Good methods for coping with missing data in decision trees

B.E.T.H. Twala[1], M.C. Jones[2,*], D.J. Hand[3]

[1]*Methodology & Standards Division, Statistics South Africa, Pretoria 0001, South Africa*
[2]*Department of Statistics, The Open University, Milton Keynes MK7 6AA, UK*
[3]*Department of Mathematics, Imperial College of Science, Technology and Medicine, London SW7 2AZ, UK*

*Corresponding author. Fax: +44 1908 655515. E-mail address: m.c.jones@open.ac.uk

Abstract

We propose a simple and effective method for dealing with missing data in decision trees used for classification. We call this approach "missingness incorporated in attributes" (MIA). It is very closely related to the technique of treating "missing" as a category in its own right, generalizing it for use with continuous as well as categorical variables. We show through a substantial data-based study of classification accuracy that MIA exhibits consistently good performance across a broad range of data types and of sources and amounts of missingness. It is competitive with the best of the rest (particularly, a multiple imputation EM algorithm method; EMMI) while being conceptually and computationally simpler. A simple combination of MIA and EMMI is slower but even more accurate.

Keywords: C4.5; C5.0; CART; EM algorithm; Fractional cases; Missingness as attribute; Multiple imputation.

1. Introduction

Decision trees provide a simple yet effective methodology for classification and prediction (e.g. Breiman et al., 1984, Quinlan, 1986, 1993). They are, therefore, popular in both statistics and machine learning and have generated a vast literature, particularly in the latter area. On the other hand, real datasets are almost synonymous with "datasets involving missing data" and another vast literature, particularly in statistics (e.g. Little and Rubin, 2002), is concerned with coping with missing data. The intersection of the two literatures – to which this paper contributes – is, however, relatively small. A review and comparison of existing methods for coping with missing data in decision trees is given in Twala (2005, 2007). Twala found an implementation of multiple imputation using an EM algorithm due to Schafer (1997; henceforth EMMI) to be consistently the best of the existing methods investigated.

In Section 2 of this paper, we introduce a simple and natural alternative method for coping with missing data in decision trees used for classification. We call this approach "missingness incorporated in attributes" or MIA for short. (It is very closely related to, but differs slightly from, the approach of treating "missing" as a category in its own right.) In Section 3, we compare MIA with EMMI within the broad data-based comparisons of Twala (2005, 2007). We find that the simple and speedy MIA approach typically performs almost as well as the complex and cumbersome EMMI approach in terms of classification accuracy and is sometimes the better of the two. In Section 4, we consider two very simple combinations of MIA and EMMI (EMIMIA and REMIMIA) and present the results of using them too. The paper closes with brief conclusions and discussion in Section 5. Throughout, we consider only binary decision trees in the sense that branches of the tree are only ever split into two parts, although it will be obvious that MIA could be extended to multiway splitting too.

2. The MIA approach

Let X be an attribute for which a split is currently being sought. MIA treats individuals with missingness in X as a single group, defining splits to contain all such individuals in one or other part of the split, together with allowing splits on missingness per se. Concretely, let Y be a subset of X corresponding to one part of a putative split; if X is an ordered or numeric attribute, Y is the set of all values of X of the form $\{X : X \leq x\}$ for some split-point x; if X is a nominal attribute, Y is simply some subset of the values of X. Then, in choosing a split, we choose between the following options as well as varying Y:

Split A: $\{X \in Y \text{ or } X \text{ is missing}\}$ versus $\{X \notin Y\}$;
Split B: $\{X \in Y\}$ versus $\{X \notin Y \text{ or } X \text{ is missing}\}$;
Split C: X is missing versus X is not missing.

So, if there were o options for splitting a branch without missingness, there are $2o+1$ options to be explored with missingness present.

This MIA algorithm is very simple and natural and applicable to any method of constructing decision trees, regardless of that method's detailed splitting/stopping/ pruning rules. It has a very close antecedent: the approach of handling unknown attribute values by treating all attributes as categorical and adding missingness as a further category. The two approaches are the same for categorical attributes, but differ a little in their treatment of continuous attributes: rather than categorizing continuous variables, we incorporate missingness directly in splits of continuous variables. The "missingness as category" approach was, on the basis of a single artificial example and, in our view, prematurely, dismissed by Quinlan (1986, pp. 97–98); see Section 5 for discussion. It has, nonetheless, been used since (Hastie et al., 2001, Section 9.2.4). Both approaches can be expected to be particularly useful when missingness is not random but informative. Classifying a new individual whose value of a branching attribute is missing is immediate provided there was missingness in that attribute in the training set that led to the decision tree. In the remainder of the paper, we show that MIA can be an extremely effective method for coping with missing data in decision trees.

3. Experimental setup and results

3.1. Experimental set-up

We add MIA to the experiment reported by Twala (2007), to which the reader should refer for implementation details of the outline description given here.

The experiment was based on a suite of 21 datasets taken from the Repository of Machine Learning Databases provided by the Department of Information and Computer Science at the University of California at Irvine (Newman et al., 1998). See Table 1 of Twala (2007) for details of the specific datasets used: they range from 57 to 20,000 in terms of sample size, from 4 to 60 in terms of numbers of attributes, from 2 to 26 in terms of numbers of classes, and display a mix of numerical and nominal attributes. All are complete datasets into which missingness is artificially introduced at rates of 15%, 30% and 50% into either the single attribute which is most highly correlated with class or else evenly distributed across all attributes. Three missing data mechanisms were employed, coming under the headings (Little and Rubin, 2002) of: missing completely at random (MCAR); missing at random (MAR), under which the probability of being missing depends on the value of another, non-missing, attribute; and informative missingness (IM), under which the probability of being missing depends on the actual (but in non-simulation practice unobserved) value of the attribute itself. The six combinations of missingness mechanisms and distribution will be referred to as MCARuniva, MARuniva and IMuniva when missingness is in a single attribute and as MCARunifo, MARunifo and IMunifo when missingness is spread uniformly across all attributes.

Performance of methods is measured by the excess classification error rate, that is, the difference between the error rate observed in cases incorporating missingness and the

error rate observed for the complete dataset. (A smoothed error rate was used to cope with ties between competing classes.) Five-fold cross-validation was used to provide separate training and test sets from each complete dataset. (Note, therefore, that missing values occur in both training and test data.) Decision trees on complete training data were grown using the *Tree* function in S-PLUS (Becker et al., 1988, Venables and Ripley, 1999) which in turn uses the *GINI* impurity index (Breiman et al., 1984) as splitting rule and cross-validation cost-complexity pruning.

Twala (2007) compared the performances of seven methods for coping with missing data in decision trees. These were: deletion of instances with any missing data; Shapiro's decision tree single imputation technique (Quinlan, 1993); maximum likelihood imputation of data for both continuous (assuming multivariate normality) and categorical data via the EM algorithm as developed by Schafer (1997), and considered in both single and (five replicate) multiple imputation (EMMI) forms; mean or mode single imputation; fractional cases (FC; Cestnik et al., 1987, Quinlan, 1993); and surrogate variable splitting (Breiman et al., 1984, Therneau and Atkinson, 1997). To cut a long story short, EMMI proved to be the overall best of the seven techniques and FC the second best, and it is only these two best techniques with which we directly compare MIA in the results that follow. (Note that for the largest datasets, we modified Schafer's version of EMMI by splitting up the dataset in order for it to run in a reasonable time.)

3.2. Experimental results

Fig. 1 shows the excess error rates, averaged over the 21 datasets, attained by MIA, EMMI and FC; the six frames of Fig. 1 refer to the missingness mechanism/ distribution combinations described in Section 3.1. One can readily observe the consistently superior performance of EMMI over FC. When missingness is in only one attribute (the `univa' cases) MIA is broadly on a par with FC under the MCAR mechanism, intermediate between FC and EMMI under MAR and comparable to EMMI under IM. When missingness is spread across all attributes (the `unifo' cases), the performance of MIA improves relative to the other two methods. Indeed, in these important cases, MIA is broadly comparable with EMMI and superior to it when the missingness is informative. (In further averaging of error rates over missingness mechanisms, distributions and percentages, but still for these particular 21 datasets, MIA takes second place to EMMI, only a statistically insignificant amount ahead of FC; see Fig. 5.5 of Twala, 2005.)

Further investigation of performance on individual datasets within the collection suggests that MIA is especially effective (and superior to both EMMI and FC) for datasets consisting primarily of nominal, as opposed to quantitative attributes. A case in point is the `kr-vs-kp' chess dataset of A. Shapiro. This consists of 3196 instances each having 35 binary attributes and one further nominal attribute with three categories; there are just two classes. Results are shown for this dataset in Fig. 2. As for any individual dataset, one has to be wary of reading too much into the results.

Fig. 1. Excess error rates attained by MIA, EMMI and FC, averaged over the 21 datasets and plotted against percentages of missing values. The six frames correspond to MCARuniva, MCARunifo, MARuniva, MARunifo, IMuniva and IMunifo missingness mechanism/distribution combinations.

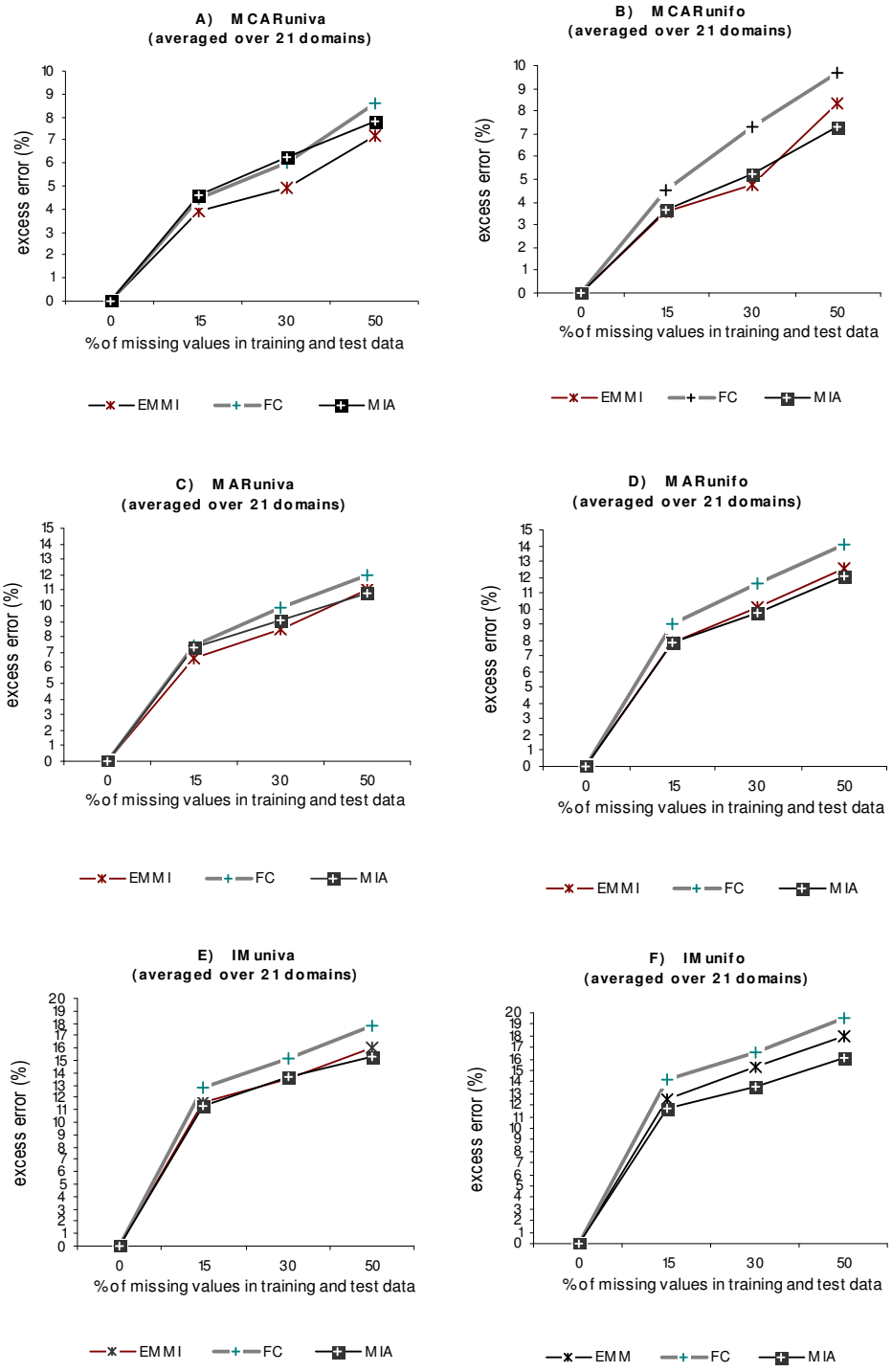
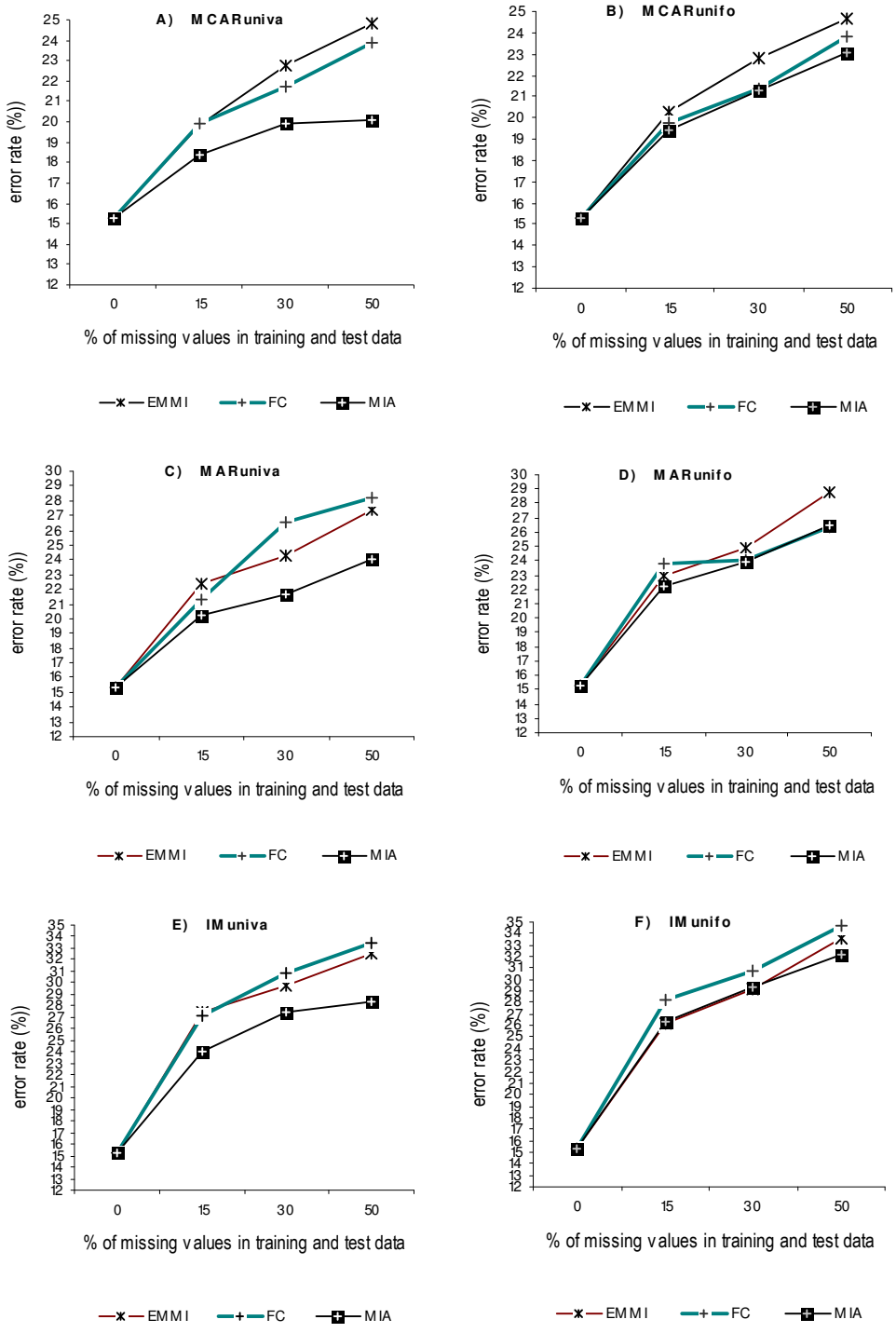


Fig. 2. Excess error rates attained by MIA, EMMI and FC for the `kr-vs-kp' dataset, plotted against percentages of missing values. The six frames correspond to MCARuniva, MCARunifo, MARuniva, MARunifo, IMuniva and IMunifo missingness mechanism/distribution combinations.



Here, the 'bucked trend' is that MIA is more effective when missingness is in only one variable than in all. This appears to be because the one variable on which missingness occurs is the single non-binary one.

In limited timing experiments involving the largest datasets in our simulation testbed, the most complex missingness, and full training and classification, MIA proves to be no faster than (the inferior performing) FC, but gives savings of around 25-35% over EMMI. It seems, therefore, that MIA is a quicker and much more readily comprehensible method that is competitive with what we believe to be the best of the current methods for dealing with missing data in decision trees, namely EMMI.

4. EMIMIA and REMIMIA

4.1. The methods

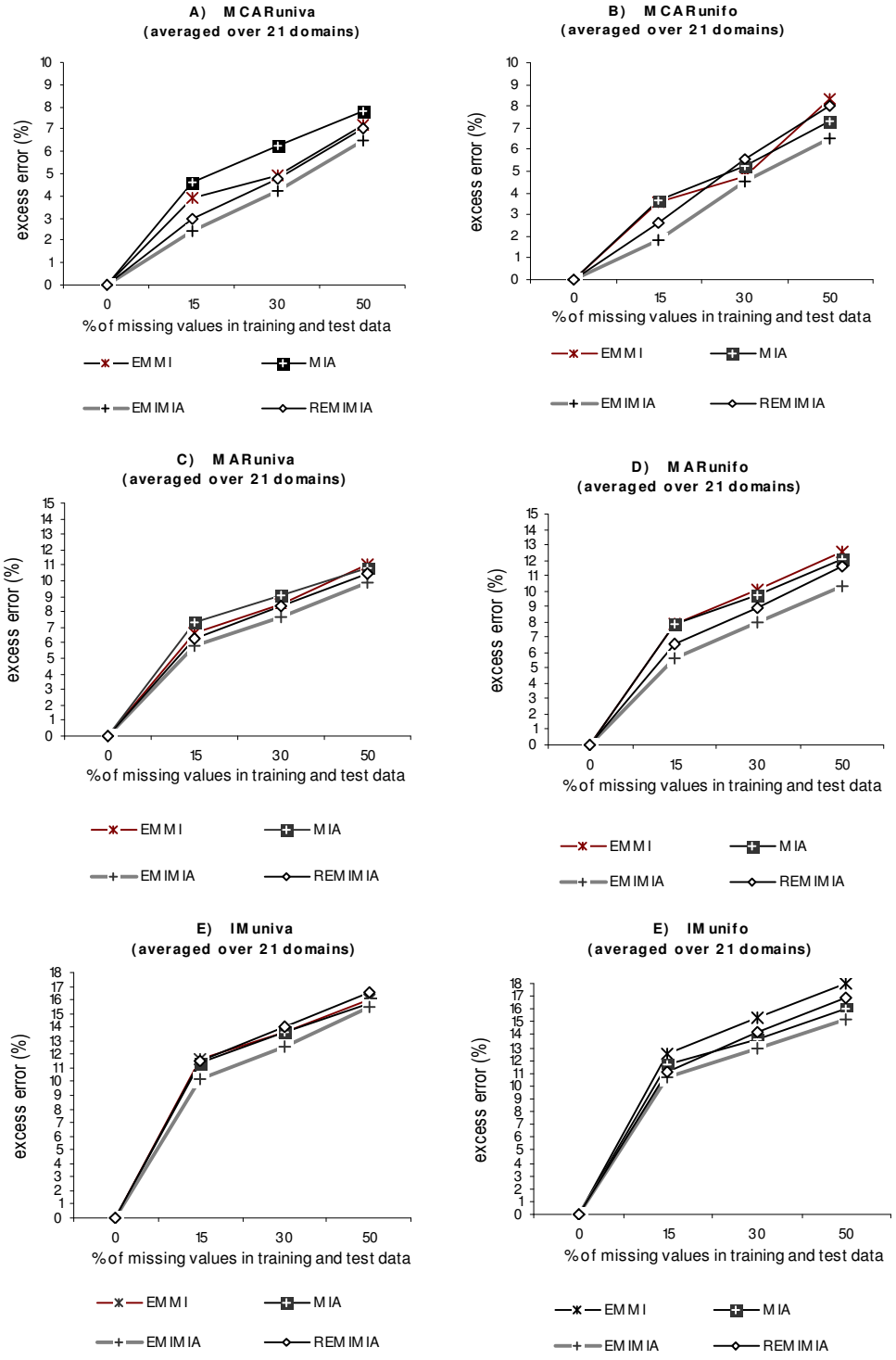
Can we do even better with simple combinations of these two successful methods (or extremely simple 'ensembles'), perhaps attaining performance equal to the best of either singly? We briefly investigate two such combinations. The first of these is EMIMIA (standing for Ensemble Multiple Imputation and Missing Incorporated in Attributes). In both EMMI and MIA, class predictions are made on the basis of estimated class probabilities. In EMIMIA, run both EMMI and MIA on your data. When the two yield the same classification for an instance, as they will usually do, take that answer. When the two suggest different classifications for an instance, choose the class has the higher probability assigned by the two methods. (If the two assign different classes with the same probability choose randomly between them.)

The most obvious disadvantage of this technique is the computational cost of having to run both EMMI and MIA. To alleviate this, we also consider REMIMIA (Resampling Ensemble Multiple Imputation and Missing Incorporated in Attributes). In this version, the training data is randomly divided into two halves and EMMI applied to one half and MIA to the other. Then the results of each are combined in the same way as for EMIMIA.

4.2. Experimental results

We simply add results for EMIMIA and REMIMIA to those of Section 3.2 and drop those of FC for clarity. See Fig. 3. A pleasingly consistent improved performance by EMIMIA relative to EMMI and MIA singly is observed. Of course, there is a computational cost involved which turns out to be an increase of around 70% relative to EMMI. REMIMIA, on the other hand, is more 'in the same ballpark' as EMMI and MIA, although it too improves on them, a little, sometimes. Its computational cost remains quite high, however, and greater than EMMI alone. In terms of overall mean excess errors, EMIMIA is best (with average error rate of 9.2%), followed by EMMI and REMIMIA, closely together on 9.8% and 10.2%, respectively, and finally, but not too far behind, MIA with 10.7%.

Fig. 3. Excess error rates attained by EMIMIA and REMIMIA as well as MIA and EMMI, averaged over the 21 datasets and plotted against percentages of missing values. The six frames correspond to MCARuniva, MCARunifo, MARuniva, MARunifo, IMuniva and IMunifo missingness mechanism/distribution combinations.



5. Conclusions and discussion

We have put forward the method of MIA as a conceptually and computationally simple method for dealing with missing data in decision trees when classification is the goal. It is closely related to treating "missing as category" per se, generalizing that approach for use with continuous as well as categorical variables. MIA shares with EMMI consistently good performance across a broad range of data types and of sources and amounts of missingness. We recommend its use if, as well as excellent classification accuracy, it is desirable that the end-user understand the methodology employed. If simplicity is less of a concern, EMMI is outstanding, but then the combination of EMMI and MIA through EMIMIA is to be recommended.

We have already mentioned that "missing as category" was summarily dismissed by Quinlan (1986, pp.97-98). Quinlan set up a very simple example involving a binary attribute and then removed knowledge of one (out of four) values. His (correct) calculations show a higher information gain in the latter case than the former. It was concluded that, since "having unknown values may apparently increase the desirability of an attribute," this is "a result entirely opposed to common sense" and hence that "treating 'unknown' as a separate value is not a solution to the problem". Our alternative view is that missingness is informative in that case, in the sense that the single missing value is associated with a particular class: the method can be envisaged as taking it that missingness is a strong indicator of that class. (If we had more data, this would either be confirmed by a strong correlation between missingness and class which we could make use of or else debunked – but in the latter case, a mixture of classes associated with missingness would not, we imagine, lead to an increased information gain.) We believe our substantial study to have shown that MIA actually shows very promising results and should be taken seriously.

References

Becker, R.A., Chambers, J.M., Wilks, A.R., 1988. The New S Language: A Programming Environment for Data Analysis and Graphics. Chapman and Hall, New York.

Breiman, L., Friedman, J.H., Olshen, R.A., Stone, C.J., 1984. Classification and Regression Trees. Wadsworth, Pacific Grove, CA.

Cestnik, B., Kononenko, I., Bratko, I., 1987. Assistant 86: a knowledge-elicitation tool for sophisticated users, in Bratko, I., Lavrac, N. (Eds.), European Working Session on Learning – EWSL87. Sigma, Wilmslow, UK.

Hastie, T., Tibshirani, R., Friedman, J.H., 2001. The Elements of Statistical Learning: Data mining, Inference, and Prediction. Springer, New York.

Little, R.J.A., Rubin, D.B., 2002. Statistical Analysis with Missing Data, second ed. Wiley, New York.

Newman, D.J., Hettich, S., Blake, C.L., Merz, C.J., 1998. UCI Repository of Machine Learning Databases, http://www.ics.uci.edu/~mlearn/MLRepository.html.

Quinlan, J.R., 1986. Induction of decision trees. *Mach. Learn.* 1, 81−106.

Quinlan, J.R., 1993. C4.5: Programs for Machine Learning. Morgan Kaufmann, Los Altos, CA.

Schafer, J.L., 1997. Analysis of Incomplete Multivariate Data. Chapman and Hall, London.

Therneau, T.M., Atkinson, E.J., 1997. An Introduction to Recursive Partioning Using the RPART Routines. Technical Report, Mayo Foundation.

Twala, B.E.T.H., 2005. Effective Techniques for Handling Incomplete Data Using Decision Trees. Ph.D. thesis, Department of Statistics, The Open University, UK.

Twala, B.E.T.H., 2007. An empirical comparison of techniques for handling incomplete data when using decision trees. Under revision.

Venables, W.N., Ripley, B.D., 1999. Modern Applied Statistics With S-Plus, third ed. Springer, New York.