文章编号: 1001-0920(2011)04-0592-05

一种基于类别属性关联程度最大化离散算法

杨 萍, 杨天社, 杜小宁, 李济生, 黄永宣

(西安交通大学 系统工程研究所, 西安 710049)

摘 要: 针对现有离散化算法难以兼顾计算速度和求解质量这一难题, 提出一种新的基于类别属性关联程度最大化监督离散化算法. 该算法考虑了类别、属性值的空间分布特征, 根据类别与属性之间的内在联系构造离散化框架, 使离散化后类别和属性的关联程度最大. 实验结果表明, 基于类别属性关联程度最大化离散算法在保证计算速度的前提下能有效提高分类精度, 减少分类规则个数.

关键词: 离散化; 关联程度最大化; 分类; 数据挖掘

中图分类号: TP391.9 **文献标识码:** A

A class-attribute interdependency maximization based algorithm for supervised discretization

YANG Ping, *YANG Tian-she*, *DU Xiao-ning*, *LI Ji-sheng*, *HUANG Yong-xuan*

(Systems Engineering Institute, Xi'an Jiaotong University, Xi'an 710049, China. Correspondent: YANG Ping, E-mail: xjtuyp@gmail.com)

Abstract: Considering that the existing discretization algorithms do not give simultaneously attention to evolution speed and solution's quality, a new class-attribute interdependency maximization based algorithm for supervised discretization is proposed in this paper. The algorithm considers the distribution of both class and continuous attributes, and according to the underlying correlation structure of them, the discretization scheme is constructed which can maintain the highest interdependence between the target class and all the discretized attributes. The experiment results show that, with a reasonable execution time, the proposed algorithm can improve the accuracy of the classification result and reduce the number of classification rules.

Key words: discretization; interdependency maximization; classification; data mining

1 引 言

数据挖掘是一种从大型复杂数据库中提取有用信息的强有力的工具. 但是, 数据挖掘工具的成功应用很大程度上依赖于数据库的质量. 因此, 数据的预处理是数据挖掘领域中一个至关重要的研究课题. 现实世界中常常涉及到连续的数值属性, 而许多数据挖掘算法要求所处理的属性取离散值, 因此, 必须对连续的数值属性进行离散化处理, 将其变为离散的符号量. 离散化是一种基本的数据预处理技术, 受到人们越来越多的重视.

根据不同的划分方式, 离散化算法可分为 3 类: 根据是否利用整个属性空间进行离散化处理, 可分为整体离散化和局部离散化; 根据离散化处理时是否使用了类别属性作参考, 可分为监督离散化和非监督离散化; 根据离散化与决策树的生成是否同时进行, 分为静态离散化和动态离散化.

典型的离散化方法有: 等宽算法; 等频算法; 基于信息熵的 D2[1], Ent-MDLP[2]算法; 基于统计量的 ChiMerge[3], Chi2[4], StatDisc[5]算法; 基于类别属性关联程度的 CAIM[6], CADD[7]算法; 基于聚类的 K 均值离散化算法[8]; 基于粗糙集与布尔逻辑相结合的离散化算法[9]等. 近年来, 人们进行了多种尝试, 从各个角度提出了许多可行有效的算法. Mehta 等人[10]提出一种基于主成分分析的非监督离散化算法, 通过应用主成分分析, 能有效地对数据进行缩减, 适用于高维大型的数据库. Tsai 等人[11]提出一种基于类别属性相依系数的离散化算法, 扩展了相依系数的概念, 并将其与贪婪算法相结合, 提高了离散化结果的质量. Ruiz 等人[12]提出一种基于间隔误差的监督离散化算法, 定义了一种邻域概念, 对类别属性变量的取值顺

收稿日期: 2010-01-22; **修回日期:** 2010-05-20.

作者简介: 杨萍(1981–), 女, 博士生, 从事卫星故障诊断的研究; 李济生(1943–), 男, 中国科学院院士, 教授, 博士生导师, 从事卫星轨道动力学和卫星测控等研究.

序敏感, 适用于类别属性变量取值连续的情况.

现有的离散化算法中, 绝大多数都属于单属性离散化方法, 离散化过程中没有考虑属性之间的内在联系, 因此存在计算复杂、离散化质量不高等问题. 一些基于粗糙集的算法, 例如粗糙集和布尔逻辑相结合的算法, 考虑了属性之间的相互影响, 但是算法的计算复杂度太高 ($o(n^3k)$, n 为对象的个数, k 为属性的个数). 对此, 本文提出一种基于类别属性关联程度最大化离散算法, 将算法在 7 个不同类别的数据库上进行实验, 并与其他 3 种典型的离散化算法进行对比. 实验结果表明了本文算法的有效性和实用性.

2　基于类别属性关联程度最大化离散算法

2.1　类别属性关联程度最大化 (CAIM) 准则

CAIM 准则[6]是Kurgan 和 Cios在 2004 年提出的, 它是类别和属性值之间关联程度的一种度量. 下面对 CAIM 准则进行简单的介绍.

给定一个数据库 S, 它包含 n 个对象, l 个连续条件属性, 对象分属于 k 个类别. 假定 c 为其中任意一个连续属性, 则存在一个离散化框架, 将 c 的值域划分为 m 个离散化区间, 由如下断点来标定: $\{[b_0, b_1], (b_1, b_2], \cdots, (b_{m-1}, b_m]\}$. 其中: b_0 为属性 c 的最小值, b_m 为 c 的最大值, 对于任意的 $i\,(0 \leqslant i < m)$, $b_i < b_{i+1}$.

对于一个确定的离散化框架, 类别变量和属性 c 的离散化区间构成了表 1 所示的一个二维矩阵, 也称量子矩阵. 在表 1 中, q_{ir} 表示类别为 d_i 且属性 c 的取值属于区间 $(b_{r-1}, b_r]$ 的对象的个数, n_{i+} 表示类别为 d_i 的对象的个数, n_{+r} 表示属性 c 的取值属于区间 $(b_{r-1}, b_r]$ 的对象的个数, $i = 1, 2, \cdots, k$, $r = 1, 2, \cdots, m$.

表 1　二维量子矩阵

类别/区间	$[b_0, b_1]$	$\cdots$	$(b_{r-1}, b_r]$	$\cdots$	$(b_{m-1}, b_m]$	总和
d_1	q_{11}	$\cdots$	q_{1r}	$\cdots$	q_{1m}	n_{1+}
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
d_i	q_{i1}	$\cdots$	q_{ir}	$\cdots$	q_{im}	n_{i+}
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
d_k	q_{k1}	$\cdots$	q_{kr}	$\cdots$	q_{km}	n_{k+}
总 和	n_{+1}	$\cdots$	n_{+r}	$\cdots$	n_{+m}	n

对于一个给定的量子矩阵, CAIM 准则定义如下:

$$\mathrm{CAIM} = \frac{1}{m}\sum_{r=1}^{m}\frac{\max_r^2}{n_{+r}}, \tag{1}$$

其中 $\max_r$ 表示当 i 的取值变化时, q_{ir} 的最大值. 如果 $\max_r = q_{hr}, 1 \leqslant h \leqslant k$, 则称 d_h 为区间 $(b_{r-1}, b_r]$ 中的主导类. 主导类中包含对象越多, CAIM 值越大, 类别与离散化区间的关联程度越大.

2.2　断点的选择标准

应用 CAIM 规则求取离散化框架时, CAIM 值的变化会出现两种趋势: 1) 随着断点个数的增加, CAIM 的值增加, 直到出现理论最大值 CAIM $= n/k$. 此时离散化框架最优, 离散化区间的个数为 k, 且每个离散化区间中, 对象的类别相同, 即在表 1 中 $m = k$, $q_{hr} = n_{+r}$. 2) 随着断点的增加, CAIM 值达到局部最大, 然后开始下降, 且局部最大值往往出现在第 1 个断点处. 在实际应用中, 绝大多数的数据库中对象没有规则的类别分布, CAIM 值的变化呈现出第 2 种趋势. 在这种情况下, 为使类别属性的关联程度达到最大, 必须选取足够的断点, 使每个离散化区间中对象的类别相同. 否则, 只考虑每个离散化区间中的主导类而忽略其他类别的对象时, 会丢失很多有用的信息, 使离散化结果的质量下降.

CAIM 准则测量的是类别和单个属性之间的联系, 因此会出现上面的缺点. 在实际应用中, 仅依靠单个属性是不能区别对象的, 因此, 应考虑类别和所有属性之间的联系, 使得分类的结果更加准确.

本文提出一种基于类别属性关联程度最大化离散算法, 试图找到一种离散化框架, 使类别和所有属性之间的关联程度达到最大. 换言之, 算法是寻找一个最小断点集合, 这些断点将连续属性空间划分为有限的超立方体, 每个超立方体中对象的类别相同. 由于 CAIM 准则能测量类别和单个属性之间的联系, 可用它来进行局部断点的选择. 关键的问题是如何决定断点的个数. 分两种情况: 当 CAIM 值的变化呈现出第 1 种趋势时, 容易确定断点的个数为 $k-1$; 当 CAIM 值的变化呈现出第 2 种趋势时, 如何确定合理的断点个数则比较困难.

在本文算法中, 当出现第 2 种情况时, 期望每个属性的离散化区间个数为 k, 且每个离散化区间中, 主导类的个数大于某个阈值. 以表 1 为例, 假设 R 为一个常数, 且 $0.5 \leqslant R \leqslant 1$, 则 $\max_r > Rn_{+r}$, $r = 1, 2, \cdots, k$. 于是, 当一个属性的离散化过程完成之后, CAIM 的最小期望值为

$$\mathrm{CAIM}_{\exp} = \frac{1}{k}\sum_{r=1}^{m}\frac{\max_r^2}{n_{+r}} = \frac{1}{k}\sum_{r=1}^{m}\frac{(Rn_{+r})^2}{n_{+r}} = \frac{nR^2}{k}. \tag{2}$$

假定 $\mathrm{CAIM}_{\mathrm{loc}}$ 为局部最大值, 为了得到预期的离散化结果, 每增加一个断点时, CAIM 值下降的幅度不能超过 $(\mathrm{CAIM}_{\mathrm{loc}} - \mathrm{CAIM}_{\exp})/(k-1)$.

2.3 算法步骤和时间复杂度分析

2.3.1 算法步骤

给定一个训练数据库S, 它包含n个对象$U=\{x_1,x_2,\cdots,x_n\}$, l个连续条件属性$C=\{c_1,c_2,\cdots,c_l\}$, k个决策类. V_{c_i}为属性c_i的值域, $|V_{c_i}|$为集合V_{c_i}的势. R为常量, 且$0.5\leqslant R\leqslant 1$. 算法步骤如下:

Step 1: 对于待离散属性, 初始化其候选断点集合和离散化框架.

Step 1.1: 初始化$i=1$, $\text{Object}_1=\{U\}$.

Step 1.2: 初始化$\text{CutInt}=\varnothing$, $\text{Object}_{i+1}=\varnothing$, $r=1$. 假定集合Object_i的第r个元素为U^r. 对于U^r, 属性c_i的值域为V_{ci}^r, 决策类为k^r.

Step 1.3: 将集合V_{ci}^r中的值按照升序排列$V_{ci}^r=\{c_i^0,c_i^1,\cdots,c_i^m\}$, 初始化候选断点集合$B=\{(c_i^0+c_i^1)/2,(c_i^1+c_i^2)/2,\cdots,(c_i^{m-1}+c_i^m)/2\}$. (c_i^0,c_i^1)为断点$(c_i^0+c_i^1)/2$的断点区间.

Step 1.4: 设定初始化离散框架为$\{[c_i^0,c_i^m]\}$, $\text{Global}=0$.

Step 2: 从候选断点集合中不断添加断点, 选择使CAIM值最大或CAIM值下降幅度不超过一定阈值的断点.

Step 2.1: 初始化$h=1$, CAIM的最小期望值$\text{CAIM}_{\text{exp}}=|U^r|R^2/k^r$.

Step 2.2: 从B中选择一个断点, 加入离散化框架中, 计算相应的CAIM值.

Step 2.3: 当B中的点被遍历后, 选取一个使CAIM取值最大的断点, 此时CAIM值记为CAIM_h.

Step 2.4: 如果$\text{CAIM}_h>\text{Global}$或$(\text{Global}-\text{CAIM}_h)<(\text{CAIM}_1-\text{CAIM}_{\text{exp}})/(k^r-1)$, 则令$\text{Global}=\text{CAIM}_h$, 将Step 2.3中选取的断点加入离散化框架中, 将对应的断点区间添加到CutInt中, 并将该断点从B中删除; 否则, 转Step 2.6.

Step 2.5: 令$h=h+1$, 如果$h<k^r$, 则转Step 2.2.

Step 2.6: 假设离散化框架将对象集合U^r分成若干子集合, U'是其中任意一个. 如果U'中的对象属于不同的类别, 则更新$\text{Object}_{i+1}=\text{Object}_{i+1}\bigcup\{U'\}$.

Step 2.7: 令$r=r+1$. 如果$r\leqslant|\text{Object}_i|$, 则转Step 1.3.

Step 3: 删除冗余断点.

Step 3.1: 将CutInt中的断点区间按照升序排列. 如果两个相邻的区间有交集, 则将这两个区间用交集代替.

Step 3.2: 计算CutInt中所有区间的中点, 它们即为属性c_i的断点.

Step 3.3: 令$i=i+1$, 如果$i\leqslant l$, 则转Step 1.2; 否则, 结束.

上述算法中, 符号Object表示对象集的集合. 对c_i进行离散化处理时, $c_1,c_2,\cdots,c_{i-1}$的断点将连续属性空间划分为有限个超立方体. Object_i中的一个元素为某个超立方体中对象的集合. 符号CutInt表示断点区间的集合. 符号Global表示在本步计算中CAIM的最大值.

2.3.2 时间复杂度分析

以属性c_i为例, 假设在c_{i-1}离散化结束后, 对象集合被分成M个子集, 即$\text{Object}_i=\{U^1,U^2,\cdots,U^M\}$. 其中: $|U^r|=nP_r$, $r=1,2,\cdots,M$, $0<P_r<1$, 并且

$$\sum_{r=1}^{M}P_r=1. \tag{3}$$

在Step 1.3中, 对$V_{c_i}^r$中的值排序需要时间$o(n\times P_r\log(nP_r))$, 对所有的$V_{c_i}^r(r=1,2,\cdots,M)$排序, 所需的时间总和为

$$\sum_{r=1}^{M}o(nP_r\log(nP_r)), \tag{4}$$

它决定了Step 1的时间复杂度.

Step 2的时间复杂度取决于CAIM值的计算. 对于任意一个对象子集U^r, 离散化过程开始于一个单个区间, 并且期间个数的最大期望值为$o(k)$. 因此, 计算CAIM值的时间复杂度为$o(k^2)$. 对于U^r, 候选断点的个数为nP_r-1, 在Step 2.2中, CAIM的值被计算了$o(nP_r)$次, 因而决定了Step 2.2的时间复杂度为$o(nP_rk^2)$. U^r离散区间的个数期望值为$o(k)$, Step 2.2被执行$o(k)$次. 因此, 对于所有的$U^r(r=1,2,\cdots,M)$, Step 2的时间复杂度为

$$\sum_{r=1}^{M}o(nP_rk^2)o(k)=o(nk^3). \tag{5}$$

相对于对象个数n而言, 属性c_i的断点个数是非常小的, 因此Step 3的时间复杂度可以忽略不计.

基于以上分析可以得出结论: 离散化单个属性c_i, 其时间复杂度为

$$\sum_{r=1}^{M}o(nP_r\log(nP_r))+o(nk^3). \tag{6}$$

命题 1 假定n是问题的规模, 并且

$$0<P_r<1,\ \sum_{r=1}^{M}P_r=1,$$

则问题的时间复杂度为

$$\sum_{r=1}^{M}o(nP_r\log(nP_r))=o(n\log(n)).$$

证明 因为

$$nP_r\log(nP_r)=nP_r(\log(n)+\log(P_r))=$$
$$P_rn\log(n)+nP_r\log(P_r),$$

所以有

$$\sum_{r=1}^{M}nP_r\log(nP_r)=$$
$$\Big(\sum_{r=1}^{M}P_r\Big)n\log(n)+n\sum_{r=1}^{M}P_r\log(P_r)=$$
$$n\log(n)+n\sum_{r=1}^{M}P_r\log(P_r),$$
$$\sum_{r=1}^{M}o(nP_r\log(nP_r))=$$
$$o\Big(\sum_{r=1}^{M}nP_r\log(nP_r)\Big)=$$
$$o\Big(n\log(n)+n\sum_{r=1}^{M}P_r\log(P_r)\Big)=$$
$$o(n\log(n)).\qquad\square$$

连续属性的个数为 l, 离散化单个属性的时间复杂度为 $o(n\log(n))+o(nk^3)$. 在实际应用中, l 和 k 的值都很小, 因此, 整个算法的时间复杂度为 $o(n\log(n))$.

3　实验结果分析

为验证本文算法的有效性, 从著名的 UCI 机器学习数据库[13]中选取 7 个数据集进行实验, 并将 Ent-MDLP, ChiMerge 和 CAIM 三种算法与本文算法离散化结果进行对比. 表 2 给出了这 7 个数据集的详细信息. ChiMerge 算法需要预先指定参数, 为了得到较好的离散化结果, 这些参数的取值如表 3 所示. 本文算法中, 参数 $R=0.8$.

表 2　数据集信息

基本参数	数据集						
	iris	ion	cancer	pima	yeast	page	thy
对象个数	150	351	569	768	1 484	5 473	7 200
属性个数	4	34	30	8	8	10	21
连续属性	4	33	30	8	8	10	6
类别	3	7	2	10	5	6	3

表 3　ChiMerge 算法的参数取值

参　数	数据集						
	iris	ion	cancer	pima	yeast	page	thy
χ^2-threshold	0.95	0.99	0.99	0.95	0.95	0.95	0.95
max−interval	6	3	3	6	10	6	6

下面用 C5.0 算法来评价以上 4 种离散化算法的性能. 在数据挖掘领域中, 人们提出了多种决策树的分类算法, C5.0 算法是具有代表性的一种. 它根据最大信息增益自动计算出达到最佳决策树时各叶子的阈值, 从而提高了决策树的客观性和精度. 但是, 在连续数值属性离散化处理方面, C5.0 算法存在两个不足: 1) 由于采用范围划分的方式离散化, 在创建决策树的分类准确率方面有所下降; 2) 因采用信息增益率作为结点分割的标准, 使用价值函数来评价每一个可能的分割点, 计算量偏大.

由于 C5.0 算法能直接处理连续数据, 将连续数据和 4 种算法的离散化结果分别作为 C5.0 算法的输入, 用来构造决策树, 产生分类规则. 每个输入数据集被随机分为 10 组, 其中 9 组作为训练数据, 1 组作为测试数据. 实验重复 10 次, 结果取 10 次的平均值. 对预期错误分类率、提取规则个数、建立决策树的时间进行对比, 对比结果见表 4 和表 5. 其中: 表 4, 表 5 的最后一列是对实验结果的最直接的比较, "RANK" 值[14]定义为

$$\mathrm{RANK}_j=\frac{1}{N}\sum_i r_i^j. \tag{7}$$

这里: N 为数据集的个数, r_i^j 为第 j 个算法在第 i 个数据集上的级别. 对于一个特定的数据集而言, 效果最好的算法级别为 1. 表 4, 表 5 中的粗体数值表示最好的结果.

表 4　预期错误分类率和规则个数的对比

评价标准	离散算法	数据集							RANK
		iris	ion	cancer	pima	yeast	page	thy	
预期错误分类率	Continuous data set	7.3±3.2	10.6±1.9	6.5±1.2	26.8±1.9	44.5±0.7	**3.0**±0.3	**0.3**±0.1	3.4
	Ent-MDLP	6.7±2.2	10.5±1.4	4.6±0.7	24.7±1.2	**41.9**±0.8	3.3±0.3	0.7±0.1	2.7
	ChiMerge	6.0±1.8	9.4±1.9	5.6±1.2	25.4±1.4	47.3±1.0	3.5±0.3	0.8±0.2	3.7
	CAIM	6.0±1.2	9.7±1.1	**4.4**±1.0	24.5±1.1	44.8±1.0	3.4±0.2	1.3±0.1	2.7
	本文算法	**4.0**±1.1	**5.4**±1.5	5.4±1.0	**24.2**±1.3	45.8±1.3	3.4±0.2	**0.3**±0.1	**2**
规则个数	Continuous data set	3.9±0.2	13.3±1.1	11.1±0.9	23.5±2.5	162.8±7.6	**28.2**±1.2	15.5±0.7	4
	Ent-MDLP	3.2±0.2	9.4±0.5	8.8±0.3	9.0±0.4	**34.8**±1.0	35.8±1.2	21.5±0.5	3.1
	ChiMerge	4.0±0.3	7.6±0.5	9.4±0.6	15.5±1.5	101.9±3.2	37.3±1.3	26.2±0.8	4.1
	CAIM	3.1±0.1	8.8±0.4	**7.3**±0.2	5.1±0.5	90.9±2.2	33.9±0.9	**10.0**±0.3	2.1
	本文算法	**3.0**±0.0	**5.4**±0.3	8.5±0.2	**3.8**±0.2	39.2±1.2	28.8±0.9	**10.0**±0.2	**1.4**

表 5 构造决策树所需时间的对比

离散算法	数据集							RANK
	iris	ion	cancer	pima	yeast	page	thy	
Continuous data set	0.2	0.8	0.6	**0.7**	2	2.7	1	4.1
Ent-MDLP	**0.1**	0.5	0.5	**0.7**	**0.8**	0.8	0.6	2.1
ChiMerge	0.2	0.6	0.5	1.2	1.6	**0.7**	0.7	3.1
CAIM	0.2	0.4	**0.4**	0.7	1.6	**0.7**	**0.5**	**1.7**
本文算法	**0.1**	**0.3**	**0.4**	1.2	1.3	0.8	**0.5**	1.9

从表 4 可以看出, 本文算法在 4 个数据集上均取得了最小预期错误分类率, 并且在所有数据集上的“RANK”值最高. 这组对比结果说明, 在这 4 种算法中, 本文算法的分类精度最高.

与其他算法相比, 在 4 个数据集上本文算法产生的分类规则数最少; 在 cancer, yeast 和 page 上次之. 在这组实验中, 本文算法的平均级别最高, 说明从分类规则个数的角度, 本文算法的效果最好.

用 C5.0 算法构造决策树时, CAIM 算法所需的平均执行时间最短, 本文算法次之. 但是, CAIM 算法和本文算法所需的时间是可以比较的.

通过以上的对比分析, 可得出以下结论: 与其他 3 种离散化算法相比, 本文算法所得的离散化结果能提高后续机器学习算法的分类精度, 减少分类规则的个数, 缩短建立决策树的时间.

4 结 论

本文提出一种新的基于类别属性关联程度最大化的监督离散化算法. 该算法采取一种自上而下的划分方式, 将 n 维连续属性空间划分为有限个超立方体, 每个超立方体中对象的类别属性相同. 算法不受对象类别个数和属性维数的限制, 且断点个数不需要预先设定. 将本文算法与其他 3 种典型离散化算法的结果作为 C5.0 算法的输入, 用来构造决策树, 产生分类规则. 对比结果显示, 本文算法能取得最优的离散化结果.

参考文献(References)

[1] Catlett J. On changing continuous attributes into ordered discrete attributes[C]. Proc of European Working Session on Learning. Porto: Kodratoff, 1991: 164-178.

[2] Fayyad U M, Irani K B. Multi-interval discretization of continuous-valued attributes for classification learning[C]. Proc of 13th Int Joint Conf on Artificial Intelligence. Chambery: Morgan Kuafman, 1993: 1022-1027.

[3] Kerber R. ChiMerge: Discretization of numeric attributes[C]. Proc of 10th Int Conf on Artificial Intelligence. California: AAAI Press, 1992: 123-128.

[4] Liu H, Setiono R. Feature selection via discretization[J]. IEEE Trans on Knowledge and Data Engineering, 1997, 9(4): 642-645.

[5] Richeldi M, Rossotto M. Class-driven statistical discretization of continuous attributes[C]. Proc of 8th European Conf of Machine Learning. Berlin: Springer, 1995: 335-338.

[6] Kurgan L A, Cios K J. CAIM discretization algorithm[J]. IEEE Trans on Knowledge and Data Engineering, 2004, 16(2): 145-153.

[7] Ching J Y, Wong A K C, Chan K C C. Class-dependent discretization for inductive learning from continuous and mixed mode data[J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 1995, 17(7): 641-651.

[8] Tou J T, Gonzalez R C. Pattern recognition principles[M]. Massachusetts: Addison-Wesley, 1974.

[9] Nguyen H S, Skowron A. Quantization of real value attributes: Rough set and boolean reasoning approach[J]. Bullitin of Int Rough Set Society, 1997, 1(1): 5-16.

[10] Mehta S, Parthasarathy S, Yang H. Toward unsupervised correlation preserving discretization[J]. IEEE Trans on Knowledge and Data Engineering, 2005, 17(9): 1174-1185.

[11] Tsai C J, Lee C I, Yang W P. A discretization algorithm based on class-attribute contingency coefficient[J]. Information Sciences, 2008, 178(3): 714-731.

[12] Ruiz F J, Angulo C, Agell N. IDD: A supervised interval distance-based method for discretization[J]. IEEE Trans on Knowledge and Data Engineering, 2008, 20(9): 1230-1238.

[13] Blake C L, Merz C J. UCI repository of machine learning databases[EB/OL]. http://www.ics.uci.edu/mlearn/MLRepository.html.

[14] Demsar J. Statistical comparisons of classifiers over multiple data sets[J]. J of Machine Learning Research, 2006, 7(7): 1-30.