Zeta: A Global Method for Discretization of Continuous Variables

K. M. Ho and P. D. Scott

Department of Computer Science,
University of Essex, Colchester, CO4 3SQ, UK
hokokx@essex.ac.uk and *scotp@essex.ac.uk*

Abstract

This paper introduces a new technique for discretization of continuous variables based on *zeta*, a measure of strength of association between nominal variables developed for this purpose. Zeta is defined as the maximal accuracy achievable if each value of an independent variable must predict a different value of a dependent variable. We describe both how a continuous variable may be dichotomised by searching for a maximum value of zeta, and how a heuristic extension of this method can partition a continuous variable into more than two categories. Experimental comparisons with other published methods, show that zeta-discretization runs considerably faster than other techniques without any loss of accuracy.

Introduction

Many machine learning techniques can only be applied to data sets composed entirely of nominal variables but a very large proportion of real data sets include continuous variables. One solution to this problem is to partition numeric variables into a number of sub-ranges and treat each such sub-range as a category. This process of partitioning continuous variables into categories is usually termed *discretization*. In this paper we describe a new technique for discretization of continuous variables based on *zeta*, a measure of strength of association that we have developed for this purpose.

Discretization for Decision Tree Construction

Procedures for constructing decision trees using sets of pre-classified examples (Breiman, Friedman, Olshen & Stone 1984; Quinlan 1986) are inherently only applicable to data sets composed entirely of nominal variables. If they are to be applied to continuous variables some means must be found of partitioning the range of values taken by a continuous variable into sub-ranges which can be treated as discrete categories. Such a partitioning process is frequently termed *discretization*. A variety of discretization methods have been developed in recent years. Dougherty, Kohavi and Sahami (1995) have provided a systematic review in which discretization techniques are located along two dimensions: *unsupervised* vs. *supervised*, and *global* vs. *local*.

Unsupervised discretization procedures partition a variable using only information about the distribution of values of that variable: in contrast, supervised procedures also use the classification label of each example. Typical unsupervised techniques include equal interval width and equal frequency width methods.

Supervised techniques normally attempt to maximise some measure of the relationship between the partitioned variable and the classification label. Entropy or information gain is often used to measure the strength of the relationship (Quinlan 1986, 1993; Catlett 1991; Fayyad & Irani 1992, 1993). Both ChiMerge (Kerber 1992) and StatDisc (Richeldi & Rossotto 1995) employ procedures similar to agglomerative hierarchical clustering techniques. Holte's (1993) 1R attempts to form partitions such that each group contains a large majority of a single classification.

Global discretization procedures are applied once to the entire data set before the process of building the decision tree begins; local discretization procedures are applied to the subsets of examples associated with the nodes of the tree during tree construction. The majority of systems using unsupervised methods carry out global discretizations. Examples of supervised global methods include D-2 (Catlett 1991), ChiMerge (Kerber 1992), Holte's (1993) 1R method, and StatDisc (Richeldi & Rossotto 1995). C4.5 (Quinlan 1993,1996) and Fayyad and Irani's (1993) entropy minimisation method use a supervised technique to perform local discretization. The majority of supervised techniques could be used for either local or global discretization: for example, Fayyad and Irani's method has been successfully employed to form global discretizations (Ting 1994).

Dougherty *et al.* (1995) report a comparative study of two unsupervised global methods (equal width interval procedures), two supervised global methods (1RD (Holte 1993) and Fayyad & Irani's (1993) entropy minimisation), and C4.5, a supervised local method. They found only small differences between the classification accuracies achieved by resulting decision trees. None produced the highest accuracy for all data sets but our own replication of these experiments (see below) shows a clear speed/accuracy trade-off.

	C4.5					
Data Set	Continuous	Entropy	1RD	Bin-log I	n-Bins	Zeta
allbp	97.45+-0.10	97.22+-0.16	96.05+-0.32	96.39+-0.32	96.32+-0.13	96.63+-0.23
ann-thyroid	99.61+-0.11	99.38+-0.13	97.64+-0.04	94.06+-0.19	92.72+-0.24	98.38+-0.16
australian	84.93+-0.81	85.65+-1.82	85.22+-1.35	84.06+-0.97	84.93+-0.77	86.38+-0.96
breast	94.28+-0.60	94.42+-0.89	95.13+-0.57	94.85+-1.28	94.85+-0.41	95.85+-0.89
cleve	79.23+-1.63	80.23+-3.25	80.24+-4.15	76.57+-2.60	76.91+-2.11	78.23+-2.37
crx	86.09+-1.11	84.78+-1.94	85.22+-1.93	84.78+-1.82	85.07+-1.80	84.93+-1.99
diabetes	72.66+-1.08	73.70+-0.78	70.45+-1.16	73.44+-1.07	64.85+-1.21	75.13+-1.32
german	71.30+-0.93	72.20+-1.23	70.00+-1.14	72.10+-0.99	71.80+-0.46	73.80+-1.21
glass2	81.00+-2.59	76.67+-1.63	71.23+-5.06	80.42+-3.55	66.86+-2.06	76.14+-1.63
heart	75.19+-1.91	78.52+-1.26	78.52+-0.74	80.74+-1.11	78.52+-1.72	77.41+-3.07
horse-colic	85.87+-1.32	85.60+-1.24	85.60+-1.24	85.33+-1.23	85.60+-1.25	86.15+-1.44
ionosphere	89.45+-1.41	91.15+-1.78	88.88+-1.67	88.60+-1.29	83.17+-2.21	89.72+-2.00
iris	94.00+-1.25	94.00+-1.25	94.00+-1.25	96.00+-1.25	73.33+-2.58	94.00+-1.25
vehicle	73.17+-0.95	68.68+-1.91	66.21+-3.07	68.45+-2.19	62.06+-1.42	69.27+-1.67
waveform-21	76.30+-0.53	74.58+-0.58	52.94+-0.43	70.36+-0.65	74.60+-0.87	76.44+-0.59
Average	84.04	83.79	81.16	83.08	79.44	83.90

Table 1: Classification accuracies and standard deviations using C4.5 (Quinlan 1996) with different discretization methods. Continuous: C4.5 on undiscretized data. Entropy: Global variant of Fayyad & Irani's (1993) method. 1RD: Holte's (1993) 1R discretizer. Bin-log l and n-Bins: Equal width binning. Zeta: Method proposed in this paper. (c.f. Dougherty et al. 1995)

Zeta: A New Measure of Association

Our initial attempts to develop a fast and accurate discretization technique based upon *lambda*, a widely used measure of strength of association between nominal variables (Healey 1990) that measures the proportionate reduction in prediction error that would be obtained by using one variable to predict the other, using a modal value prediction strategy in all cases. Unfortunately lambda is an ineffective measure in those situations were the dependency between two variables is not large enough to produce different modal predictions since in such cases its value is zero.

A closely related measure, which we term *zeta*, has been developed that overcomes this limitation because it is not based on a modal value prediction strategy: the assumption made in determining zeta is that each value of the independent variable will be used to predict a *different* value of the dependent variable.

Zeta is most readily understood by first considering the simplest case: using a dichotomous variable A to predict values of another dichotomous variable B. Suppose we have a sample of N items whose value distribution is given in the following 2 by 2 table:

	A_1	A_2
B_1	n_{11}	n_{12}
B_2	n_{21}	n_{22}

where

$$N = \sum_{i=1}^{2}\sum_{j=1}^{2} n_{ij}$$

If each value of A is used to predict a different value of B then there are only two possibilities: either $A_1 \rightarrow B_1$ and $A_2 \rightarrow B_2$,or $A_1 \rightarrow B_2$ and $A_2 \rightarrow B_1$. If the former is used then the number of correct prediction would be $n_{11} + n_{22}$; if the latter then $n_{12} + n_{21}$ would be correct. Zeta is defined to be the percentage accuracy that would be achieved if the pairings that lead to greater accuracy were used for prediction. Hence it is defined as follows:

$$Z = \frac{\max(n_{11}+n_{22}, n_{12}+n_{21}) \times 100\%}{N}$$

This definition may be generalised to the case of one k-valued variable A being used to predict the values of another variable B that has at least k values thus:

$$Z = \frac{\sum_{i=1}^{k} n_{f(i),i}}{N} \times 100\%$$

where f(i) should be understood as follows. In order to make predictions each of the k values of A must be paired with a non-empty set of values of B: these k sets must together form a partition of the set of possible values for B. If B has k distinct values there will be k! ways in which such sets of pairings could be made. One such set of pairing will give the greatest prediction accuracy; call this the *best pairing assignment.* Then $B_{f(i)}$ is the value of B that is paired with A_i in the best pairing assignment.

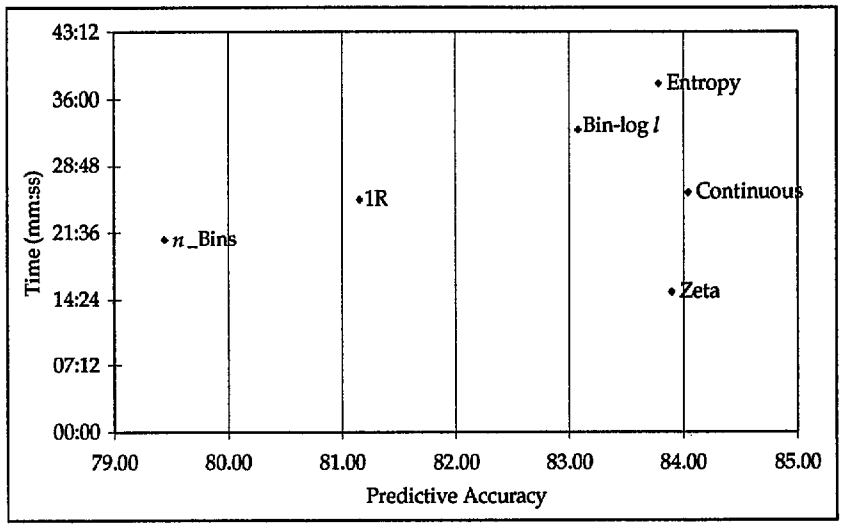

Figure 2: Total execution time for all data sets plotted as a function of average final classification accuracy for different discretization methods (see caption to Table 1).

Discretization Using Zeta

We now proceed to describe how this measure may be used to partition a continuous variable. The underlying principle is very simple. In theory, given a k-valued classification variable C, a continuous variable X could be partitioned into k sub-ranges by calculating zeta for each of the possible assignments of the k-1 cut points and selecting that combination of cut points that gives the largest value of zeta. In general such a method will not be practicable because the number of combinations of cut points is extremely large.

Dichotomising Using Zeta

However it is practicable for the special case of k = 2 since there will only be one cut point. Fortunately this is an extremely common special case in real world classification problems, many of which reduce to a choice between positive and negative diagnoses. If there are N examples in the data set there are at most N - 1 candidate cut points. Zeta can be calculated for every one of these and the point yielding the maximum value selected.

In practice it is not necessary to consider every possible cut point. Fayyad and Irani (1992) have shown that optimal cut points for entropy minimisation must lie between examples of different classes. A similar result can be proved for zeta maximisation. Hence it is only necessary to calculate zeta at points corresponding to transitions between classes, thus reducing the computational cost.

More Than Two Classes

The dichotomising procedure forms the basis of a heuristic method of discretizing a variable into k categories. This is a stepwise hill-climbing procedure that locates and fixes each cut point in turn. It therefore finds a good combination of cut points but offers no guarantee that it is the best. As noted above, examining all possible combinations is likely to be too time consuming.

The procedure for discretizing a variable A into k classes, given a classification variable B which takes k distinct values is as follows. First find the best dichotomy of A using the procedure described above. If k is greater than 2 then at least one of the resulting sub-ranges of A will be associated with 2 or more values of B: use the dichotomising procedure again on such a sub-range to place the second cut point. Proceed in a similar fashion until k - 1 cutpoints have been placed and each sub-range is associated with a different value of B.

This is a heuristic method: once a cut point is placed it is not moved so not all cut point combinations are considered. Nevertheless, as some of the results discussed later show, the cut points chosen lead to high predictive accuracy and hence the use of the heuristic is justified.

Experimental Results

A series of experiments were carried out using both real and artificial data sets to establish whether the zeta technique partitioned individual variable in a useful fashion. (see Ho and Scott 1997). These established that zeta discretization is an effective procedure for locating good cut points within the ranges of continuous variables.

The next set of experiments was designed to evaluate the performance of zeta in the role for which it was developed: the construction of decision trees. Our experimental procedure was closely modelled on that employed by Dougherty *et al.* (1995) in their comparative study of five discretization techniques.

We compared the five methods considered by Dougherty *et al.* and zeta discretization. C4.5 (Quinlan 1996) was used to construct all of the decision trees. In five of the six cases, the data was first processed by the global discretization procedure and then passed to C4.5. In the sixth case no prior discretization took place; hence the local discretization procedures that form part of C4.5 were used.

The code for zeta discretization was written in C by one of the authors (Ho); the code for C4.5, also written in C, was the version distributed to accompany Quinlan (1993) updated to Release 8 (Quinlan 1996); all the remaining code was taken from the MLC++ machine learning library (Kohavi, John, Long, Manley & Pfleger 1994). The data sets used for these experiments were all obtained from the UC Irvine repository. Each set was tested five times with each discretization method.

The results are shown in Table 1. As is to be expected the results for the first five columns are very similar to the results reported by Dougherty *et al.* (1995). The zeta discretization method stands up to the comparison very well. The average accuracy over all the data sets was higher than all the other global methods and only slightly, but not significantly, less than that achieved by C4.5 using local discretization. Thus we can conclude that on average zeta discretization method achieves accuracies at least as good as the best global methods.

However, the zeta method is also fast. Figure 2 shows the total execution time required by each of the six methods to complete all the data sets listed in Table 1, plotted as a function of final classification accuracy. It is clear that four of the six data points lie roughly in a straight line, indicating a time accuracy trade-off. Two points lie well below this line: continuous (i.e. C4.5's local method) and zeta. These two methods not only achieve high accuracy but do so in appreciably less time.

Conclusion

These results show that zeta discretization is both an effective and a computationally efficient method of partitioning continuous variables for use in decision trees. Indeed of the methods considered in our comparative study it would appear to be the method of choice.

Acknowledgements

We are grateful to the Economic and Social Research Council's programme on the Analysis of Large and Complex Datasets for supporting part of the work reported in this paper under grant number H519255030.

References

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

Catlett, J. 1991. On changing continuous attributes into ordered discrete attributes. In Machine Learning: EWSL-91, Proceedings European Working Session on Learning, Lecture Notes in Artificial Intelligence 482. pp. 164-178. Springer Verlag.

Dougherty, J., Kohavi, R., & Sahami, M. 1995. Supervised and Unsupervised Discretization of Continuous Features. In *Proc. Twelfth International Conference on Machine Learning.* Morgan Kaufmann, Los Altos, CA.

Fayyad, U. M., & Irani, K. B. 1992. On the handling of continuous-valued attributes in decision tree generation. *Machine Learning* **8** pp. 87-102.

Fayyad, U. M., & Irani, K. B. 1993. Multi-interval discretization of continuous-valued attributes for classification learning. In *Proc. 13th International Joint Conference on Artificial Intelligence.* pp 1022-1027 Morgan Kaufmann, Los Altos, CA.

Healey, J. 1990. *Statistics: A Tool for Social Research.* Wadsworth, Belmont, CA.

Ho, K. M. and Scott, P. D. 1997. Zeta: A Global Method for Discretization of Continuous Variables. Technical Report CSM-287, Dept of Computer Science, University of Essex, Colchester, UK.

Holte, R. C. 1993. Very simple classification rules perform well on most commonly used datasets. In *Machine Learning*, **11**, pp. 63-91.

Kerber, R. 1992. ChiMerge: Discretization of numeric attributes. In *Proc. Tenth National Conference on Artificial Intelligence*, pp. 123-128. MIT Press.

Kohavi, R., John, G., Long, R., Manley, D. & Pfleger, K. 1994. MLC++: A machine learning library in C++. In *Tools with Artificial Intelligence*, IEEE Computer Society Press, pp. 740-743.

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

Quinlan, J. R. 1993. *Programs for Machine Learning.* Morgan Kaufmann, Los Altos CA.

Quinlan, J. R. 1996. Improved use of continuous attributes in C4.5. *Journal of Artificial Intelligence Research* **4** pp. 77-90.

Richeldi, M. & Rossotto, M. 1995. Class-Driven statistical discretization of continuous attributes. In *Machine Learning: ECML-95 Proceedings European Conference on Machine Learning, Lecture Notes in Artificial Intelligence 914.* pp 335-338. Springer Verlag.

Ting, K. M. 1994. Discretization of continuous-valued attributes and instance-based learning. Technical Report 491, University of Sydney.

Van de Merckt, T. 1993. Decision trees in numerical attribute spaces. In *Proc. 13th International Joint Conference on Artificial Intelligence.* pp 1016-1021 Morgan Kaufmann, Los Altos, CA.