UNKNOWN ATTRIBUTE VALUES IN INDUCTION

J. R. Quinlan
Basser Department of Computer Science
University of Sydney
Sydney NSW Australia 2006

ABSTRACT

Simple techniques for the development and use of decision tree classifiers assume that all attribute values of all cases are available. Numerous approaches have been proposed with the aim of extending these techniques to cover real-world situations in which unknown attribute values are not uncommon. This paper compares the effectiveness of several approaches as measured by their performance on a collection of datasets.

INTRODUCTION

The 'standard' technique for constructing a decision tree classifier from a *training set* of *cases* with known classes, each described in terms of fixed *attributes*, can be summarised as follows:

- If all training cases belong to a single class, the tree is a leaf labelled with that class.
- Otherwise,
  - select a test, based on one attribute, with mutually exclusive outcomes;
  - divide the training set into subsets, each corresponding to one outcome; and
  - apply the same procedure to each subset

Once constructed, such a decision tree can be used to classify a new, unseen case described in terms of the same attributes. We start with the root of the tree. If the current node is a leaf, the case is assigned to the class associated with that leaf. Otherwise, the outcome of the test at the current node is determined and we follow the corresponding branch of the tree.

In real-world applications it is not unusual to encounter cases, some of whose attribute values are not known. This causes three problems for the procedures sketched above:

- The selection of a test to partition the training set may require comparison of tests based on attributes with different numbers of unknown values. How should this comparison be made in a sensible manner?
- Once a test has been selected (based on attribute A, say), training cases with unknown values of A cannot be associated with any one outcome of the test. How should these cases be treated in the division of the training set into subsets?
- When the decision tree is used to classify an unseen case, how should we proceed when we encounter a test on an attribute whose value is not known?

This paper evaluates several methods of circumventing these problems through controlled experiments on small variations (Buchanan, 1989). We start with a description of the datasets used in the trials.

DESCRIPTION OF DATASETS

In the following, the *unknown rate* of some attribute over a set of cases means the proportion of those cases whose value of that attribute is unknown. To *apply an unknown rate of x* to some attribute, we examine each case in the set and, with probability x, replace the value of the attribute with '?'.

Breiman et al (1984) report experiments with a system called CART. One domain involved recognising digits on a faulty 7-element LED display, each element of which has 10% probability of having the wrong on/off status. Using a training set of 200 cases, Breiman et al observed the effect on CART's retrial accuracy of applying various unknown rates to all seven attributes. The first dataset consists of their training set and a randomly-generated test set of 5000 cases, with an unknown rate of 25% applied to all attributes.

Breiman et al found that, for this induction task, the reduced accuracy was almost entirely due to unknown values in the test set. Two further datasets for this domain were derived from the above so as to highlight the effect of unknown values on the tree-construction process, especially when attributes have different unknown rates. The *step* variant uses a training set of 100 cases and applies an unknown rate of 50% to only four of the attributes. The *slope* variant also uses 100 training cases, but applies an unknown rate of 10% to the first attribute, 20% to the second, ... , 70% to the last.

The fourth dataset is a corrupted version of a *chess endgame* domain. There are two classes and 39 binary-valued attributes with an unknown rate of 25% applied to half of them. Training and test sets number 367 and 184 cases respectively.

The remaining datasets are all from real-world domains in which unknown values occur frequently. Location of *primary tumor* has 22 classes and 19 attributes, two of which have unknown rates of 46% and 10% respectively. There are 237 training cases and 102 test cases. The *sick euthyroid* dataset is a stratified sample from a thyroid assay domain in which the five key hormone measurements have been categorised as high, normal or low, and have unknown rates varying from 4% to 15%. This dataset has 462 training cases and 231 test cases. The *auto insurance* data has 25 attributes and 6 classes, with a moderate level of unknown values of some attributes; there are 100 training cases and 105 test cases.

For each dataset, 10 training and test sets were generated either by reapplying unknown rates (the LED datasets) or by randomly dividing the available data into training and test sets (the others).

DESCRIPTION OF APPROACHES

All the approaches described here were implemented as variants of a single tree-building program that uses *gain ratio*, an information-based heuristic, to select tests (Quinlan, 1986). The trees produced in these experiments were not *pruned* (Quinlan, 1987b).

Several methods of overcoming the three problems have been explored. Each of them has an identifying letter, so that a 'package' can be described succinctly by three letters denoting its approach to each problem.

- When evaluating a test based on attribute A,
  - I - Ignore cases in the training set with unknown values of A (Friedman, 1977; Breiman et al, 1984).
  - R - Reduce the apparent information gain from testing A by the proportion of cases with unknown values of A. The rationale for this reduction is that, if A has an unknown rate of x%, testing A will yield no information x% of the time.
  - S - "Fill in" the missing values of A before calculating the gain of A (Shapiro, 1983). Shapiro's method builds a decision tree for each attribute that attempts to determine a case's value of that attribute in terms of the values of other attributes (Quinlan, 1986). The method of *surrogate splits* (Breiman et al, 1984) may be viewed as a special case of this approach.
  - C - Similarly, fill in unknown values of A with its most common known value before calculating gain (Clark and Niblett, 1989).

- When partitioning the training set using a test on attribute A and a training case has unknown value of A,
  - I - Ignore this case (Quinlan, 1986).
  - S - Determine the likely value of A using Shapiro's method and assign it to the corresponding subset.
  - C - Treat this case as if it had the most common value of A.
  - P - Assign the case to one of the subsets with probability proportional to the number of cases with known value in each subset.
  - F - Assign a fraction of this case to each subset, using the proportions above (Kononenko et al, 1984).
  - A - Include the training case in *all* subsets (Friedman, 1977).
  - U - Develop a separate branch of the tree for cases with unknown values of attribute A.
- When classifying a new case with unknown value of a tested attribute A,
  - U - If there is a special branch for *unknown value of* A, take it.
  - S - Determine the most likely outcome of the test as above, and act accordingly.
  - C - Treat this case as if it had the most common value of A.
  - F - Explore all branches, combining the results to reflect the relative probabilities of the different outcomes (Quinlan, 1987a).
  - H - Halt at this point and assign the case to the most likely class.

Needless to say, not all of the possible combinations of these methods make sense.

UNKNOWN VALUES WHEN PARTITIONING

Seven packages that differ principally in their approach to partitioning were evaluated as follows. For each dataset, each of the ten training sets was used to construct trees whose error rates on the corresponding test set were measured. The means of the error rates on the test cases and the standard errors of the sample means are shown in Table 1.

An analysis of significant differences between packages brings out some interesting patterns. For each dataset, the results from each pair of packages were analysed to determine when one package was performing significantly better than the other.[1] The results of these significance tests are summarised in Table 2 which shows, for each package p, the number of packages significantly worse (p^+) and better (p^-) than p on each dataset. The entries have been sorted in terms of a rough index of merit, $p^+ - p^-$, and reveal the very clear superiority of RFF (assigning fractional cases to subsets) and the equally clear undesirability of RIF (ignoring training cases with unknown values of the test attribute) on these datasets.

UNKNOWN VALUES WHEN CLASSIFYING

A similar set of experiments was used to examine alternative approaches when a case to be classified has an unknown value of a tested attribute.

Two trees were constructed for each training set using reduced gain for assessing tests and then replacement (by the Shapiro tree or most common value respectively) when partitioning. The cases in the corresponding test set were then classified by both trees using three different strategies on encountering an unknown value

[1] For the 10 test sets in each dataset, we used the one-tailed Student t-test on pair differences with a 5% confidence level.

	LED (orig)	LED (step)	LED (slope)	chess endgame	primary tumor	sick euthyroid	auto insurance
RFF	43.8±0.3	49.4±0.6	62.7±0.6	15.0±0.6	60.2±1.5	3.8±0.3	33.2±1.8
RCF	45.9±0.4	50.2±0.3	62.0±0.6	14.6±0.8	60.7±1.3	3.2±0.3	37.3±1.5
RAF	43.9±0.4	51.9±0.6	62.7±0.4	17.2±0.6	60.1±1.4	5.0±0.5	33.2±1.9
RSS	44.7±0.5	51.6±0.7	62.7±0.8	13.4±0.8	60.9±1.3	4.6±0.4	36.6±1.3
RUU	44.5±0.3	51.5±0.9	62.3±0.6	17.3±0.9	61.1±1.3	4.8±0.4	34.6±1.5
RPF	47.1±0.5	51.7±0.9	63.5±0.9	16.4±1.1	60.7±1.3	3.9±0.6	34.9±1.8
RIF	50.8±0.9	55.0±1.1	66.6±1.2	19.0±1.1	61.2±1.2	3.7±0.4	36.3±1.2

Table 1. Partitioning: average error rates over ten trials

	LED (orig)		LED (step)		LED (slope)		chess endgame		primary tumor		sick euthyr'd		auto insur'ce		Total	
p	p^+	p^-	p^+	p^-	p^+	p^-	p^+	p^-	p^+	p^-	p^+	p^-	p^+	p^-	p^+	p^-
RFF	3		5		6		3		3		3		2		25	0
RCF	2	4	3		1	1	4		1	1	4			3	15	9
RAF	3		1	2	1	1		3	4			4	2		11	10
RSS	3		1	2	1	1	4					3		3	9	9
RUU	3		1	1	1	1	1	3		2		3	3		9	10
RPF	1	5	1	1	1	1	1	2		2	1				5	11
RIF		6		6		6		5		3	3	1		1	3	28

Table 2. Partitioning: significance comparison

	LED (orig)		LED (step)		LED (slope)		chess endgame		primary tumor		sick euthyr'd		auto insur'ce		Total	
p	p^+	p^-	p^+	p^-	p^+	p^-	p^+	p^-	p^+	p^-	p^+	p^-	p^+	p^-	p^+	p^-
RSF	1	1	1		2		1					1	2		7	2
RSS	2		1		1	1	1					1		2	5	4
RSH		2		2		2		2			2		1	1	3	9
RCF	2		2		2		2						1		9	0
RCC	1	1	1	1	1	1	1	1					1		5	4
RCH		2		2		2		2						2	0	10

Table 3. Classifying: significance comparison of two groups

of a tested attribute: make the best classification possible at this stage; replace the value; or follow multiple paths in the tree. Significance tests on the error rate differences were carried out as before.

Results of the significance tests are summarised in Table 3. The performance of the groups of packages is clearly differentiated on the LED datasets with their high unknown rates, but less so on the others. Overall, though, the strategy of halting on an unknown value emerges as a clear loser, while that of following multiple paths is probably better.

UNKNOWN VALUES IN SELECTING TESTS

The final set of trials concerns the effect of unknown values when selecting a test to partition the training set. Two groups of packages were investigated, each employing a constant approach to the treatment of unknown values in partitioning and classification. The information gain attributable to a test on attribute A was

	LED		LED		LED		chess		primary		sick		auto		Total	
	(orig)		(step)		(slope)		endgame		tumor		euthyr'd		insur'ce			
p	p^+	p^-	p^+	p^-	p^+	p^-	p^+	p^-	p^+	p^-	p^+	p^-	p^+	p^-	p^+	p^-
CMF		1	1		1				1						3	1
RMF	1		1		1					2					3	2
IMF				2		2			1						1	4
CSS			1		1	1			1						3	1
RSS			1		2					1					3	1
ISS				2		2									0	4

Table 4. Selecting tests: significance comparison of two groups

assessed under three rubrics: using only the training cases with known values of A; ditto, but multiplying the apparent gain by the proportion of cases with known values of A; and filling in unknown values before assessing the gain.

A similar experimental procedure was followed, giving the significance results in Table 4. Ignoring unknown values comes out as (weakly) inferior to reducing gain or filling in values, but no conclusion can be drawn regarding the relative desirability of these latter two.

CONCLUSIONS

This study has focussed on domains with relatively high levels of unknown values and small training sets. The investigation of a variety of strategies on several such domains, some constructed to highlight differences and some using real-world data, has provided evidence for the following hypotheses:

- In test evaluation, approaches that ignore cases with unknown values (and thus do not take account of unknown rates) perform badly when this rate varies markedly from attribute to attribute.
- When the training set is partitioned, ignoring cases with unknown values of the tested attribute leads to very inferior performance (a bitter pill to swallow, as this is how ID3 (Quinlan, 1986) handles partitioning!) The approach of dividing such cases among the subsets was found to perform well.
- During classification, attempting to determine the most likely outcome of a test works well in some domains (those in which such replacement can be performed reliably), but poorly in others. Combining all possible outcomes is more resilient, giving better overall classification accuracy in these domains.

Acknowledgement

The research reported here was supported by a grant from the Australian Research Council.

References

Breiman, L., Friedman, J.H., Olshen, R.A., and Stone, C.J. (1984), *Classification and regression trees*, Belmont: Wadsworth.

Buchanan, B.G. (1989), What do expert systems offer the science of AI?, in Quinlan (Ed), *Applications of Expert Systems (Vol 2)*, Wokingham: Addison-Wesley.

Clark and Niblett (1989), The CN2 induction algorithm, *Machine Learning*, to appear.

Friedman, J.H. (1977), A recursive partitioning decision rule for nonparametric classification, *IEEE Transactions on Computers*, pp 404-408.

Kononenko, I., Bratko, I. and Roškar (1984), Experiments in automatic learning of medical disgnostic rules, Technical Report, Jozef Stefan Institute, Ljubljana.

Quinlan, J.R. (1986), Induction of decision trees, *Machine Learning 1*, 1.

Quinlan, J.R. (1987a), Decision trees as probabilistic classifiers, in Langley(Ed), Proceedings of the Fourth International Workshop on Machine Learning, Los Altos: Morgan Kaufmann.

Quinlan, J.R. (1987b), Simplifying decision trees, *Int Journal of Man-Machine Studies 27*, 221-234.

Shapiro, A. (1983), private communication.