A Modified Chi2 Algorithm for Discretization

Francis E.H. Tay, *Member*, *IEEE*, and
Lixiang Shen

Abstract—Since the ChiMerge algorithm was first proposed by Kerber in 1992, it has become a widely used and discussed discretization method. The Chi2 algorithm is a modification to the ChiMerge method. It automates the discretization process by introducing an inconsistency rate as the stopping criterion and it automatically selects the significance value. In addition, it adds a finer phase aimed at feature selection to broaden the applications of the ChiMerge algorithm. However, the Chi2 algorithm does not consider the inaccuracy inherent in ChiMerge's merging criterion. The user-defined inconsistency rate also brings about inaccuracy to the discretization process. These two drawbacks are first discussed in this paper and modifications to overcome them are then proposed. By comparison, results with original Chi2 algorithm using C4.5, the modified Chi2 algorithm, performs better than the original Chi2 algorithm. It becomes a completely automatic discretization method.

Index Terms—Discretization, degree of freedom, χ^2 test.

1 Introduction

Many algorithms developed in the machine learning community focus on learning in nominal feature spaces [1]. However, many of such algorithms cannot be applied to the real-world classification tasks involving continuous features before these features are first discretized. This demands the studies on the discretization methods. There are three different axes by which discretization methods can be classified: global versus local, supervised versus unsupervised, and static versus dynamic [2]. A local method would discretize in a localized region of the instance space (i.e., a subset of instances) while a global discretization method uses the entire instance space to discretize [3]. Many discretization methods, such as equal-width-intervals and equal-frequency-intervals methods, do not use the class information during the discretization. These methods are called unsupervised methods. Likewise, those methods that make use of the class information are supervised methods. Many discretization methods require a parameter, k, indicating the maximal number of partition intervals in discretizing a feature. Static methods, such as Ent-MDLPC [4], perform the discretization on each feature and determine the value of k for each feature independent of the other features. However, the dynamic methods search through the space of possible k values for all features simultaneously, thereby capturing interdependencies in feature discretization.

The ChiMerge method was first proposed by Kerber [5] in 1992 to provide a statistically justified heuristic method for supervised discretization. The ChiMerge algorithm consists of an initialization step (i.e., placing each observed real value into its own interval) and proceeds by using the χ^2 test to determine when adjacent intervals should be merged. This bottom-up merging process is repeated until a stopping criterion (set manually) is met. Here, the χ^2 test is a statistical measure used to test the hypothesis that two discrete attributes are statistically independent. Applied to the discretization problem, it tests the hypothesis that the class attribute is independent of the two adjacent intervals an example belongs to. If the conclusion of the χ^2 test is that the class is independent of the intervals, then the intervals should be merged. On the other hand, if the χ^2 test concludes that they are not independent, it indicates that the difference in relative class frequencies is statistically significant and, therefore, the intervals should remain separate. The formula for computing the χ^2 value is:

$$\chi^2 = \sum_{i=1}^{2} \sum_{j=1}^{k} \frac{(A_{ij} - E_{ij})^2}{E_{ij}}, \quad (1)$$

where:

$\mathrm{k} =$ number of classes,
$\mathrm{A_{ij}} =$ number of patterns in the ith interval , jth class,
$\mathrm{R_i} =$ number of patterns in the ith interval $= \sum_{j=1}^{k} A_{ij}$,
$\mathrm{C_j} =$ number of patterns in the jth class $= \sum_{i=1}^{2} A_{ij}$,
$\mathrm{N} =$ total number of patterns $= \sum_{i=1}^{2} R_i$,
$\mathrm{E_{ij}} =$ expected frequency of A_{ij}; $E_{ij} = R_i * C_j / N$.

The value for the χ^2 threshold is determined by selecting a desired significance level α and then using a table or formula to obtain the corresponding χ^2 value. Obtaining the χ^2 value also requires specifying the number of degrees of freedom, which is one less than the number of classes. For example, when there are three classes, the degree of freedom is 2, the χ^2 value at $\alpha = 0.1$ level is 4.6. The meaning of this threshold is that among the cases where the class and attribute are independent, there is a 90 percent probability that the computed χ^2 value will be less than 4.6.

Liu and Setino [6] used the ChiMerge algorithm as a basis for their Chi2 algorithm. Specifying a proper value for α in many cases can be difficult. It would therefore be ideal if α can be determined from the data itself. The Chi2 algorithm enhanced the ChiMerge algorithm in that the value of α was calculated based on the training data itself. The calculated value of α differed from attribute to attribute, so that it would only continue merging intervals on those attributes which needed it.

The Chi2 algorithm can be sectioned into two different phases:

```
Phase 1:
    Set α = 0.5;
    do while (InConCheck(data) < δ)
    { for each numeric attribute
      {     Sort(attribute, data); /* Sort data on Attribute*/
            Chi-sq-init(att, data);
            do    {Chi-sq-calculation(att, data);}
                  while (Merge(data))
      }
      α0 = α; α = DecreSigLevel(α);
    }
Phase 2:
    Set all sigLvL[i] = α0 for attribute i;
    do until no attribute can be merged
    { for each mergeable attribute i
      {     Sort(attribute, data); /* Sort data on Attribute*/
            Chi-sq-init(att, data);
            do    {Chi-sq-calculation(att, data); }
                  while (Merge(data))
            if (InConCheck(data) < δ)
                  sigLvl[i] = DecreSigLevel(sigLvl[i]);
            else attribute i is not mergeable;
      }
    }
```

• *The authors are with the Department of Mechanical Engineering, National University of Singapore, 10 Kent Ridge Crescent, Singapore 119260. E-mail: {mpetayeh, engp8633}@nus.edu.sg.*

Manuscript received 1 Mar. 2000; revised 14 July 2000; accepted 23 Feb. 2001.
For information on obtaining reprints of this article, please send e-mail to: tkde@computer.org, and reference IEEECS Log Number 111606.

Where:

InConCheck()—Calculation of inconsistency rate; it is calculated as follows:

1. Two instances are considered inconsistent if they match, except for their class label.
2. For all the matching instances (without considering their class labels), the inconsistency count is the number of the instances minus the largest number of the instances of the class labels; for example, there are n matching instances, among them, c_1 instances belong to label_1, c_2 to label_2, and c_3 to label_3 where $\mathrm{c}_1 + \mathrm{c}_2 + \mathrm{c}_3 = \mathrm{n}$. If c_3 is the largest among the three, the inconsistency count is $(\mathrm{n} - \mathrm{c}_3)$.
3. The inconsistency rate is the sum of all the inconsistency counts divided by the total number of the instances.

DecreSigLevel()—decreasing the significance level by one level;

Merge()—returning true or false depending on whether the concerned attribute is merged or not;

Chi-sq-init()—calculation of the A_{ij}, R_i, C_j, N, E_{ij}, k for the calculation of χ^2 value;

Chi-sq-calculation()—calculation of χ^2 value according to (1).

The first phase of this algorithm can be regarded as a generalization of the ChiMerge algorithm. Instead of a predefined significance level α, the Chi2 algorithm provided a wrapping that automatically incremented the threshold (decreasing the significance level α). Consistency checking was utilized as a stopping criterion. These enhancements ensured that the Chi2 algorithm automatically determined a proper threshold while still keeping the fidelity of the original data.

The second phase refined the intervals. If any of the attributes consisting of any of the intervals can be further merged without increasing the inconsistency of the training data above the given limit, then the merging phase was carried out. While the first phase worked on a global significance level α, the second phase used separate local significance levels for each attribute [7]. In the following sections, original Chi2 algorithm referred to the Chi2 algorithm.

2 Problem Analysis and Modification

In the original Chi2 algorithm, the stopping criterion was defined as the point at which the inconsistency rate exceeded the predefined rate $\delta(\mathrm{InConCheck(data)} > \delta)$. In [6], different δ values were assigned to different data sets for feature selection. Some attributes were removed according to this larger δ value. However, these results were obtained on the basis of decreasing the fidelity of the original data set and this method ignored the conflict cases existing in the data set. In addition, before the attributes were discretized, a significant value for this threshold was not available for different data sets. The δ value was always given after some tests were performed on the training data set. This was unreasonable for the case of an unknown data set. Therefore, a level of consistency [3], coined from Rough Sets Theory [8], was introduced to replace the inconsistency checking. Its definition was generalized as follows:

Let U denote the set of all instances of the data set and A is the set of attributes (i.e., condition attributes and one class label). For every subset of attributes $B \subset A$, an *indiscernibility relation IND(B)* is defined in the following way: Two instances, x_i and x_j, are indiscernible by the set of attributes B, if $b(x_i) = b(x_j)$ for every $b \subset B$. The equivalence class of $IND(B)$ is called *elementary set* in B because it represents the smallest discernible groups of instances. For any instance x_i of U, the equivalence class of x_i in relation $IND(B)$ is represented as $[x_i]_{INB(B)}$.

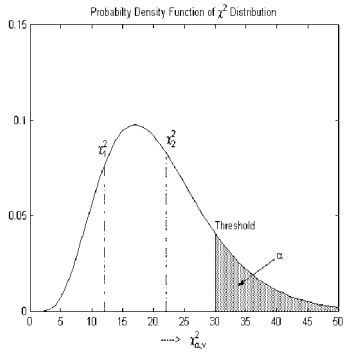

Fig. 1. Probability density function χ^2 distribution (Degree of freedom $\mathrm{v} = 10$).

Let X be any subset of $U(X \subset U)$. The lower approximation of X in B is denoted by $\underline{B}X$ and it is defined as the union of all these elementary sets contained in X. More formally,

$$\underline{B}X = \cup\{x_i \in U | [x_i]_{IND(B)} \subseteq X\}. \tag{2}$$

The lower approximation $\underline{B}X$ is the set of all instances of U, which can be classified with certainty as instances of X, with respect to the values of attributes from B.

The level of consistency, denoted L_c, is defined as follows:

$$L_c = \frac{\sum |\underline{B}X_i|}{|U|}, \tag{3}$$

where $X_i(X_i \in U)$ is a classification of U and $X_i \cap X_j = \emptyset$ and $\cup X_i = U(i = 1, 2, \ldots, n)$.

From (3), it can be seen that L_c represents the percentage of instances which can be correctly classified into class X_i with respect to B. It is intended to capture the degree of completeness of our knowledge about set U. In the Rough Sets Theory, the level of consistency is known as the degree of dependency of d (class label) on A. For a consistent data set with respect to the class label d, $L_c = 1$.

In the modified Chi2 algorithm, inconsistency checking $(\mathrm{InConCheck}() < \delta)$ of the original Chi2 algorithm was replaced by maintaining the level of consistency L_c after each step of discretization $(\mathrm{L_{c\text{-}discretized}} \leq \mathrm{L_{c\text{-}original}})$. By using this inconsistency rate as the stopping criterion, it guaranteed that the fidelity of the training data could be maintained to be the same after discretization. In addition, it made the discretization process completely automatic.

Another problem in the original Chi2 algorithm was that merging criterion (function Merge()) was not very accurate, which would lead to overmerging. This merging criterion was defined as selecting the pair with the lowest χ^2 value to merge into one interval. However, it did not consider the factor of degree of

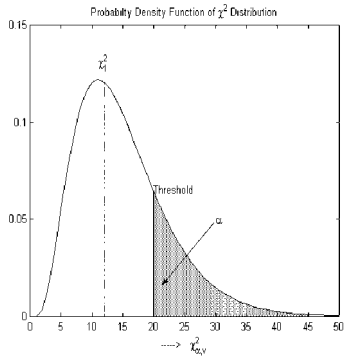

Fig. 2. Probability density function of χ^2 distribution (Degree of freedom $\mathrm{v} = 7$).

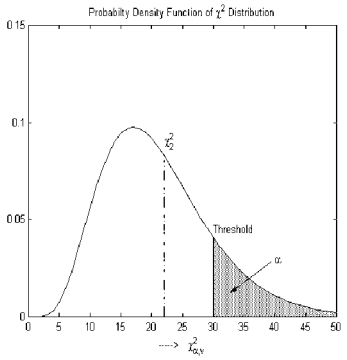

Fig. 3. Probability density function of χ^2 distribution (Degree of freedom $v = 10$).

freedom, which must first be specified as mentioned in Section 1. As illustrated in Fig. 1, it only used the fixed degree of freedom (the class number-1) and a specified significance value α to calculate the threshold—χ^2 value. After all of the χ^2 values from an adjacent interval were computed, the two intervals with the minimal value were chosen to merge, despite the fact that the degrees of freedom were not the same for different adjacent intervals. From the view of statistics, this was inaccurate [9]. The interpretation of the above conclusion was depicted in Fig. 2 and Fig. 3. For the convenience of clarification, the minimal difference between the calculated χ^2 values were enlarged. The probability density function of χ^2 distribution was presented in Fig. 2 and Fig. 3 with different degrees of freedom. The two vertical lines represented the χ^2 values calculated from the adjacent intervals and threshold. The shaded area represented the significance value α. The threshold referred to the point at which there was $(1-\alpha)$ percent probability that the computed χ^2 value would be less than this value among the cases which the class and attribute were independent. In Fig. 2, the χ^2 value was 13, while the corresponding threshold value was 20. In Fig. 3, the χ^2 value was 22, while the corresponding threshold value was 30. If the original merging criterion was applied, the adjacent intervals with χ^2 value equaled to 13 were merged and compared with threshold 30 without considering the effect of the degree of freedom. In this case, if the difference in degrees of freedom was considered, from Fig. 2 and Fig. 3, the second one (from 22 to threshold 30, the distance is 8) was much further from the threshold than the first one (from 13 to threshold 20, the distance is 7). It meant that independence of these two adjacent intervals was higher than that of the first case. Therefore, these two intervals should be merged first. From the above analysis, it was shown that it was more reasonable and more accurate to take into consideration the degree of freedom.

Furthermore, the original Chi2 algorithm only considered the maximal degree of freedom, which meant that the merging procedure would continue until all the χ^2 values exceeded the threshold. This could result in some attributes being overmerged, while others were not be discretized fully. Consequently, this merging procedure would bring about more inconsistency after discretization.

Considering both the above modifications, the algorithm with modified merging criterion and new stopping criterion was termed the modified Chi2 algorithm.

3 Experimental Results

For the convenience of comparing the modified Chi2 algorithm with the original Chi2 algorithm, the same 11 data sets were chosen to be discretized. The data contained real-life information from the medical and scientific fields which had been used previously in testing pattern recognition and machine learning methods. Table 1 gives a summary of data sets used in this experiment.

The data set *hsv-r* [10] represented raw data on treatment of duodenal ulcer by HSV. The remaining 10 data sets were taken from the University of California at Irvine repository of machine learning databases [11], including four large data sets

TABLE 1
Data Sets Information

Data Set	Name	Examples	Continuous attributes	Discrete attributes	classes
Medical data from the University of Wisconsin Hospitals	Breast Cancer	699	9	0	2
A liver disorder data set gathered by BUPA Medical Research Ltd, England	Bupa	345	6	0	2
The glass types data created by Home Office Forensic Science Service, Canada	Glass	214	9	0	6
Medical data from the Cleveland Clinic Foundation	Heart disease	297	5	8	5
Raw data on treatment of duodenal ulcer by HSV[10]	Hsv – r	122	9	2	4
The famous iris classification data by R. A. Fisher	Iris	150	4	0	3
Wine recognition data fromInstitute of Pharmaceutical and Food Analysis and Technologies, Italy	Wine	178	13	0	3
Blocks Classification from University of Bari, Italy	Page-blocks	5473	10	0	5
Optical Recognition of Handwritten Digits data from Bogazici University, Turkey	Optdigit	5620	56	8	10
The training part of Pen-Based Recognition of Handwritten Digits data from Bogazici University, Turkey	Pendigit	7494	12	0	10
Data set generated from Landsat Multi-Spectral Scanner image data	Satellite	6435	36	0	6

TABLE 2
The Predictive Accuracy (Percent) Using C4.5 With Different Discretization Algorithm

Data Set	C4.5			
	Continuous	Original Chi2 algorithm (δ=0)	Modified Chi2 algorithm	Ent – MDLPC
Breast Cancer	95.00± 3.00	96.77± 3.81	97.21± 5.54	95.88± 4.43
Bupa	68.41± 4.81	50.00± 9.10	50.29±11.13	63.53± 8.23
Glass	68.69±13.95	36.19±23.14	32.86±18.21	71.43± 8.69
Heart disease	53.87± 7.67	55.17±10.41	60.69± 9.06	54.48±11.92
Hsv – r	59.02±10.47	54.17±22.29	58.33±11.91	60.67±13.59
Iris	95.33± 6.32	94.00± 7.98	94.67± 7.57	94.00± 8.58
Wine	92.70± 7.42	88.82±14.79	93.21±12.02	87.65±11.25
Page-blocks	96.84± 1.13	93.97± 2.39 (δ=0.0022)	94.61± 3.25	95.14± 2.51
Optdigit	81.23± 0.97	72.72± 2.98	76.05± 3.49	79.63± 2.20
Pendigit	90.01± 0.53	78.40± 1.63	85.41± 1.11	88.77± 1.63
Satellite	86.22± 1.26	80.86± 4.38	81.26± 3.99	81.62± 3.95
Average	80.67	72.82	74.96	79.35

(examples $> 1,000$). The reason for choosing these four large data sets is to test if the modified Chi2 algorithm produces large discrete groups because χ^2 values grow proportionally with the number of instances. All of the above 11 data sets have a level of consistency equal to 1 except that *page-blocks* data set is 0.9916.

In the following experiments, C4.5 (Release 8) [12] was chosen to be the benchmark for evaluating and comparing the performance of the modified Chi2 algorithm and original Chi2 algorithm. The reasons for our choice were that C4.5 [13] worked well for many decision-making problems and it was a well-known method, thus requiring no further descriptions. C4.5 was selected as the benchmark to evaluate the original Chi2 algorithm in [6] and has shown that the Chi2 algorithm was an effective discretization method. To compare the efficacy of these two methods, the predictive accuracy of C4.5 on the undiscretized data sets was presented, which was denoted by *Continuous* in Table 2 and Table 3. C4.5 was run using its default setting and the predictive accuracy was chosen as the evaluation benchmark.

The ten-fold cross-validation test method [14] was applied to all the data sets. The data set was divided into 10 parts of which nine parts were used as training sets and the remaining one part as the testing set. The experiments were repeated 10 times. The final predictive accuracy was taken as the average of the 10 predictive accuracy values. Because the two modifications on the original Chi2 algorithm were made to maintain the fidelity of the original data set, the modified Chi2 algorithm was compared with the original Chi2 algorithm with threshold δ value equaled to 0 in the

TABLE 3
The Tree Size (Before/After Pruning) Comparison of Four Methods

Data Set	C4.5			
	Continuous	Original Chi2 algorithm (δ=0)	Modified Chi2 algorithm	Ent – MDLPC
Breast-cancer	128.0±13.37/ 37.0±17.63	52.3±7.94/ 19.5±2.99	53.7±7.92/ 22.4±6.20	48.2±7.77/ 20.8±3.39
Bupa	57.0±12.33/ 43.8±12.51	292.0±80.44/ 43.5±31.49	298.7±45.93/ 48.0±34.15	52.4±11.39/ 45.8±10.16
Glass	48.6±5.72/ 44.6±5.72	148.8±42.76/ 65.4±17.13	161.7±33.09/ 73.3±32.62	40.3±8.58/ 33.2±7.56
Heart disease	124.2±5.79/ 83.1±15.25	185.5±22.06/ 71.2±14.83	162.4±24.78/ 74.3±11.10	124.7±11.55/ 81.3±16.24
Hsv-r	41.8±3.58/ 30.8±5.75	62.4±14.52/ 15.9±9.04	58.0±11.73/ 3.1±4.48	38.6±6.74/ 31.0±8.18
Iris	8.8±1.14/ 8.8±1.14	21.7±3.86/ 5.2±2.70	24.2±6.48/ 6.6±3.24	5.9±3.07/ 4.0±0.00
Wine	9.6±1.35/ 9.2±0.63	26.2±5.18/ 19.7±4.06	29.1±9.36/ 19.2±3.85	19.7±3.43/ 16.5±2.64
Page-blocks	119.4±12.50/ 80.4±12.00	3056.9±363.32/ 417.1±115.85 (δ=0.0022)	3876.9±912.32/ 558.9±185.69	407.2±45.75/ 132.0±15.30
Optdigit	438.8±18.12/ 410.4±16.39	4878.7±91.79/ 3028.7±86.49	3123.9±89.85/ 2020.0±100.19	1942.3±62.51/ 1323.7±48.34
Pendigit	311.2±10.64/ 286.2±11.63	9716.5±520.62/ 4605.2±261.33	4643.5±137.29/ 2448.7±132.73	2595.9±79.62/ 1480.5±56.64
Satellite	656.4±18.21/ 48.8±17.87	5322.3±225.16/ 1763.7±111.71	4501.3±240.11/ 1546.5±85.30	3534.7±132.57/ 1416.9±81.65

experiments, except for the *page-blocks* data set with an inconsistency rate equal to 0.0022. In addition, since the modified Chi2 algorithm was a parameter-free discretization method, its efficacy in discretization was compared to another parameter-setting-free method—the Ent-MDLPC algorithm [4], which has been accepted as one of the best supervised discretization methods [2], [15]. First, all 11 data sets were discretized using the original Chi2 algorithm, the modified Chi2 algorithm, and the Ent-MDLPC algorithm, after which the discretized data sets were sent into C4.5. The predictive accuracy and its standard deviation of these four methods were presented in Table 2. The tree size using C4.5 with different discretization methods was presented in Table 3.

To analyze the results obtained in Table 2, the Wilcoxon matched-pairs sign-rank test [9] was applied. The purpose of this nonparametric test was to determine if significant differences existed between two populations. Paired observations from the two populations were the basis of the test and magnitudes of differences were taken into consideration. This was a straightforward procedure to either accept or reject the null hypothesis, which was commonly taken to be identical population distributions.

The modified Chi2 algorithm outperforms the original Chi2 algorithm at 1 percent significance level for a one-tailed test. However, it shows no significant performance difference from the C4.5 and Ent-MDLPC algorithm, i.e., the null hypothesis could not be rejected even at the 5 percent significance level for a one-tailed test. The C4.5 outperforms the original Chi2 algorithm with 0.5 percent significance level for a one-tailed test. However, it shows no significant difference in performance from the Ent-MDLPC algorithm. The Ent-MDLPC algorithm outperforms the original Chi2 algorithm with 5 percent significance level for a one-tailed test.

From Table 3, it can be seen that the tree size after applications of the three discretization algorithms is mostly reduced when compared with the C4.5. This indicates that the three methods effectively discretize the numerical attributes and remove the irrelevant and redundant attributes for the follow-up C4.5 processing. However, for the four large data sets, although there is no significant difference in the predictive accuracy, the tree size is significantly greater than those of the C4.5 for the remaining three algorithms. This means these three methods generate too many discrete intervals for a larger data set and they are more suitable for the medium size data set than the C4.5.

4 Conclusions

In this paper, a modified Chi2 algorithm is proposed as a completely automated discretization method. It replaces the inconsistency check in the original Chi2 algorithm using a Level of Consistency, coined from the Rough Sets Theory, which maintains the fidelity of the training set after discretization. In contrast to the original Chi2 algorithm which ignores the effect of the degree of freedom, this modified algorithm takes into consideration the effect of the degree of freedom which consequently results in greater accuracy. Although this algorithm adds one step (i.e., to select merging intervals), it does not increase the computational complexity as compared to the original Chi2 algorithm. With these modifications, the ChiMerge has become a completely automated discretization method and its predictive accuracy is better than the original Chi2 algorithm. However, compared with the C4.5 and Ent-MDLPC algorithm, the modified Chi2 algorithm has no significant performance difference in predictive accuracy. Further, for a large data set, it generates a larger tree compared with the C4.5. This is not favorable to its application and this problem will be studied in the following research. Thus, a simple algorithm that can automatically select a proper critical value for the χ^2 test and determine the intervals of a numeric attribute according to the characteristics of the data has been realized.

References

[1] R. Kohavi, "Bottom-Up Induction of Oblivious Read-Once Decision Graphs: Strengths and Limitation," *Proc. 12th Nat'l Conf. Artificial Intelligence,* pp. 613-618, 1994.

[2] J. Dougherty, R. Kohavi, and M. Sahami, "Supervised and Unsupervised Discretization of Continuous Features," *Machine Learning: Proc. 12th Int'l Conf.,* pp.194-202, 1995.

[3] M. Chmielewski and J. Grzrmala-Busse, "Global Discretization of Continuous Attributes as Preprocessing for Machine Learning," *Int'l J. Approximate Reasoning,* vol. 15, no. 4, pp. 319-331, Nov. 1996.

[4] U.M. Fayyad and K.B. Irani, "Multi-Interval Discretization of Continuous-Valued Attributes for Classification Learning," *Proc. 13th Int'l Joint Conf. Artificial Intelligence,* pp. 1022-1027, 1993.

[5] R. Kerber, "ChiMerge: Discretization of Numeric Attributes," *Proc. Ninth Int'l Conf. Artificial Intelligence,* pp.123-128, 1992.

[6] H. Liu and R. Setiono, "Feature Selection via Discretization of Numeric Attributes," *IEEE Trans. Knowledge and Data Eng.,* vol. 9, no. 4, pp. 642-645, July/Aug. 1997.

[7] K.M. Risvik, "Discretization of Numerical Attributes, Preprocessing for Machine Learning," project report, Knowledge Systems Group, Dept. of Computer Systems and Telematics, The Norwegian Inst. of Technology, Univ. of Trondheim, 1997.

[8] Z. Pawlak, "Rough Sets," *Int'l J. Computer and Information Sciences,* vol. 11, no. 5, pp.341-356, 1982.

[9] D.C. Montgomery and G.C. Runger, *Applied Statistics and Probability for Engineers,* second ed. Jones Wiley & Sons, Inc., 1999.

[10] K. Slowinski, "Rough Classification of HSV Patients," *Intelligent Decision Support—Handbook of Applications and Advances of the Rough Sets Theory,* chapter 6, pp. 77-94, Kluwer Academic, 1992.

[11] C.J. Merz and P.M. Murphy, *UCI Repository of Machine Learning Databases,* http://www.ics.uci.edu/~mlearn/MLRepository.html.

[12] J.M. Quinlan, "Improved Use of Continuous Attributes in C4.5," *J. Artificial Intelligence Research,* vol. 4, pp. 77-90, 1996.

[13] J.M. Quinlan, *C4.5: Programs for Machine Learning.* San Mateo, Calif.: Morgan Kaufmann, 1993.

[14] S.M. Weiss and C.A. Kulikowski, *Computer Systems That Learn: Classification and Prediction Methods from Statistics, Neural Nets, Machine Learning, and Expert Systems.* San Mateo, Calif.: Morgan Kaufmann, 1990.

[15] R. Kohavi and M. Sahami, "Error-Based and Entropy-Based Discretization of Continuous Features," *Proc. Second Int'l Conf. Knowledge Discovery & Data Mining,* pp. 114-119, 1996.