Data Discretization Unification

Ruoming Jin Yuri Breitbart Chibuike Muoh
Department of Computer Science
Kent State University, Kent, OH 44241
{jin,yuri,cmuoh}@cs.kent.edu

Abstract

Data discretization is defined as a process of converting continuous data attribute values into a finite set of intervals with minimal loss of information. In this paper, we prove that discretization methods based on informational theoretical complexity and the methods based on statistical measures of data dependency are asymptotically equivalent. Furthermore, we define a notion of generalized entropy and prove that discretization methods based on MDLP, Gini Index, AIC, BIC, and Pearson's X^2 and G^2 statistics are all derivable from the generalized entropy function. We design a dynamic programming algorithm that guarantees the best discretization based on the generalized entropy notion. Furthermore, we conducted an extensive performance evaluation of our method for several publicly available data sets. Our results show that our method delivers on the average 31% less classification errors than many previously known discretization methods.

1 Introduction

Many real-world data mining tasks involve continuous attributes. However, many of the existing data mining systems cannot handle such attributes. Furthermore, even if a data mining task can handle a continuous attribute, its performance can be significantly improved by replacing a continuous attribute with its discretized values. Data discretization is defined as a process of converting continuous data attribute values into a finite set of intervals and associating with each interval some specific data value. There are no restrictions on discrete values associated with a given data interval except that these values must induce some ordering on the discretized attribute domain. Discretization significantly improves the quality of discovered knowledge [8, 30] and also reduces the running time of various data mining tasks such as association rule discovery, classification, and prediction. Catlett in [8] reported ten fold performance improvement for domains with a large number of continuous attributes with little or no loss of accuracy. However, any discretization process generally leads to a loss of information. Thus, the goal of the *good* discretization algorithm is to minimize such information loss.

Discretization of continuous attributes has been extensively studied [5, 8, 9, 10, 13, 15, 24, 25]. There are a wide variety of discretization methods starting with naive (often referred to as *unsupervised*) methods such as equal-width and equal-frequency [26], to much more sophisticated (often referred to as *supervised*) methods such as *Entropy* [15] and Pearson's X^2 or Wilks' G^2 statistics based discretization algorithms [18, 5]. Unsupervised discretization methods are not provided with class label information whereas supervised discretization methods are supplied with a class label for each data item value. Liu *et. al.* [26] introduce a nice categorization of a large number of existing discretization methods.

In spite of the wealth of literature on discretization methods, there are very few attempts to *analytically* compare them. Typically, researchers compare the performance of different algorithms by providing experimental results of running these algorithms on publicly available data sets. In [13], Dougherty *et al.* compare discretization results obtained by unsupervised discretization versus a supervised method proposed by [19] and the entropy based method proposed by [15]. They conclude that supervised methods are better than unsupervised discretization method in that they generate fewer classification errors. In [25], Kohavi and Sahami report that the number of classification errors generated by the discretization method of [15] is comparatively smaller than the number of errors generated by the discretization algorithm of [3]. They conclude that entropy based discretization methods are usually better than other supervised discretization algorithms.

Recently, many researchers have concentrated on the generation of new discretization algorithms [38, 24, 5, 6]. The goal of the CAIM algorithm [24] is to find the minimum number of intervals that minimize the loss between class-attribute interdependency. Boulle [5] has proposed a new discretization method called $Khiops$, which uses Pearson's X^2 statistic to merge consecutive intervals in order to improve the global dependence measure. *MODL* is another latest discretization method proposed by Boulle [5]. This method builds an optimal criteria based on a Bayesian model. A dynamic programming approach and a greedy heuristic approach are developed to find the optimal criteria. Finally, Yang and Webb have studied discretization for naive-Bayes classifiers [38]. They have proposed a couple of methods, such as *proportional k-interval discretization* and *equal size discretization*, to manage the discretization *bias* and *variance*. All these algorithms have shown certain ad-

Intervals	Class 1	Class 2	$\cdots$	Class J	Row Sum
S_1	c_{11}	c_{12}	$\cdots$	c_{1J}	N_1
S_2	c_{21}	c_{22}	$\cdots$	c_{2J}	N_2
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
S_I	c_{I1}	c_{I2}	$\cdots$	c_{IJ}	N_I
Column Sum	M_1	M_2	$\cdots$	M_J	N (Total)

Table 1: **Notations for Contingency Table** C

vantages, such as improving classification accuracy and/or complexity.

Several fundamental questions of discretization, however, remain to be answered. How these different methods are related to each other and how different or how similar are they? Is there an objective function which can measure the goodness of different approaches? If so, how would this function look like? In this paper we provide a set of positive results toward answering these questions.

1.1 Problem Statement

For the purpose of discretization, the entire dataset is projected onto the targeted continuous attribute. The result of such a projection is a two dimensional *contingency table*, C with I rows and J columns. Each row corresponds to either a point in the continuous domain, or an initial data interval. We treat each row as an atomic unit which cannot be further subdivided. Each column corresponds to a different class and we assume that the dataset has a total of J classes. A cell c_{ij} represents the number of points with j-th class label falling in the i-th point (or interval) in the targeted continuous domain. Table 1 lists the basic notations for the contingency table C.

In the most straightforward way, each continuous point (or initial data interval) corresponds to a row of a contingency table. Generally, in the initially given set of intervals each interval contains points from different classes and thus, c_{ij} may be more than zero for several columns in the same row.

The goal of a discretization method is to find another contingency table, C', with $I' << I$, where each row in the new table C' is the combination of several consecutive rows in the original C table, and each row in the original table is covered by exactly one row in the new table.

The quality of the discretization function is measured by a $goodness$ function, which depends on two parameters. The first parameter (termed $cost(data)$) reflects the number of classification errors generated by the discretization function, whereas the second one (termed $penalty(model)$) is the complexity of the discretization which reflects the number of discretization intervals generated by the discretization function. Clearly, the more discretization intervals created, the fewer the number of classification errors, and thus the cost of the data is lower. That is, if one is interested only in minimizing the number of classification errors, the $best$ discretization function would generate I intervals – the number of data points in the initial contingency table. Conversely, if one is only interested in minimizing the number of intervals (and therefore reducing the penalty of the model), then the $best$ discretization function would generate a single interval by merging all data points into one interval. Thus, finding the $best$ discretization is to find the best trade-off between the $cost(data)$ and the $penalty(model)$.

1.2 Our Contribution

Our results can be summarized as follows:

1. We demonstrate a somewhat unexpected connection between discretization methods based on information theoretical complexity, on one hand, and the methods which are based on statistical measures of the data dependency of the contingency table, such as Pearson's X^2 or G^2 statistics on the other hand. Namely, we prove that each goodness function defined in [15, 16, 5, 23, 27, 4] is a combination of G^2 defined by Wilks' statistic [1] and degrees of freedom of the contingency table multiplied by a function that is bounded by $O(logN)$, where N is the number of data samples in the contingency table.

2. We define a notion of generalized entropy and introduce a notion of generalized goodness function. We prove that goodness functions for discretization methods based on MDLP, Gini Index, AIC, BIC, Pearson's X^2, and G^2 statistic are all derivable from the generalized goodness function.

3. We design a dynamic programming algorithm that guarantees the best discretization based on a generalized goodness function.

4. We conduct an extensive performance evaluation of our discretization method and demonstrate that our method on the average outperforms by 31% the prior discretization methods for all publicly available data sets we tried.

2 Goodness Functions

In this section we introduce a list of goodness functions which are used to evaluate different discretization for numerical attributes. These goodness functions intend to measure three different qualities of a contingency table: the information complexity, the fitness of statistical models, and the confidence level for statistical independence tests .

Information Theoretical Approach and MDLP: In the information theoretical approach, we treat discretization of a single continuous attribute as a 1-*dimension classification* problem. The Minimal Description Length Principle (MDLP) is a commonly used approach for choosing the best classification model [31, 18]. It considers two factors: how good the discretization fit the data, and the penalty for the discretization which is based on the complexity of discretization. Formally, MDLP associates a cost with each discretization, which has the following form:

$$cost(model) = cost(data|model) + penalty(model)$$

where both terms correspond to these two factors, respectively. Intuitively, when a classification error increases, the penalty decreases and vice versa.

To facilitate our discussion, let $H(S_i)$ be the *entropy* [12] of interval S_i and $H(S_1, \cdots, S_{I'})$ be the total entropy for the I' intervals:

$$H(S_i) = -\sum_{j=1}^{J} \frac{c_{ij}}{N_i} log_2 \frac{c_{ij}}{N_i}$$

$$H(S_1, \cdots, S_{I'}) = \sum_{i=1}^{I'} \frac{N_i}{N} H(S_i)$$

Given this, the cost of discretization based on MDLP can be expressed as follows:

$$Cost_{MDLP} = N \times H(S_1, \cdots, S_{I'}) + (I'-1) log_2 \frac{N}{I'-1} + I'(J-1) log_2 J \quad (1)$$

The first term corresponds to $cost(data|model)$, which are the cost to transfer the labels of each continuous point, and the rest corresponds to $penalty(model)$, which describes the coding book of labels and necessary delimiters. The detailed derivation is given in the technical report [22]. Note that similar formulation has been used in Fayyad and Irani's *Entropy* approach for choosing the best cut point for discretization [15].

To facilitate the comparison with other cost functions, we formally define a goodness function of a MDLP based discretization method applied to contingency table C to be the difference between the cost of C^0, which is the resulting table after merging all the rows of C into a single row, and the cost of C. We will also use *natural log* instead of the log_2 function. Formally, we denote the goodness function based on MDLP as GF_{MDLP}.

$$GF_{MDLP}(C) = Cost_{MDLP}(C^0) - Cost_{MDLP}(C)$$
$$= N \times H(S_1 \cup \cdots \cup S_{I'}) - N \times H(S_1, \cdots, S_{I'}) - ((I'-1) log \frac{N}{I'-1} + (I'-1)(J-1) log J) \quad (2)$$

where, $S_1 \cup \cdots \cup S_{I'}$ is the merged interval of $S_1, \cdots, S_{I'}$. Note that for a discretization problem, any discretization method shares the same C^0. Thus, the least cost of transferring a contingency table corresponds to the maximum of the goodness function.

Statistical Model Selection (AIC and BIC): A different way to look at a contingency table is to assume that all data points are generated from certain distributions (models) with unknown parameters. Given a distribution, the maximal likelihood principle (MLP) can help us to find the best parameters to fit the data [16]. However, to provide a better data fitting, more expensive models (including more parameters) are needed. Statistical model selection tries to find the right balance between the complexity of a model corresponding to the number of parameters, and the fitness of the data to the selected model, which corresponds to the likelihood of the data being generated by the given model.

For choosing the best discretization model, two criteria are often used: the *Akaike information criterion* (AIC) and *Bayesian information criterion* (BIC) [16].

$$Cost_{AIC} = 2S_L + 2(I' \times (J-1)) \quad (3)$$

$$Cost_{BIC} = 2S_L + (I' \times (J-1)) logN \quad (4)$$

where, $S_L = -\sum_{i=1}^{I'} \sum_{j=1}^{J} c_{ij} log \frac{c_{ij}}{N_i}$ is the log-likelihood for the corresponding discretization. Generally speaking, the first term corresponds to the fitness of the data given the discretization model, and the second term corresponds to the complexity of the model. Clearly, since BIC takes into account the size of the training set N, the penalty of the model is higher than the one in the AIC by a factor of $logN/2$.

For the same reason as MDLP, we denote the goodness function of a given contingency table based on AIC and BIC as follows:

$$GF_{AIC}(C) = Cost_{AIC}(C^0) - Cost_{AIC}(C) \quad (5)$$

$$GF_{BIC}(C) = Cost_{BIC}(C^0) - Cost_{BIC}(C) \quad (6)$$

Confidence Level from Independence Tests: Another way to treat discretization is to merge intervals so that the rows (intervals) and columns (classes) of the entire contingency table become more statistically *dependent*. In other words, the goodness function of a contingency table measures its statistical quality in terms of independence tests. Two statistics, *Pearson's* X^2 and *Wilks'* G^2, are commonly used for such purpose [1]:

$$X^2 = \sum \sum \frac{(c_{ij} - \hat{m}_{ij})^2}{\hat{m}_{ij}} \quad (7)$$

$$G^2 = 2 \sum_{i=1}^{I'} \sum_{j=1}^{j} c_{ij} log \frac{c_{ij}}{N_i M_j / N} \quad (8)$$

where, $\hat{m}_{ij} = N(N_i/N)(M_j/N)$ is the expected frequencies. It is well known that both Pearson's X^2 and Wilks' G^2 statistics have an asymptotic χ^2 distribution with degrees of freedom $df = (I'-1)(J-1)$, where I' is the total number of rows [1].

Consider a null hypothesis H_0 (the rows and columns are statistically independent) against an alternative hypothesis H_a. We can obtain the confidence level of the statistical test to reject the independence hypothesis (H_0). Given this, we use the confidence level as our goodness function to compare different discretization methods that use X^2 and G^2 statistics.

Our goodness functions are formally defined as

$$GF_{X^2}(C) = F_{\chi^2_{df}}(X^2) \quad (9)$$

$$GF_{G^2}(C) = F_{\chi^2_{df}}(G^2) \quad (10)$$

where, $F_{\chi^2_{df}}$ is the cumulative χ^2 distribution function. We note that $1 - F_{\chi^2_{df}}(\cdot)$ is essentially the P-value of the aforementioned statistical independence test [7]. The lower the P-value (or equivalently, the higher the goodness), with more

confidence we can reject the independence hypothesis (H_0). Finally, we note that the confidence interval together with X^2 has been used in Khiops [5], and G^2 has been used in [37] and is referred to as class-attributes interdependency information.

3 Equivalence of Goodness Functions

In this section, we analytically compare different discretization goodness functions introduced in Section 2. In particular, we find some rather surprising connection between these seemingly quite different approaches: the information theoretical complexity , the statistical fitness, and the statistical independence tests. We basically prove that all these functions can be expressed in a uniform format as follows:

$$GF \ = \ G^2 \ - \ df \times f(G^2, N, I, J) \tag{11}$$

where, df is a degree of freedom of the contingency table, N is the number of data points, I is the number of data rows in the contingency table, J is the number of class labels and f is bounded by $O(logN)$. The first term G^2 corresponds to the cost of the data given a discretization model ($cost(data|model)$), and the second corresponds to the penalty or the complexity of the model ($penalty(model)$).

To derive this expression, we first derive an expression for the cost of the data for different goodness functions discussed in section 2 (Subsection 3.1). This is achieved by expressing G^2 statistics through information entropy (Theorem 2). Then, using a Wallace's result [35, 36] on approximating χ^2 distribution with a normal distribution, we transform the goodness function based on statistical independence tests into the format of Formula 11. Further, a detailed analysis of function f reveals a deeper relationship shared by these different goodness functions (Subsection 3.3).

3.1 Unifying the Cost of Data to G^2

In the following, we establish the relationship among *entropy*, *log-likelihood* and G^2. This is the first step for an analytical comparison of goodness functions based on the information theoretical, the statistical model selection, and the statistical independence test approaches. Due to the lack of space, the proofs of Theorem 1 and 2 can be found in the technical report [22].

First, it is easy to see that for a given contingency table, the cost of the data transfer ($cost(data|model)$, a key term in the information theoretical approach) is equivalent to the log likelihood S_L (used in the statistical model selection approach) as the following theorem asserts.

Theorem 1 *For a given contingency table $C_{I' \times J}$, the cost of data transfer ($cost_1(data|model)$) is equal to the log likelihood S_L, i.e.*

$$N \times H(S_1, \cdots, S_{I'}) = S_L = -\sum_{i=1}^{I'} \sum_{j=1}^{J} c_{ij} log \frac{c_{ij}}{N_i}$$

The next theorem establishes a relationship between entropy criteria and the likelihood independence testing statistics G^2. This is the key to discover the connection between the information theoretical and the statistical independence test approaches.

Theorem 2 *Let C be a contingency table. Then*

$$G^2/2 = N \times H(S_1 \cup \cdots \cup S_{I'}) - N \times H(S_1, \cdots, S_{I'})$$

Consequently, we rewrite the goodness functions GF_{MDLP}, GF_{AIC} and GF_{BIC} as follows.

$$GF_{MDLP} =$$
$$G^2 - 2(I'-1)log\frac{N}{I'-1} - 2(I'-1)(J'-1)logJ \tag{12}$$
$$GF_{AIC} = G^2 - (I'-1)(J-1) \tag{13}$$
$$GF_{BIC} = G^2 - (I'-1)(J-1)logN/2 \tag{14}$$

For the rest of the paper we use the above formulas for GF_{MDLP}, GF_{AIC} and GF_{BIC}.

It has long been known that they are asymptotically equivalent. The next theorem provides tool to connect the information theoretical approach and the statistical independence test approach based on Pearson's chi-square (X^2) statistic.

Theorem 3 *[1] Let N be the total number of data values in the contingency table T of $I \times J$ dimensions. If the rows (columns) of contingency table are independent, then probability of X^2 - G^2 = 0 converges to one as $N \to \infty$.*

In the following, we mainly focus on the asymptotic properties shared by X^2 and G^2 based cost functions. Thus, our further discussions on G^2 can also be applied to X^2.

Note that Theorem 1 and 2 basically establish the basis for Formula 11 of goodness functions based on the information theoretical approach and statistical model selection approaches. Even though Theorems 2 and 3 relate the information theoretical approach (based on entropy) to the statistical independence test approach (based on G^2 and X^2), it is still unclear how to compare them directly since the goodness function of the former one is based on the total *cost* of transferring the data and the goodness function of the latter one is the *confidence level* for a hypothesis test. Subsection 3.2 presents our approach on tackling this issue.

3.2 Unifying Statistical Independence Tests

In order to compare the quality of different goodness functions, we introduce a notion of *equivalent* goodness functions. Intuitively, the equivalence between goodness functions means that these functions rank different discretization of the same contingency table identically.

Definition 1 *Let C be a contingency table and $GF_1(C)$, $GF_2(C)$ be two different goodness functions. GF_1 and GF_2 are equivalent if and only if for any two contingency tables C_1 and C_2, $GF_1(C_1) \leq GF_1(C_2) \iff GF_2(C_1) \leq GF_2(C_2)$*

Using the equivalence notion, we transform goodness functions to different scales and/or to different formats. In

the sequel, we apply this notion to compare seemingly different goodness functions.

The relationship between the G^2 and the confidence level is rather complicated. It is clearly not a simple one-to-one mapping as the same G^2 may correspond to very different confidence level depending on degrees of freedom of the χ^2 distribution and, vice versa the same confidence level may correspond to very different G^2 values. Interestingly enough, such many-to-many mapping actually holds the key for the aforementioned transformation. Intuitively, we have to transform the confidence interval to a scale of entropy or G^2 parameterized by the degree of freedom for the χ^2 distribution.

Our proposed transformation is as follows.

Definition 2 *Let $u(t)$ be the normal deviate of the chi-square distributed variable t [21]. That is, the following equality holds :*

$$F_{\chi^2_{df}}(t) = \Phi(u(t))$$

where, $F_{\chi^2_{df}}$ is the χ^2 distribution function with df degrees of freedom, and Φ is the normal distribution function. For a given contingency table C, which has the log likelihood ratio G^2, we define

$$GF'_{G^2} = u(G^2) \tag{15}$$

as a new goodness function for C.

The next theorem establishes the equivalence between a goodness functions GF_{G^2} and GF'_{G^2}.

Theorem 4 *The goodness function $GF'_{G^2} = u(G^2)$ is equivalent to the goodness function $GF_{G^2} = F_{\chi^2_{df}}(G^2)$.*

Proof: Assuming we have two contingency tables C_1 and C_2 with degree of freedom df_1 and df_2, respectively. Their respective G^2 statistics are denoted as G_1^2 and G_2^2. Clearly, we have

$$\begin{aligned} F_{\chi^2_{df_1}}(G_1^2) \leq F_{\chi^2_{df_2}}(G_2^2) &\Longleftrightarrow \\ \Phi(u(G_1^2)) \leq \Phi(u(G_2^2)) &\Longleftrightarrow \\ u(G_1^2) \leq u(G_2^2) & \end{aligned}$$

This basically establishes the equivalence of these two goodness functions. □

The newly introduced goodness function GF'_{G^2} is rather complicated and it is hard to find for it a closed form expression. In the following, we use a theorem from Wallace [35, 36] to derive an asymptotically accurate closed form expression for a simple variant of GF'_{G^2}.

Theorem 5 *[35, 36] For all $t > df$, all $df > .37$, and with $w(t) = [t - df - df\ log(t/df)]^{\frac{1}{2}}$,*

$$0 < w(t) \leq u(t) \leq w(t) + .60df^{-\frac{1}{2}}$$

Note that if $u(G^2) \geq 0$, then, $u^2(G^2)$ is *equivalent* to $u(G^2)$. Here, we limit our attention only to the case when $G^2 > df$, which is the condition for Theorem 5. This condition implies that $u(G^2) \geq 0$. [1] We show that under some conditions, $u^2(G^2)$ can be approximated as $w^2(G^2)$.

$$w^2(G^2) \leq u^2(G^2) \leq w^2(G^2) + \frac{0.36}{df} + 1.2\frac{w(G^2)}{\sqrt{df}}$$

$$1 \leq \frac{u^2(G^2)}{w^2(G^2)} \leq 1 + \frac{0.36}{w^2(G^2) \times df} + \frac{1.2}{w(G^2)\sqrt{df}}$$

$$If\ df \to \infty\ and\ w(t) >> 0,\ then$$

$$\frac{0.36}{w^2(G^2) \times df} \to 0\ and\ \frac{1.2}{w(G^2)\sqrt{df}} \to 0$$

$$Therefore,\ \frac{u^2(G^2)}{w^2(G^2)} \to 1$$

Thus, we can have the following goodness function:

$$GF''_{G^2} = u^2(G^2) = G^2 - df(1 + log(\frac{G^2}{df})) \tag{16}$$

Similarly, function GF''_{χ^2} is obtained from GF''_{G^2} by replacing in the GF''_{G^2} expression G^2 with X^2. Formulas 12, 13, 14 and 16 indicate that all goodness functions introduced in section 2 can be (asymptotically) expressed in the same closed form (Formula 11). Specifically, all of them can be decomposed into two parts. The first part contains G^2, which corresponds to the cost of transferring the data using information theoretical view. The second part is a linear function of degrees of freedom, and can be treated as the penalty of the model using the same view.

3.3 Penalty Analysis

In this section, we perform a detailed analysis of the relationship between penalty functions of these different goodness functions . Our analysis reveals a deeper similarity shared by these functions and at the same time reveals differences between them.

Simply put, the penalties of these goodness functions are essentially bounded by two extremes. On the lower end, which is represented by AIC, the penalty is on the order of degree of freedom, $O(df)$. On the higher end, which is represented by BIC, the penalty is $O(df log N)$.

Penalty of GF''_{G^2} (Formula 16): The penalty of our new goodness function $GF''_{G^2} = u^2(G^2)$ is between $O(df)$ and $O(df log N)$. The lower bound is achieved, provided that G^2 being strictly higher than df ($G^2 > df$). Lemma 1 gives the upper bound (The proof is in the technical report [22]).

Lemma 1 *G^2 is bounded by $2N log J$ ($G^2 \leq 2N log J$).*

In the following, we consider two cases for the penalty $GF''_{G^2} = u^2(G^2)$. Note that these two cases corresponding to the lower bound and upper bound of G^2, respectively.

1. if $G^2 = c_1 \times df$, where $c_1 > 1$, the penalty of this goodness function is $(1 + log c_1)df$, which is $O(df)$.
2. if $G^2 = c_2 \times N log J$, where $c_2 \leq 2$ and $c_2 >> 0$, the penalty of the goodness function is $(1 + log(c_2 N log J/df))$.

The second case is further subdivided into two subcases.

[1] If $u(G^2) < 0$, it becomes very hard to reject the hypothesis that the entire table is statistically independent. Here, we basically focus on the cases where this hypothesis is likely to be reject.

1. If $N/df \approx N/(IJ) = c$, where c is some constant, the penalty is $O(df)$.
2. If $N \to \infty$ and $N/df \approx N/(IJ) \to \infty$, the penalty is

$$df(1+log(c_2 N log J/df) \approx df(1+logN/df) \approx df(logN)$$

Penalty of GF_{MDLP} (Formula 12): The penalty function f derived in the goodness function based on the information theoretical approach can be written as

$$\frac{df}{J-1} log \frac{N}{I-1} + df log J = df(log \frac{N}{I-1}/(J-1) + log J)$$

Here, we again consider two cases:

1. If $N/(I-1) = c$, where c is some constant, we have the penalty of MDLP is $O(df)$.
2. If $N >> I$ and $N \to \infty$, we have the penalty of MDLP is $O(df logN)$.

Note that in the first case, the contingency table is very sparse ($N/(IJ)$ is small). In the second case, the contingency table is very dense ($N/(IJ)$ is very large).

To summarize, the penalty can be represented in a generic form as $df \times f(G^2, N, I, J)$ (Formula 11). This function f is bounded by $O(logN)$. Finally, we observe that different penalty clearly results in different discretization. The higher penalty in the goodness function results in the less number of intervals in the discretization results. For instance, we can state the following theorem.

Theorem 6 *Given an initial contingency table C with $logN \geq 2$ (the condition for the penalty of BIC is higher than the penalty of AIC), let I_{AIC} be the number of intervals of the discretization generated by using GF_{AIC} and I_{BIC} be the number of intervals of the discretization generated by using GF_{BIC}. Then $I_{AIC} \geq I_{BIC}$.*

Note that this is essentially a direct application of the well-known facts from statistical machine learning research: higher penalty will result in more concise models [16].

Finally, we note that several well-known discretization algorithms based on local independence test include ChiMerge [23] and Chi2 [27], etc. Specifically, for consecutive intervals, these algorithms perform a statistical independence test based on Pearson's X^2 or G^2. If they could not reject the independence hypothesis for those intervals, they merge them into one row. A simple analysis in [22] suggests that the local merge condition essentially shares the penalty in the same order of magnitude as GF_{AIC}. Interested readers can refer [22] for detailed discussion.

4 Parametrized Goodness Function

The goodness functions discussed so far are either entropy or χ^2 or G^2 statistics based. In this section we introduce a new goodness function which is based on *gini* index [4]. *Gini* index based goodness function is strikingly different from goodness functions introduced so far. In this section we show that a newly introduced goodness function GF_{gini} along with the goodness functions discussed in section 2 are all can be derived from a generalized notion of entropy [29].

4.1 Gini Based Goodness Function

Let S_i be a row in contingency table C. Gini index of row S_i is defined as follows [4]:

$$Gini(S_i) = \sum_{j=1}^{J} \frac{c_{ij}}{N_i}[1 - \frac{c_{ij}}{N_i}]$$

and $Cost_{Gini}(C) = \sum_{i=1}^{I'} N_i \times Gini(S_i)$

The penalty of the model based on gini index can be approximated as $2I' - 1$ (see detailed derivation in the technical report [22]). The basic idea is to apply a generalized MDLP principle in such a way so that the cost of transferring the data ($cost(data|model)$) and the cost of transferring the coding book as well as necessary delimiters ($penalty(model)$) are treated as the *complexity* measure. Therefore, the gini index can be utilized to provide such a measure. Thus, the goodness function based on gini index is as follows:

$$GF_{gini}(C) = -\sum_{i=1}^{I'} \sum_{j=1}^{J} \frac{c_{ij}^2}{N_i} + \sum_{j=1}^{J} \frac{M_j^2}{N} + 2(I'-1) \quad (17)$$

4.2 Generalized Entropy

In this subsection, we introduce a notion of *generalized* entropy, which is used to uniformly represent a variety of *complexity* measures, including both information entropy and gini index by assigning different values to the parameters of the generalized entropy expression. Thus, it serves as the basis to derive the parameterized goodness function which represents all the aforementioned goodness functions, such as GF_{MDLP}, GF_{AIC}, GF_{BIC}, GF_{G^2}, and GF_{gini}, in a closed form.

Definition 3 *[32, 29] For a given interval S_i, the generalized entropy is defined as*

$$H_\beta(S_i) = \sum_{j=1}^{J} \frac{c_{ij}}{N_i}[1 - (\frac{c_{ij}}{N_i})^\beta]/\beta, \beta > 0$$

When $\beta = 1$, we can see that

$$H_1(S_i) = \sum_{j=1}^{J} \frac{c_{ij}}{N_i}[1 - \frac{c_{ij}}{N_i}] = gini(S_i)$$

When $\beta \to 0$,

$$H_{\beta \to 0}(S_i) = -\sum_{j=1}^{J} \frac{c_{ij}}{N_1} log \frac{c_{ij}}{N_1} = H(S_i)$$

Let $C_{I \times J}$ be a contingency table., We define the generalized entropy for C as follows.

$$H_\beta(S_1, \cdots, S_I) = \sum_{i=1}^{I} \frac{N_i}{N} H_\beta(S_i)$$

$$H_\beta(S_1 \cup \cdots \cup S_I) = \sum_{j=1}^{J} \frac{M_j}{N}[1 - (\frac{M_j}{N})^\beta]/\beta$$

4.3 Parameterized Goodness Function

Based on the discussion in Section 3, we derive that different goodness functions basically can be decomposed into two parts. The first part is for G^2, which corresponds to the information theoretical difference between the contingency table under consideration and the marginal distribution along classes. The second part is the penalty which counts the difference of complexity for the model between the contingency table under consideration and the one-row contingency table. The different goodness functions essentially have different penalties ranging from $O(df)$ to $O(dflogN)$.

In the following, we propose a parameterized goodness function which treats all the aforementioned goodness functions in a uniform way.

Definition 4 *Given two parameters, α and β, where $0 < \beta \leq 1$ and $0 < \alpha$, the parameterized goodness function for contingency table C is represented as*

$$GF_{\alpha,\beta}(C) = N \times H_\beta(S_1 \cup \cdots \cup S_{I'}) - \sum_{i=1}^{I'} N_i \times H_\beta(S_i) - \alpha \times (I'-1)(J-1)[1-(\frac{1}{N})^\beta]/\beta \quad (18)$$

By adjusting different parameter values, we show how goodness functions defined in section 2 can be obtained from the parametrized goodness function. We consider several cases:

1. Let $\beta = 1$ and $\alpha = 2(N-1)/(N(J-1))$. Then $GF_{2(N-1)/(N(J-1)),1} = GF_{gini}$.
2. Let $\alpha = 1/logN$ and $\beta \to 0$. Then $GF_{1/logN\beta\to 0} = GF_{AIC}$.
3. Let $\alpha = 1/2$ and $\beta \to 0$. Then $GF_{1/2,\beta\to 0} = GF_{BIC}$.
4. Let $\alpha = const, \beta \to 0$ and $N >> I$. Then $GF_{const,\beta\to 0} = G^2 - O(dflogN) = GF_{MDLP}$.
5. Let $\alpha = const, \beta \to 0$, and $G^2 = O(NlogJ), N/(IJ) \to \infty$. Then $GF_{const,\beta\to 0} = G^2 - O(dflogN) = GF''_{G^2} \approx GF''_{X^2}$.

The parameterized goodness function not only allows us to represent the existing goodness functions in a closed uniform form, but, more importantly, it provides a new way to understand and handle discretization. First, the parameterized approach provides a flexible framework to access a large collection (potentially infinite) of goodness functions. Any valid pair of α and β corresponds to a potential goodness function. Note that this treatment is in the same spirit of regularization theory developed in the statistical machine learning field [17, 34].

Secondly, finding the best discretization for different data mining tasks for a given dataset is transformed into a parameter selection problem. However, it is an open problem how we may automatically select the parameters without running the targeted data mining task. In other words, can we analytically determine the best discretization for different data mining tasks for a given dataset? This problem is beyond the scope of this paper and we plan to investigate it in future work.

Finally, the unification of goodness functions allows to develop efficient algorithms to discretize the continuous attributes with respect to different parameters in a uniform way. This is the topic of the next subsection.

4.4 Dynamic Programming for Discretization

This section presents a dynamic programming approach to find the best discretization function to maximize the parameterized goodness function. Note that the dynamic programming has been used in discretization before [14]. However, the existing approaches do not have a global goodness function to optimize, and almost all of them have to require the knowledge of targeted number of intervals. In other words, the user has to define the number of intervals for discretization. Thus, the existing approaches can not be directly applied to discretization for maximizing the parameterized goodness function.

In the following, we introduce our dynamic programming approach for discretization. To facilitate our discussion, we use GF for $GF_{\alpha,\beta}$, and we simplify the GF formula as follows. Since a given table C, $N \times H_\beta(S_1 \cup \cdots \cup S_I)$ (the first term in GF, Formula 18) is fixed, we define

$$F(C) = N \times H_\beta(S_1 \cup \cdots \cup S_I) - GF(C) = \sum_{i=1}^{I'} N_i \times H_\beta(S_i) + \alpha \times (I'-1)(J-1)[1-(\frac{1}{N})^\beta]/\beta$$

Clearly, the minimization of the new function F is equivalent to maximizing GF. In the following, we will focus on finding the best discretization to minimize F. First, we define a sub-contingency table of C as $C[i : i+k] = \{S_i, \cdots, S_{i+k}\}$, and let $C^0[i : i+k] = S_i \cup \cdots \cup S_{i+k}$ be the merged column sum for the sub-contingency table $C[i : i+k]$. Thus, the new function F of the row $C^0[i : i+k]$ is:

$$F(C^0[i : i+k]) = (\sum_{r=i}^{i+k} N_r) \times H_\beta(S_i \cup \cdots \cup S_{i+k})$$

Let C be the input contingency table for discretization. Let $Opt(i, i+k)$ be the minimum of the F function from the partial contingency table from row i to $i+k, k > 1$. The optimum which corresponds to the best discretization can be calculated recursively as follows:

$$Opt(i, i+k) = min(F(C^0[i : i+k]), min_{1\leq l\leq k-1}(Opt(i, i+l) + Opt(i+l+1, i+k) + \alpha \times (J-1)[1-(\frac{1}{N})^\beta]/\beta))$$

where $k > 0$ and $Opt(i, i) = F(C^0[i : i])$. Given this, we can apply the dynamic programming to find the discretization with the minimum of the goodness function, which are described in Algorithm 1. The complexity of the algorithm is $O(I^3)$, where I is the number of intervals of the input contingency table C.

Algorithm 1 Discretization(Contingency Table $C_{I\times J}$)

for $i = 1$ to I **do**
  for $j = i$ downto 1 **do**
    $Opt(j, i) = F(C^0[j : i])$
    for $k = j$ to $i - 1$ **do**
      $Opt(j, i) = min(Opt(j, i), Opt(j, k)+$
        $Opt(k + 1, i) + \alpha(J - 1)[1 - (\frac{1}{N})^{\beta}]/\beta)$
    end for
  end for
end for
return $Opt(1, I)$

Table 2: Summary of dataset

Dataset	Instances	Continuous Feature	Nominal Feature
anneal	898	6	32
australian	690	6	8
diabetes	768	8	0
glass	214	9	0
heart	270	13	0
hepatitis	155	6	13
hypothyroid	3168	7	18
iris	150	4	0
labor	57	8	8
liver	345	6	0
sick-euthyroid	3163	7	18
vehicle	846	18	0

5 Experimental Results

The major goal of our experimental evaluation is to demonstrate that the dynamic programming approach with appropriate parameters can significantly reduce the classification errors compared with the existing discretization approaches.

We chose 12 datasets from the UCI machine learning repository [39]. Most of the datasets have been used in the previous experimental evaluation for discretization study [13, 26]. Table 2 describes the size and the number of continuous and nominal features of each dataset.

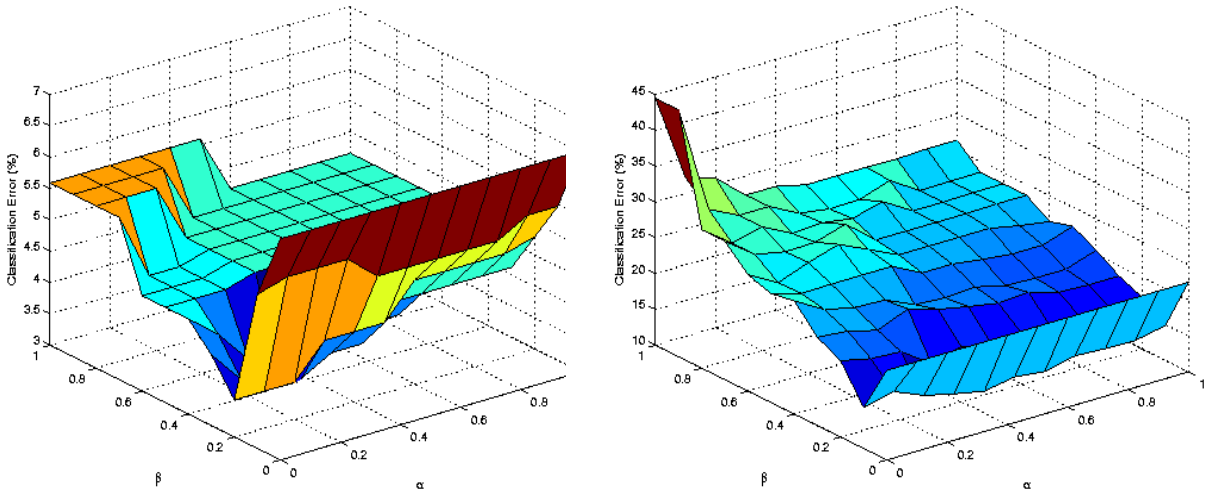
We apply discretization as a preprocessing step for two well-known classification methods: the C4.5 decision tree and Naive Bayes classifier [16]. For comparison purpose, we apply four discretization methods: *equal-width* (EQW), *equal-frequency* (EQF), *Entropy* [15], and *ChiMerge* [23]. The first two are unsupervised approaches and the last two are supervised approaches. We set the number of discretization intervals to be 10 for the first two. All their implementations are from Weka 3 [40].

Our dynamic programming approach for discretization (referred to as *Unification* in the experimental results) depends on two parameters, α and β. How to analytically determine the best parameters which can result in the minimal classification error is still an open question and beyond the scope of this paper. Here, we apply an experimental-validation approach to choose the optimal parameters α and β. For a given dataset and the data mining task, we create a 10×10 uniform grid for $0 \leq \alpha \leq 1$ and $0 \leq \beta \leq 1$ In addition, we use a value $10^{-}5$ to replace 0 for β since it cannot be equal to 0. Then we apply the dynamic programming at each grid point to discretize the dataset. We score each point using the mean classification error based on a five-trial five-fold cross-validation on the discretized data. Figure 1(a) shows the surface of the classification error rate of $C4.5$ running on the discretized *iris* dataset [39] using the unification approach with parameters from the 10×10 grid points. Figure 1(b) illustrates the surface of the classification error rate of Naive Bayes classifier running on the discretized *glass* dataset [39]. Clearly, we can see that different α and β parameters can result in very different classification error rates. Given this, we choose the α and β pair which achieves the minimal classification error rate as the selected unification parameters for discretization. For instance, in these two figures, we choose $\alpha = 0.3$ and $\beta = 0.3$ as the parameters to discretize *iris* for $C4.5$, and choose $\alpha = 0.4$ and $\beta = 0.1$ to discretize *glass* for Naive Bayes classifier. Note that the objective of using five trials instead of only one is to choose parameters in a more robust fashion to avoid outliers.

Finally, for each of the discretization method (our *Unification* method with the best predicated parameter), we run a five-trial five-fold cross-validation, and report their mean and standard deviation of the cross-validation. Note that here each trial will re-shuffle the dataset and is different from the trials in the parameter selection process.

Table 3 and Table 4 show the experimental results for C4.5 and Naive Bayes Classifier, respectively. In the left part of each table, we show the mean classification error and standard deviation using different discretization methods (the first one, *Continuous* corresponding to no-discretization). The right part of each table shows the percentage differences between two leading discretization approaches, *Entropy* and *ChiMerge*, with our new approach *Unification*. The last column chooses the minimal classification errors from all five existing approaches to compare with the unification approach.

We can see that the unification approach performs significantly better than the existing approaches. First, based on the average classification error for all the 12 datasets, the unification is the best among all these approaches (14.45% error rate for C4.5 and 10.60% for Naive Bayes classifier). For C4.5, it reduces the error rate on an average of 19.40% compared with *Entropy*, and reduces the error rate on average of 26.65% compared with *ChiMerge*. For Naive Bayes classifier, it reduces the error rate on an average of 58.74% compared with *Entropy*, and reduces the error rate on an average of 20.82% compared with *ChiMerge*. The overall improvement is on an average of 31% in terms of classification error rate. Finally, in 9 out of 12 datasets for C4.5, the unification approach shows better or equal performance with the best existing approach. In other 3 datasets, the performance are fairly close to the minimal error rate as well. For Naive Bayes classifier, the unification method perform the best in 10 out of 12 datasets and the second for the other 2 datasets.



(a) Iris+C4.5 (b) glass+Naive Bayes classifier

Figure 1: The surface of classification error rate using parameters from 10×10 grid

Table 3: C4.5 Results

Dataset	Experimental results 5x5 validation						Comparision with Unification		
	Continuous	EQW	EQF	Entropy	ChiMerge	Unification	Entropy	ChiMerge	Min
anneal	8.62±2.16	9.84±2.07	9.40±1.95	8.58±1.49	7.32±1.19	7.33±1.40	17.05	-0.14	-0.14
australian	14.34±2.55	15.10±2.99	12.83±3.51	13.70±2.94	14.64±3.11	12.46±2.78	9.95	17.50	2.97
diabetes	26.07±3.07	25.84±2.83	26.02±3.11	22.75±3.13	26.05±3.02	22.34±2.35	1.84	16.61	1.84
glass	33.44±6.33	44.29±6.09	42.05±4.78	26.72±7.52	27.23±5.39	25.15±4.64	6.24	8.27	6.24
heart	20.30±5.18	21.42±5.21	22.01±4.62	16.52±4.01	20.60±5.30	16.52±4.01	0.00	24.70	0.00
hepatitis	18.60±5.85	16.92±4.96	15.88±5.06	19.36±5.38	17.81±5.19	15.36±5.03	26.04	15.95	3.39
hypothyroid	0.78±0.26	2.69±0.54	1.73±0.39	0.76±0.30	1.69±0.48	0.76±0.30	0.00	122.37	0.00
iris	5.60±3.33	4.00±2.98	6.01±2.04	5.46±3.28	3.34±3.13	3.06±3.43	78.43	9.15	9.15
labor	15.77±7.75	28.84±7.06	28.05±6.76	14.37±10.84	9.09±8.10	9.82±9.33	46.33	-7.43	-7.43
liver	34.54±5.05	39.18±5.44	42.77±5.60	36.80±5.23	34.32±3.72	30.54±4.93	20.50	12.38	12.38
sick-euthyroid	2.09±0.53	3.94±0.62	4.96±0.67	2.49±0.64	4.11±0.86	2.09±0.52	19.14	96.65	0.00
vehicle	27.64±3.41	30.62±2.50	34.51±3.10	30.00±1.94	29.03±3.51	27.97±3.53	7.26	3.79	-1.18
Average	17.32	20.22	20.52	16.46	16.27	14.45	19.40	26.65	2.27

Table 4: Naive Bayes Results

Dataset	Experimental results 5x5 validation						Comparisons with Unification		
	Continuous	EQW	EQF	Entropy	ChiMerge	Unification	Entropy	ChiMerge	Min
anneal	20±2.15	6.1±1.55	3.07±1.4	3.67±1.45	2.36±1.25	2.18±1.15	68.35	8.26	8.26
australian	22.55±2.6	14.81±3	13.48±2.7	14.29±3.1	10.52±2.25	10.11±1.95	41.28	4.00	4.00
diabetes	24.45±2.85	24.40±3.6	25.36±3.8	22.03±2.75	15.26±2.35	13.13±2.6	67.82	16.26	16.26
glass	53.74±7	42.06±5.6	28.59±5.35	25.8±4	20±4.6	13.36±4.45	93.04	49.64	49.64
heart	16±5.7	15.70±3.3	16.96±2.7	16.3±4.55	12.885±3.45	12.74±3.05	27.94	1.14	1.14
hepatitis	16.12±5.6	16.12±4.05	17.42±5.7	14.32±3.95	9.93±3.35	10.32±3.9	38.76	-3.73	-3.73
hypothyroid	2.11±0.45	2.99±0.45	2.835±0.75	1.36±0.5	1.23±0.5	0.99±0.45	37.37	24.75	24.75
iris	4.26±4.15	5.2±3.65	7.33±4.55	5.73±4.55	4.93±4.15	2.8±3.95	104.64	76.07	52.32
labor	8.03±6.25	8.48±8.7	9.42±9.65	6.36±6.85	5.3±5.35	2.88±4.4	120.83	84.03	84.03
liver	44.87±7.6	36.23±3.8	37.97±5.2	36.81±5.3	20.465±4.25	20.23±4.15	81.96	1.16	1.16
sick-euthyroid	15.63±2.15	6.375±0.85	5.855±1.05	3.8±0.8	3.32±0.65	3.26±0.7	16.56	1.84	1.84
vehicle	55.05±3.1	39.52±3.2	37.28±2.6	37.49±3.5	30.47±2.95	35.27±2.9	6.31	-13.60	-13.60
Average	23.57	18.17	17.13	15.66	11.39	10.61	58.74	20.82	18.84

6 Conclusions

In this paper we introduced a generalized goodness function to evaluate the quality of a discretization method. We have shown that seemingly disparate goodness functions based on entropy, AIC, BIC, Pearson's X^2, Wilks' G^2, and Gini index are all derivable from our generalized goodness function. Furthermore, the choice of different parameters for the generalized goodness function explains why there is a wide variety of discretization methods. Indeed, difficulties in comparing different discretization methods were widely known. Our results provide a theoretical foundation in understanding these difficulties and offer rationale as to why evaluation of different discretization methods for an arbitrary contingency table is difficult. We have designed a dynamic programming algorithm that for given set of parameters of a generalized goodness function provides an optimal discretization which achieves the minimum of the generalized goodness function. We have conducted an extensive performance tests for a set of publicly available data sets. Our experimental results demonstrate that our discretization method consistently outperforms the existing discretization metods on the average by 31%. These results clearly validate our approach and open a new way of tackling discretization problems.

References

[1] A. Agresti *Categorical Data Analysis*. Wiley, New York, 1990.

[2] H. Akaike. Information Theory and an Extension of the Maximum Likelihood Principle. In *Second International Symposium on Information Theory, 267-281*, Armenia, 1973.

[3] P. Auer, R. Holte, W. Maass. Theory and Applications of Agnostic Pac-Learning with Small Decision Trees. In *Machine Learning: Proceedings of the Twelth International Conference*, Morgan Kaufmann, 1995.

[4] L. Breiman, J. Friedman, R. Olshen, C. Stone *Classification and Regression Trees*. CRC Press, 1998.

[5] M. Boulle. Khiops: A Statistical Discretization Method of Continuous Attributes. *Machine Learning, 55, 53-69*, 2004.

[6] M. Boulle. MODL: A Bayes optimal discretization method for continuous attributes. *Mach. Learn. 65, 1 (Oct. 2006)*, 131-165.

[7] George Casella and Roger L. Berger. Statistical Inference (2nd Edition). *Duxbury Press,* 2001.

[8] J. Catlett. On Changing Continuous Attributes into Ordered Discrete Attributes. In *Proceedings of European Working Session on Learning, p. 164-178*, 1991.

[9] J. Y. Ching, A.K.C. Wong, K. C.C. Chan. Class-Dependent Discretization for Inductive Learning from Continuous and Mixed-Mode Data. *IEEE Transactions on Pattern Analysis and Machine Intelligence, V. 17, No. 7, 641-651*, 1995.

[10] M.R. Chmielewski, J.W. Grzymala-Busse. Global Discretization of Continuous Attributes as Preprocessing for Machine Learning. *International Journal of Approximate Reasoning, 15*, 1996.

[11] Y.S. Choi, B.R. Moon, S.Y. Seo. Genetic Fuzzy Discretization with Adaptive Intervals for Classification Problems. *Proceedings of 2005 Conference on Genetic and Evolutionary Computation, pp. 2037-2043*, 2005.

[12] Thomas M. Cover and Joy A. Thomas, Elements of Information Thoery, Second Edition. *Published by John Wiley & Sons, Inc.*, 2006.

[13] J. Dougherty, R. Kohavi, M. Sahavi. Supervised and Unsupervised Discretization of Continuous Attributes. *Proceedings of the 12th International Conference on Machine Learning, pp. 194-202*, 1995.

[14] Tapio Elomaa and Juho Rousu. Efficient Multisplitting Revisited: Optima-Preserving Elimination of Partition Candidates. *Data Mining and Knowledge Discovery, 8, 97-126*, 2004.

[15] U.M. Fayyad and K.B. Irani Multi-Interval Discretization of Continuous-Valued Attributes for Classification Learning. In *Proceedings of the 13th Joint Conference on Artificial Intelligence, 1022-1029*, 1993.

[16] David Hand, Heikki Mannila, Padhraic Smyth. *Principles of Data Mining* MIT Press, 2001.

[17] Federico Girosi, Michael Jones, and Tomaso Poggio. Regularization theory and neural networks architectures. In *Neural Computation, Volume 7 , Issue 2 (March 1995), Pages: 219 - 269*.

[18] M.H. Hansen, B. Yu. Model Selection and the Principle of Minimum Description Length. *Journal of the American Statistical Assciation, 96, p. 454*, 2001.

[19] R.C. Holte. Very Simple Calssification Rules Perform Well on Most Commonly Used Datasets. *Machine Learning, 11, pp. 63-90*, 1993.

[20] Janssens, D., Brijs, T., Vanhoof, K., and Wets, G. Evaluating the performance of cost-based discretization versus entropy-and error-based discretization. *Comput. Oper. Res. 33, 11 (Nov. 2006), 3107-3123.*

[21] N. Johnson, S. Kotz, N. Balakrishnan. Continuous Univariate Distributions, Second Edition. *John Wiley & Sons, INC.*, 1994.

[22] *Technical Report*, 2007.

[23] Randy Kerber. ChiMerge: Discretization of Numeric Attributes. *National Conference on Artificial Intelligence*, 1992.

[24] L.A. Kurgan, K.J. Cios CAIM Discretization Algorithm. *IEEE Transactions on Knowledge and Data Engineering, V. 16, No. 2, 145-153*, 2004.

[25] R. Kohavi,M. Sahami. Error-Based and Entropy-Based Discretization of Continuous Features. *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, pp. 114-119*, Menlo Park CA, AAAI Press, 1996.

[26] Huan Liu, Farhad Hussain, Chew Lim Tan, Manoranjan Dash. Discretization: An Enabling Technique. *Data Mining and Knowledge Discovery, 6, 393-423*, 2002.

[27] H. Liu and R. Setiono. Chi2: Feature selection and discretization of numeric attributes. *Proceedings of 7th IEEE Int'l Conference on Tools with Artificial Intelligence*, 1995.

[28] X. Liu, H.Wang A Discretization Algorithm Based on a Heterogeneity Criterion. *IEEE Transaction on Knowledge and Data Engineering, v. 17, No. 9, 1166-1173*, 2005.

[29] S. Mussard, F. Seyte, M. Terraza. Decomposition of Gini and the generalized entropy inequality measures. *Economic Bulletin, Vol. 4, No. 7, 1-6*, 2003.

[30] B. Pfahringer. Supervised and Unsupervised Discretization of Continuous Features. *Proceedings of 12th International Conference on Machine Learning, pp. 456-463*, 1995.2003.

[31] J. Rissanen Modeling by shortest data description *Automatica, 14,pp. 465-471*, 1978.

[32] D.A. Simovici and S. Jaroszewicz An axiomatization of partition entropy *IEEE Transactions on Information Theory, Vol. 48, Issue:7, 2138-2142*, 2002.

[33] Robert A. Stine. Model Selection using Information Theory and the MDL Principle. In *Sociological Methods & Research, Vol. 33, No. 2, 230-260*, 2004.

[34] Trevor Hastie, Robert Tibshirani and Jerome Friedman. The Elements of Statistical Learning *Springer-Verlag*, 2001.

[35] David L. Wallace . Bounds on Normal Approximations to Student's and the Chi-Square Distributions. *The Annals of Mathematical Statistics, Vol. 30, No. 4, pp 1121-1130*, 1959.

[36] David L. Wallace . Correction to "Bounds on Normal Approximations to Student's and the Chi-Square Distributions". *The Annals of Mathematical Statistics, Vol. 31, No. 3, p. 810*, 1960.

[37] A.K.C. Wong, D.K.Y. Chiu. Synthesizing Statistical Knowledge from Incomplete Mixed-Mode Data. *IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 9, NNo. 6, pp. 796-805,* 1987.

[38] Ying Yang and Geoffrey I. Webb. Weighted Proportional k-Interval Discretization for Naive-Bayes Classifiers. In *Advances in Knowledge Discovery and Data Mining: 7th Pacific-Asia Conference, PAKDD, page 501-512*, 2003.

[39] `http://www.ics.uci.edu/mlearn/ML.Repository.html`

[40] `http://www.cs.waikato.ac.nz/ml/weka`