for two further datasets taken from the UCI Repository: *genetics* and *agaricus_lepiota*. The *genetics* dataset comprises 3,190 instances, each with 60 categorical attributes and 3 possible classifications. The *agaricus_lepiota* dataset comprises 5,644 instances (after those containing any missing values were removed), each with 22 categorical attributes and 2 possible classifications. These datasets were chosen partly because all the attributes were categorical. It was considered that categorical values were less likely to be wrongly (or imprecisely) recorded than continuous ones.

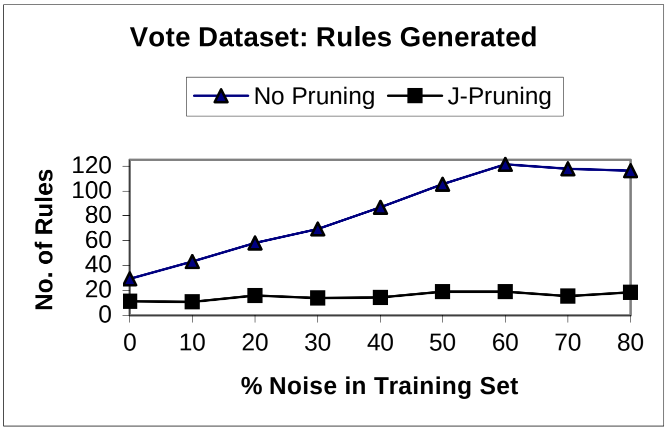

Fig. 1. Comparison of Number of Rules Generated: *vote* Dataset

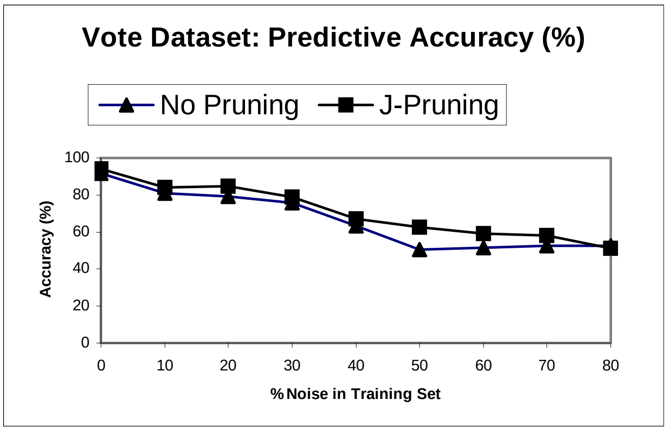

Fig. 2. Comparison of Predictive Accuracy: *vote* Dataset

The results of the experiments for these datasets (again calculated using 10-fold cross-validation) are given in Table 1, with values rounded to the nearest integer.

The reduction in the number of rules obtained using J-pruning increases substantially as the percentage of noise in the data increases. In the most extreme case, for *agaricus_lepiota_80*, the unpruned version of TDIDT gives 2916 rules and the J-pruned version only 19. The predictive accuracy obtained using J-pruning was better than that for the unpruned version of TDIDT in all cases where the proportion of noise exceeded 10%.

Table 1. Rules Generated and Predictive Accuracy: *genetics* and *agaricus_lepiota*

	genetics				*agaricus_lepiota*			
Noise %	Rules		Accuracy (%)		Rules		Accuracy (%)	
	Un-pruned	Pruned	Un-pruned	Pruned	Un-pruned	Pruned	Un-pruned	Pruned
0	357	26	89	78	15	10	100	100
10	918	122	73	72	349	96	96	95
20	1238	158	60	67	794	128	89	91
30	1447	185	54	64	1304	149	81	86
40	1652	175	44	60	1827	159	72	80
50	1815	163	36	55	2246	167	64	76
60	1908	165	33	52	2682	167	55	71
70	1998	153	29	51	3003	184	48	67
80	2074	179	27	48	2916	19	52	74

5 Conclusions

Overall these results clearly demonstrate that the J-pruning technique is robust in the presence of noise. Using J-pruning rather than the unpruned form of TDIDT (with attribute selection using Information Gain) will generally lead to a substantial reduction in the number of classification rules generated. This will often be accompanied by a gain in predictive accuracy. The advantage gained by using J-pruning becomes more pronounced as the proportion of noise in a dataset increases.

References

1. Hunt, E.B., Marin J. and Stone, P.J. (1966). Experiments in Induction. Academic Press
2. Quinlan, J.R. (1993). C4.5: Programs for Machine Learning. Morgan Kaufmann
3. Bramer, M.A. (2002). An Information-Theoretic Approach to the Pre-pruning of Classification Rules. Proceedings of the IFIP World Computer Congress, Montreal 2002.
4. Blake, C.L. and Merz, C.J. (1998). UCI Repository of Machine Learning Databases [http://www.ics.uci.edu/~mlearn/MLRepository.html]. Irvine, CA: University of California, Department of Information and Computer Science
5. Mingers, J. (1989). An Empirical Comparison of Pruning Methods for Decision Tree Induction. Machine Learning, 4, pp. 227-243
6. Bramer, M.A. (2002). Using J-Pruning to Reduce Overfitting in Classification Trees. In: Research and Development in Intelligent Systems XVIII. Springer-Verlag, pp. 25-38.
7. Smyth, P. and Goodman, R.M. (1991). Rule Induction Using Information Theory. In: Piatetsky-Shapiro, G. and Frawley, W.J. (eds.), Knowledge Discovery in Databases. AAAI Press, pp. 159-176