文章编号: 1001-0920 (2011) 10-1504-07

基于类-属性关联度的启发式离散化技术

周世昊[1,2], 倪衍森[1]

(1. 淡江大学 管理科学所，台北 10650；2. 醒吾技术学院，台北 224)

摘 要: 连续属性离散化在数据挖掘、机器学习和人工智能等领域起着重要的作用. 鉴于此, 提出一种基于类-属性关联度的启发式离散化技术. 该技术定义了一个新的离散化标准, 根据数据本身的特性选择最佳断点, 克服了目前最先进自顶向下离散化方法存在的缺陷. 基于粗糙集理论中变精度粗糙集模型, 提出一种新的不一致衡量标准, 能够有效地控制离散化所产生的信息丢失, 允许数据存在适当的分类错误度. 实验结果和统计性分析表明, 所提出的技术显著地提高了 J4.8 决策树和 SVM 分类器的学习精度.

关键词: 离散化；数据挖掘；自顶向下；变精度粗糙集；不一致

中图分类号: TP18 **文献标识码:** A

Heuristic discretization technique based on the class-attribute interdependence

ZHOU Shi-hao[1,2], *NI Yan-sen*[1]

(1. Graduate Institute of Management Sciences，Tamkang University，Taipei 10650，China；2. Hsing Wu College, Taipei 224，China．Correspondent：ZHOU Shi-hao，E-mail：0610@twse.com.tw)

Abstract: Discretization algorithms play an important role in many areas such as data mining, machine learning and artificial intelligence. Therefore, a heuristic discretization technique based on the class-attribute interdependence is proposed. A new discretization criterion is defined, which selects best cut points in terms of characteristics of the data itself and overcomes the existing deficiencies of state-of-the-art top-down discretization methods. A novel measure of inconsistency based on variable precision rough sets(VPRS) model is developed, which effectively controls information loss generated by discretization and allows an adaptive degree of misclassification. Empirical experiments and statistical analysis show that the proposed technique generates a better discretization scheme which significantly improves the accuracy of classification by running J4.8 and SVM.

Key words: discretization；data mining；top-down；variable precision rough sets；inconsistency

1 引 言

离散化方法描述的样本适用于较多数据挖掘、归纳学习等算法, 因此, 连续属性离散化是机器学习和数据挖掘研究和应用中的一个重要方面. 在规则提取、特征分类等算法中, 特别是应用粗糙集理论方法进行数据挖掘的研究和应用中, 连续 (实值) 属性必须进行离散化, 将连续属性值域转换成小数目有限的区间, 其目标是简化数据, 使得归纳算法更快、更准确地学习.

离散化算法的类型[1]可划分如下: 依据是否使用了类别标号, 分为有监督的和无监督的; 依据进行离散化的时间不同, 分为全局的和局部的; 依据是否考虑属性之间的区间进行合并还是分割, 分为自底向上的离散化和自顶向下的离散化. 比较有影响力的离散化方法有: 在有监督形式下的基于统计学思想的 Chi2 相关算法[2-5]和基于信息熵的相关算法[6-8]. 基于 Chi2 相关算法使用卡方统计来确定当前相邻区间是否被合并, 并采用显著性水平值逐渐降低的方法检验系统的不一致率, 确定离散化进程是否终止. 基于信息熵的相关算法依据类-属性之间的关联度程度提出了不同的离散化标准, 不断地选取能够使得类与属性之间的依赖程度最大的候选断点作为结果断点.

目前, 较多学者着重于新的离散化方法的产生, 例如 IDD[9]和数据离散化统一 (DU)[10]等. IDD 提出一种基于区间误差的离散化方法, 它定义了一种邻域

收稿日期: 2010-07-22；**修回日期:** 2010-11-20.

作者简介: 周世昊(1961−), 男, 讲师, 博士生, 从事人工智能、数据挖掘等研究；倪衍森(1961−), 男, 副教授, 博士生导师, 从事数据挖掘、财务工程等研究.

概念, 考虑类属性的取值顺序, 适用于类属性取值为连续的情况, 该方法既不属于 bottom-up 方法, 也不属于 top-down 方法. DU 是目前最先进的离散化技术之一, 它统一了存在的 6 种标准, 证明了基于统计独立性的离散化方法与基于信息理论的离散化方法是近似等价的, 提出了参数化的离散化标准, 6 种标准均可通过调整不同的参数值来获得, 并将离散化标准扩展到无限空间.

top-down 离散化方法[7-8,11-12]主要集中于单个连续属性的分类问题上, 没有考虑多变量下被离散数据所产生的信息丢失, 从而降低了学习精度; 此外, 最先进的类-属性关联度 top-down 离散化方法(CACC)[12]的离散化标准存在 3 个缺陷: 1) CACC 仍然存在 top-down 方法的不足; 2) CACC 没有将每个离散区间视为等同的重要性来对待; 3) CACC 忽视了每个区间中类分布位置对离散化进程的影响. 这些不足导致选取的断点是不精确的, 降低了离散化进程.

基于类-属性关联度的启发式离散化技术 (HDT) 定义了一种新的离散化标准, 根据数据本身的特性选择最合适的断点组合, 解决了 CACC 离散化标准的缺陷. 该技术基于粗糙集理论中变精度粗糙集模型, 提出一种新的不一致率衡量标准, 有效地控制了离散化过程中产生的信息丢失, 允许数据存在一定的分类错误, 在离散区间数与分类错误数之间达到了最佳平衡. 最后, 呈现出该技术的启发式算法, 并作出复杂性分析. 在 15 个 UCI 数据集[13]上的实验结果表明, 与存在的方法相比, 所提出的技术显著地提高了 J4.8 决策树[14]和 SVM 分类器[15]的学习精度.

2　HDT离散化技术

如前所述, 离散化方法的目标是得到更简化的数据, 且使得归纳算法学习得更快、更准确. Top-down 方法是目前最广泛、最有效的离散化形式之一, 但前面的研究主要集中在单个连续属性的分类问题上, 没有考虑在多变量下被离散的数据所产生的信息丢失, 从而降低了学习精度.

2.1　基于类-属性关联度的 top-down 方法的不足

对于给定数据集, 样本数为 N, 类别数为 S, A 为数据集中任意一个连续属性, 则存在如下离散化方案将连续属性的值域离散成 I 个区间:

$$P:\{[d_0,d_1],(d_1,d_2],\cdots,(d_{I-1},d_I]\}.$$

其中: d_0 是连续属性 A 的最小值, d_I 是 A 的最大值, 属性 A 的值按升序进行排列, $\{d_0,d_1,\cdots,d_I\}$ 为离散化过程中的断点集合. 属性 A 的每个值均可以划分到 I 个离散区间的某一个区间中, 且在离散化过程中, 区间中的信息会随着离散化方案的变化而不断发生变化. 离散化方案 P 作用在属性 A 上得到的区间和类别 S 构成二维信息表如表 1 所示. 在表 1 中, N_{ij} 为第 i 区间中第 j 类的样本数, $N_{i\cdot}$ 为第 i 区间的样本数, $N_{\cdot j}$ 为第 j 类的样本数, $i=1,2,\cdots,I$, $j=1,2,\cdots,S$.

表 1　二维信息表

区间	决策类				行求和
	Class 1	Class 2	⋯	Class S	
$[d_0,d_1]$	N_{11}	N_{12}	⋯	N_{1S}	$N_{1\cdot}$
$(d_1,d_2]$	N_{21}	N_{22}	⋯	N_{2S}	$N_{2\cdot}$
⋮	⋮	⋮	⋮	⋮	⋮
$(d_{I-1},d_I]$	N_{I1}	N_{I2}	⋯	N_{IS}	$N_{I\cdot}$
列求和	$N_{\cdot 1}$	$N_{\cdot 2}$	⋯	$N_{\cdot S}$	N (Total)

离散化方案 P 作用在属性 A 上得到的区间的 Shannon 熵[16]定义为

$$H(C,D|A)=\sum_{i=1}^{I}\sum_{j=1}^{S}p_{ij}\log_2(1/p_{ij}). \tag{1}$$

相似地, 对于两个随机变量 X 和 Y, 它们之间在某种程度上也是相互联系的, 即存在着一定的统计依赖关系, 互信息反映了这种依存关系的强弱. 离散化方案 P 作用在属性 A 上得到的区间和类标签的互信息定义为

$$\mathrm{MI}(C,D|A)=\sum_{i=1}^{I}\sum_{j=1}^{S}p_{ij}\log_2(p_{ij}/(p_{i\cdot}p_{\cdot j})). \tag{2}$$

其中: $p_{ij}=N_{ij}/N$ 为第 i 区间中第 j 类样本出现的概率, $p_{i\cdot}=N_{i\cdot}/N$ 为第 i 区间中样本出现的概率, $p_{\cdot j}=N_{\cdot j}/N$ 为第 j 类样本出现的概率.

Ching 等人[8]提出了基于信息理论的类-属性间相互依赖的连续属性离散化算法 (CADD), 其离散化标准 CAIR 定义为

$$\mathrm{CAIR}(C,D|A)=\frac{\mathrm{MI}(C,D|A)}{H(C,D|A)}. \tag{3}$$

CAIR 标准衡量了类与被离散属性之间的相互依赖程度, 其值越大, 类-属性间相互依赖越大. 然而, CADD 算法存在如下不足: 需要用户规定区间数目; 要求对置信区间的选择进行训练; 用最大熵方法对离散化区间初始化, 但这样的初始化可能会导致很糟的起始断点; CAIR 标准容易导致离散化程度过低而不适度.

对于以上不足, Kurgan 等人[11]在离散化标准上进行了改进, 提出了 CAIM 离散化算法, 该算法的离散化标准为

$$\mathrm{CAIM}(C,D|A)=\sum_{i=1}^{I}\frac{\max_i^2}{N_{i\cdot}}\Big/I. \tag{4}$$

其中: I 为当前离散的区间数, $\max_i$ 为第 i 区间中最大类的样本数. 该标准试图使类与属性之间的相互依赖程度最大化, 区间数最小化. 但它仅考虑了区间中最多的类与属性间的相互依赖, 这样会使得离散化过度

而导致结果不精确.

文献[12]针对上述不足, 提出了CACC离散化标准为

$$\mathrm{CACC}=\sqrt{x/(x+N)},\tag{5}$$

其中

$$x=N\Big[\Big(\sum_{i=1}^{I}\sum_{j=1}^{S}\frac{N_{ij}^2}{N_{i\cdot}N_{\cdot j}}\Big)-1\Big]\Big/\log I.$$

与CAIM离散化算法相比, CACC考虑了数据整体的分布情况. 此外, CACC标准中的除法运算有两个目的: 一是为了加速离散化进程; 二是为了防止连续属性被离散化过度, 因为在CAIM离散化标准中, 除以I很容易导致离散化过度, 所以将I改为$\log I$.

虽然CACC对前面的离散化方法作了改进, 但仍存在以下几点不足:

1) CACC仍然存在CAIR标准的不足. 假设存在$U=u/(u+N)$和$V=v/(v+N)$, 其中$u>v$, 则一定有$U>V$. 由于算法的目标是寻找具有最大CACC值的断点, 在CACC标准中, $x/(x+N)$可以由x替代, 且将x代入式(5)后, N是多余的. 这样, CACC离散化标准可以由$\Big(\sum_{i=1}^{I}\sum_{j=1}^{S}N_{ij}\log_2\Big(\frac{N_{ij}}{N_{i\cdot}N_{\cdot j}}\Big)\Big)\Big/\log I$表示. 此时, CACC标准中的$\sum_{i=1}^{I}\sum_{j=1}^{S}N_{ij}\log_2\Big(\frac{N_{ij}}{N_{i\cdot}N_{\cdot j}}\Big)$与CAIR标准

$$\mathrm{CAIR}(C,D|A)=\frac{\sum_{i=1}^{I}\sum_{j=1}^{S}\frac{N_{ij}}{N}\log_2\frac{NN_{ij}}{N_{i\cdot}N_{\cdot j}}}{\sum_{i=1}^{I}\sum_{j=1}^{S}\frac{N_{ij}}{N}\log_2\frac{N}{N_{ij}}}$$

在离散化效果上是相近的, 这样CACC仍然存在CAIR的不足. 另外, 尽管CACC除以$\log I$的目的是控制连续属性被离散化过度, 但对于候选断点而言, $\log I$与$\log(I+1)$仍然存在较大差异, 创建了较少的区间数, 使得数据丢失了较多的有效信息.

2) CACC没有将每个离散区间视为等同的重要性对待. 在CACC离散化标准中, 若添加断点后的离散区间与类之间的CACC值越大, 则该断点越好. 好的断点的实质是尽量将具有不同类的样本点分隔开, 相同类的样本点分到同一区间, 以减少分类错误. 因此, 在与当前断点相邻的区间中, 若相同类的样本数相差较大, 则每个区间中同一类的CACC值应该给予等同的待遇, 即相等的CACC值. 然而, $\sum_{i=1}^{I}\sum_{j=1}^{S}N_{ij}\log_2\Big(\frac{N_{ij}}{N_{i\cdot}N_{\cdot j}}\Big)$仅呈现出区间中含有较多类样本点的CACC值较大, 较少类样本点的CACC值较小, 没有给予公平的对待.

通过表2中的例子作简单说明. 表2中有15个样本, 其中目标类1与目标类2均可看作决策类, 本文仅将目标类1看作决策类. 如果离散属性为“年龄”, 则第1个选择的断点是$(13+19)/2=16$. 一般地, 断点为相邻两个属性值的平均值, 因此, 样本1∼5为一个区间, 6∼15为另一区间. 在分开的两个区间中, 一个包含5个“学生”类, 另一个包含1个“学生”类. 如果CACC标准被应用, 则区间1∼5中“学生”类的CACC值比较大, 而区间6∼15中“学生”类的CACC值比较小. 事实上, 该断点已经较好地将“学生”类分到同一区间, 两区间的“学生”类相差较大, 说明选取了好的断点, 两区间中“学生”类的CACC值应给予等同的待遇.

表 2 决策表

样本	年龄	其他属性	目标类1	目标类2
1	5	···	学生	学生
2	7	···	学生	学生
3	8	···	学生	学生
4	11	···	学生	学生
5	13	···	学生	学生
6	19	···	医生	医生
7	23	···	军人	军人
8	27	···	军人	学生
9	29	···	学生	军人
10	33	···	商人	教师
11	36	···	商人	教师
12	37	···	教师	商人
13	41	···	教师	教师
14	45	···	教师	商人
15	50	···	教师	教师

3) CACC忽视了每个区间中类分布位置对于离散化进程的影响. 一般而言, 寻找的断点应尽量使得分隔后的区间中对应的相同类更紧凑, 相同类的样本点应紧密地接近. 然而, CACC没有考虑类分布顺序的影响. 假设两个区间每个类的样本数分别对应相等, 但类分布的位置不同, 若CACC标准被采用, 则两个区间的CACC值相同, 此时体现不出两个区间的优劣. 仍然通过表2的例子来说明. 对于样本区间6∼15, 目标类1和目标类2的每个类的样本数对应相等, 但是目标类1相同类的样本点更紧凑, 目标类2中每个类的分布不均匀, 因此, 目标类1的分布要优于目标类2, 即目标类1的CACC值应大于目标类2的CACC值. 然而, CACC离散化标准计算出二者的CACC值是相同的, 没有区分出两区间的优劣.

2.2 新的离散化标准NDC

离散化标准直接影响着选择断点的好坏, 进而影响着学习精度. 在深入地分析了第2.1节中类-属性相互依赖的top-down离散化方法存在缺陷的基础上, 提出一种新的启发式离散化标准NDC, 能够根据数据

本身的特性选择好的断点, 克服了 top-down 方法的不足. NDC 定义为

$$\mathrm{NDC}=\frac{\sum_{i=1}^{I}\sum_{j=1}^{S}\left(N_{ij}-\frac{N_{\cdot j}^{i-1,i,i+1}}{2}\right)^{\alpha}}{N_{i\cdot}(1+\mathrm{std}(X_{ij}))}. \tag{6}$$

其中: $N_{\cdot j}^{i-1,i,i+1}$ 为第 i 区间以及与其相邻的 $i-1$ 和 $i+1$ 这 3 个区间中第 j 类的样本数; $X_{ij}=\{l_{ij1},l_{ij2},\cdots,l_{ijr}\}$ 为第 i 区间中第 j 类簇位置标号的集合 (连续相同的类视为一个类簇); r 为第 i 区间中第 j 类簇的位置标号的个数, 即 X_{ij} 的基数; l_{ijk} 为第 i 区间中第 j 类簇在第 $k(1\leqslant k\leqslant r)$ 位置标号上, l_{ijk} 取自然数 $1,2,\cdots$, 注意到, 这里的位置标号指区间中类簇所在的位置, 第 1 次出现类簇的位置标号为 $1,2,\cdots$; α 为可调实参数, 一般取 $\alpha=2$, 在实验中可以通过调整 α 值来选取好的学习精度; $\mathrm{std}(X_{ij})$ 为 X_{ij} 的标准差, 具体展开如下:

$$\mathrm{std}(X_{ij})=\sqrt{\sum_{k=1}^{r}(l_{ijk}-\mathrm{mean}(X_{ij}))^2/r}, \tag{7}$$

$$\mathrm{mean}(X_{ij})=\sum_{k=1}^{r}l_{ijk}/r, \tag{8}$$

其中 $\mathrm{mean}(X_{ij})$ 为 X_{ij} 的均值.

下面解释 NDC 中每个组件的功能. 从 NDC 标准中可以看出, $N_{ij}-N_{\cdot j}^{i-1,i,i+1}/2$ 等同了每个区间的评价重要性, 即每个区间中该类的 CACC 值给予了等同的待遇, 具有相同的 CACC 值. $\mathrm{std}(X_{ij})$ 是 X_{ij} 的标准差, 体现了类簇的位置标号的密集程度, 若 $\mathrm{std}(X_{ij})=0$, 则说明第 i 区间第 j 类全部紧凑在一起, 仅含有一个类簇, 没有被其他样本点隔开, 这是最佳的分割. 若 $\mathrm{std}(X_{ij})$ 很大, 则说明区间含有较多分散的类簇, 即 $\mathrm{std}(X_{ij})$ 越大, 类簇越分散, 被分割的区间越不好; $\mathrm{std}(X_{ij})$ 越小, 类簇越紧密, 被分割的区间就越好. 利用 $1+\mathrm{std}(X_{ij})$ 是为了避免 NDC 标准的分母出现 0 的情况. 综上分析, NDC 值越大, 选择的断点越好. 因此, 在离散化过程中不断地最大化 NDC 值来选择最佳断点. 该离散化标准能够根据数据当前的状态选择较好的断点.

以表 2 为例作进一步解释, 假设当前属性“年龄”有两个区间, 样本 1~5 为第 1 个区间, 6~15 为第 2 个区间, 且虚拟出第 0 个和第 3 个区间. 此时, 对于第 1 个和第 2 个区间的“学生”类而言, 第 1 个区间得到的 NDC 标准中分子的值为

$$(N_{1,\text{学生}}-N_{\cdot\text{学生}}^{0,1,2}/2)^2=(5-6/2)^2=4.$$

第 2 区间分子的值为

$$(N_{2,\text{学生}}-N_{\cdot\text{学生}}^{1,2,3}/2)^2=(1-6/2)^2=4.$$

由于没有第 0 个和第 3 个区间, 这两个区间的“学生”类的样本数为 0. 以上分析说明了各区间中相同类的 CACC 评价具有公平的对待. 分析 NDC 标准中分母的计算功能: 对于“年龄”属性的第 2 个区间, 目标类以“教师”为例. 对于目标类 1, $X_{2,\text{教师}}=\{1\}$, $\mathrm{std}(X_{2,\text{教师}})=0$; 对于目标类 2, $X_{2,\text{教师}}=\{1,3,5\}$, $\mathrm{std}(X_{2,\text{教师}})=2$. 这说明尽管两个区间对应的每个类的样本数相同, 但对应类分布的位置不同, 这样, 目标类 1 的 CACC 值要大于目标类 2 的 CACC 值, 表明 $\mathrm{std}(X_{ij})$ 充分反映了各区间中类分布位置对离散化进程的影响.

2.3　新的不一致衡量标准 NIC

top-down 离散化技术主要集中在单个连续属性的分类问题上, 没有考虑在多变量下被离散的数据产生的信息丢失, 从而降低了学习精度. 基于粗糙集理论中变精度粗糙集模型[17], 提出一种新的不一致率衡量标准 NIC, 有效地控制了离散化过程中产生的信息丢失, 允许数据存在一定的分类错误, 在离散区间数与分类错误数之间达到最佳的平衡.

设 $S=\{U,A,V,F\}$ 为一信息系统, 其中 $U=\{x_1,x_2,\cdots,x_n\}$ 是论域, A 是属性集合, V 是属性取值集合, F 是 $U\times A\to V$ 的映射. 若 $A=C\bigcup D$, $C\bigcap D=\varnothing$, C 称为条件属性集, D 称为决策属性集, 则该信息系统称为决策表.

定义 1　$x,y\in U$, 对于 $P\subseteq A$, θ_P 是 U 上的一个等价关系, 如果满足

$$x\theta_P y\Leftrightarrow(\forall p\in P)(f_p(x)=f_p(y)),$$

则称 θ_P 是 x,y 的一个不可分辨关系.

定义 2　设 $X\subseteq U$ 为论域的一个子集, $P\subseteq C$, 集合 X 关于 P 的 β- 下近似为

$$\underline{C_\beta}(D)=\bigcup_{1-CP(Y|x_i)\leqslant\beta}\{x_i\in[x]_P\}. \tag{9}$$

其中: $[x]_P$ 为 U 中在等价关系 P 下的等价类元素构成的集合, $CP(Y|x_i)=|Y\bigcap x_i|/|x_i|$, $Y\subseteq[x]_D$, $|\cdot|$ 为集合的基数; $\underline{C_\beta}(D)$ 为样本点关于决策属性的正确分类程度.

定义 3　设 $S=\{U,A,V,F\}$ 是一决策表, 条件属于 C 的 β- 近似精度定义为

$$\gamma_\beta=|\underline{C_\beta}(D)|/|U|, \tag{10}$$

其中 γ_β 为正确分类率, 允许 $1-\beta$ 分类误差. NIC 不一致率衡量标准定义为

$$\mathrm{NIC}=1-\gamma_\beta. \tag{11}$$

NIC 标准反映了被离散数据所产生的不一致率, 即信息丢失. 根据数据本身的特点, 设定合适的 β 值, 允许数据存在 $1-\beta$ 的分类错误而不降低学习精度, 使得数据更加简化.

2.4 算法描述 (HDA)

提出HDT技术的启发式算法HDA, 该算法分两个阶段对连续属性进行离散化以获得最佳的离散化结果. 在给出算法之前, 首先引出定义4.

定义 4 设 $S=\{U,A,V,F\}$ 为决策表, 条件属性子集 $B\subset C$, 依赖于决策属性集合 D 的任意条件属性 $a\in C$, 相对于条件属性集合 B 的属性重要度定义为

$$\mathrm{sig}(a,B,D)=\gamma_{B+\{a\}}-\gamma_B, \tag{12}$$

其中 γ_B 为原始粗糙集的近似精度[18], 此时 $\beta=0$.

连续属性离散化方法是否合理决定着对信息的表达和提取的准确性. 在信息系统中, 重要的属性对决策划分的影响大, 相对于决策属性而言也比较重要. 如果先离散化了重要的属性, 则会影响其他属性的合并, 信息系统也会过早的出现不一致, 所以在对每个属性进行离散化时, 希望先合并不重要的属性, 这样对其他属性不会产生影响, 可以得到更好的离散效果.

HDA 算法描述如下:

Step 1: 计算每个属性的重要性 (定义 4), 并按属性重要性由小到大的顺序排序 $A_1,A_2,\cdots,A_m$(A_1 代表重要性最小的属性, A_m 代表重要性最大的属性).

Step 2: 对每个属性 $A_i(i=1,2,\cdots,m)$ 作循环操作: 将候选断点值从小到大排序 $\{d_0,d_1,\cdots,d_I\}$; 置 global NDC $=0$; 计算候选断点集合中每个断点的 NDC 值, 并找出 NDC 值最大的断点; 若 $\mathrm{NDC}_{\max}>$ global NDC, 则 global NDC $=\mathrm{NDC}_{\max}$, 将该断点从候选断点集合中去掉, 加入到离散断点集合中; 否则退出循环.

Step 3: Do{对每一个属性 $A_i(i=1,2,\cdots,m)$ 作循环操作: 不断地选出 NDC 值最大的断点加入到离散断点集合中; 若选中当前断点后的不一致率 NIC $\leqslant\xi$ (ξ 为预先定义的可容忍的最小信息丢失率), 则去掉该断点; 继续执行下一个属性 } While (产生不一致率 NIC $>\xi$).

下面分析 HDA 算法的时间复杂度. 在算法的第 1 阶段, 计算属性重要性需要 $O(N^2M)$, 其中 M 为条件属性个数. 由于 M 通常较小, 可将其看作常量, 计算属性重要性的时间复杂度为 $O(N^2)$. HDA 对候选断点进行排序的时间复杂度是 $O(n\log n)$, n 为候选断点个数; 对 NDC 值进行计算的复杂度为 $O(nS)$, S 可以看作常量, 所以算法计算 NDC 值并确定断点阶段总的时间复杂度为 $O(n\log n)$. 在算法的第 2 阶段, 根据离散后产生的不一致率来判断是否应继续添加断点时, 算法复杂度为 $O(M)O(n^2)=O(n^2)$. 综上所述, 该算法总的时间复杂度为 $O(N^2)+O(n\log n)+O(n^2)=O(N^2)$.

3 性能评价

3.1 实验的建立与结果

为了评价 HDT 的性能, 采用 UCI 机器学习数据库中的 15 个数据集, 见表 3. 该数据集是数据挖掘等实验所常用的数据, 多来自于智能控制、医疗、科学等领域. 将所提出的 HDT 方法与下列 5 种方法进行比较:

1) CAIM —— 著名的基于类-属性相互依赖的 top-down 离散化方法[11];

2) CACC —— 最先进的类-属性关联系数的 top-down离散化方法[12];

3) Ext-Chi2 —— 最先进的 bottom-up 离散化方法[5];

4) Entropy-based —— 基于信息熵的最小描述长度 MDLP 离散化方法[7];

5) EQF —— 一种传统的无监督离散化方法[1].

表 3 数据信息表

数据集	连续属性	离散属性	类别数	样本数
Iris	4	0	3	150
Auto	5	2	3	392
Breast	9	0	2	683
Ionosphere	32	2	2	351
Pima	8	0	2	768
Glass	9	0	7	214
Wine	13	0	3	178
Machine	7	0	8	209
Heart	6	7	2	296
Sonar	60	0	2	208
Vehicle	18	0	4	846
Vowel	10	3	6	990
Bupa	6	0	2	345
Artificial	6	1	10	5 109
Page-blocks	10	0	5	5 473

15 个数据集全部通过上述离散化方法进行离散化, 在 VC++6.0 环境下实现. 将离散后的结果使用 Weka 数据挖掘工具[19]进行分类预测, 并将离散后的数据应用 J4.8 方法构造决策树, 采用 10 折交叉验证的方法[20]对平均学习精度统计进行对比, 结果见表 4. 同时, 使用 SVM 对离散数据用“一对多 (1-V-r)”方法进行分类[15], 仍然采用 10 折交叉验证的方法对平均学习精度统计进行对比, 结果见表 5. 表 4 和表 5 中, 加黑数据项为分类器识别的最高精度. SVM 模型类型选为 C-SVC, 核函数类型选为 RBF 函数, 惩罚因子搜索范围为 [1,100], 核函数参数 γ 搜索范围为 [0.05,0.5]. 由于核函数依赖于输入样本向量的内积, 大的属性值容易导致计算复杂, 训练时间较长. 为了避免上述情况发生, 将属性值进行归一化, 即

$$\overline{x}_i=2\frac{x_i-\min(x_i)}{\max(x_i)-\min(x_i)}-1. \tag{13}$$

表 4　J4.8 分类预测结果

数据集	离散化方法					
	HDT	CAIM	CACC	Ext-Chi2	Entropy-based	Equal-F
Iris	**94.9**	92.6	92.6	94.2	93.3	92.6
Auto	**83.4**	75.8	78.2	81.4	79.8	72.5
Breast	96.1	**96.7**	**96.7**	96.3	96.0	93.8
Ionosphere	91.7	92.0	90.8	**92.9**	86.8	82.6
Pima	**78.9**	73.4	74.6	75.8	70.3	66.2
Glass	76.9	76.5	**78.6**	70.6	73.2	57.9
Wine	**96.2**	92.3	92.3	94.1	83.3	90.7
Machine	**89.7**	81.8	83.3	85.6	80.4	72.0
Heart	81.9	74.7	76.8	73.2	74.8	69.6
Sonar	**63.8**	60.0	58.7	59.5	59.9	51.6
Vehicle	**72.9**	67.8	67.8	70.9	67.7	65.4
Vowel	97.0	97.1	97.6	**97.8**	95.0	95.0
Bupa	**69.8**	64.9	62.3	65.2	61.2	57.8
Artificial	**62.4**	59.2	60.5	56.9	59.7	56.1
Page-blocks	**96.7**	95.4	95.6	95.6	95.1	94.8
平均级别	**1.60**	3.43	3.07	2.70	4.37	5.83

表 5　SVM分类预测结果

数据集	离散化方法					
	HDT	CAIM	CACC	Ext-Chi2	Entropy-based	Equal-F
Iris	**96.1**	92.6	92.6	95.3	94.0	91.3
Auto	79.8	78.0	**80.2**	75.2	76.2	73.2
Breast	**97.6**	97.0	97.0	96.9	95.0	95.0
Ionosphere	**94.7**	92.8	91.2	93.4	93.5	93.4
Pima	**72.4**	69.5	66.3	67.0	69.4	69.1
Glass	**75.9**	68.2	67.9	70.5	56.7	69.6
Wine	92.1	**94.9**	**94.9**	**94.9**	69.9	94.4
Machine	78.9	77.0	79.6	78.4	**80.8**	72.2
Heart	75.8	76.3	**77.9**	68.5	73.6	69.2
Sonar	**89.2**	86.7	86.1	87.2	59.0	57.4
Vehicle	**69.5**	65.7	65.7	66.3	65.4	68.2
Vowel	**96.2**	94.1	93.8	93.4	93.6	91.5
Bupa	**67.9**	62.0	63.7	60.5	63.7	62.6
Artificial	**61.8**	56.4	58.5	58.9	59.6	56.6
Page-blocks	**96.2**	92.2	92.0	91.8	93.3	88.8
平均级别	**1.60**	3.56	3.47	3.77	3.67	4.80

归一化后的属性值 $x_i \in [-1,+1]$. 在HDT技术中, 选取参数 $\alpha=2$, $\beta=0.1$. 除了Artificial和Page-blocks数据集, 其余数据集 ξ 取值均为0.01. 由于Artificial和Page-blocks为不一致数据, 根据定义2($\beta=0$), 计算出 ξ 取值为两数据集初始的不一致率 $\xi_{\text{Artificial}}=0.013$, $\xi_{\text{Page-blocks}}=0.008$. 由于空间有限, 将在以后的工作中调整 β 和 ξ 值以获得更好的学习精度.

从表4和表5可以明显看出, HDT技术的平均分类学习精度最高, 最后1行为15个数据集的每个离散化算法的平均级别, 如果算法的性能最佳, 则平均级别为1, 以此类推.

3.2　统计性分析

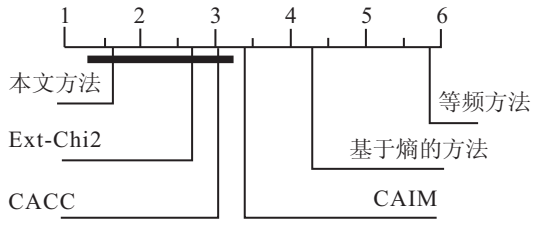
为了衡量HDT的有效性, 采用Friedman统计[21]来测试所有离散化方法是否有显著的差异, 如果存在显著差异, 则采用Holm's post-hoc统计测试, 目的是进一步衡量HDT技术与其他方法相比是否存在统计意义上的显著差异, 统计中用到的显著水平 $\alpha=0.05$. Friedman统计表示如下:

$$\chi_F^2=\frac{12J}{T(T+1)}\Big[\sum_j Q_j^2-\frac{T(T+1)^2}{4}\Big]. \tag{14}$$

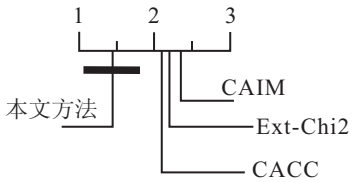
其中: J 为数据集的个数; T 为离散化算法的个数; $Q_j=\dfrac{1}{J}\sum\limits_i v_i^j$, v_i^j 为第 i 个数据集第 j 个算法的平均级别. 表4的Friedman统计值为

$$\chi_F^2=\frac{12\times 15}{6\times 7}\Big[1.6^2+3.43^2+3.07^2+2.7^2+4.37^2+5.83^2-\frac{6\times 7^2}{4}\Big]=45.538\,3.$$

计算结果大于阈值11.1, 说明6种离散化方法存在差异, 这样将采用Holm's post-hoc统计测试, 得到图1(a), 轴下的粗线表示性能相当区域的方法与其他方法有显著的差异, 即HDT, Ext-Chi2和CACC性能相当. 如果去掉Entropy-based和Equal-F方法, 则得到图1(b), 表明HDT显著优于其他3种方法. 基于表5, 通过计算得到Friedman统计值为19.269 9, 大于阈值11.1, 因此离散化方法也存在差异, Holm's post- hoc统计测试将被采用, 并得到图2. 图2表明, HDT显著优于其他所有方法.



(a)　6种方法比较



(b)　4种方法比较

图 1　采用 Friedman 统计方法的 C4.5 性能比较

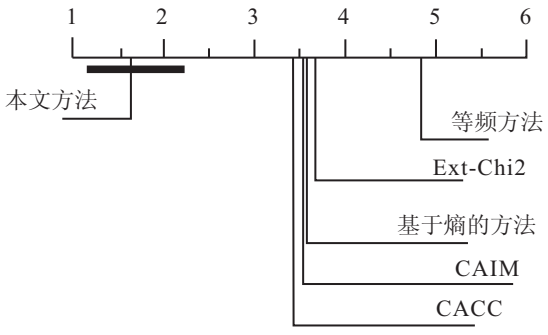


图 2　采用 Friedman 统计方法的 SVM 性能比较

4　结　论

连续属性离散化在数据挖掘、机器学习和人工

智能等领域具有重要的作用. 先前的top-down离散化方法主要集中在单个连续属性的分类问题上, 没有考虑在多变量下被离散的数据所产生的信息丢失, 从而降低了学习精度. 而且最先进的基于类-属性相互依赖的top-down离散化方法存在一定缺陷, 导致选取到不精确的断点, 降低了离散化进程. 鉴于此, 提出一种基于类-属性关联度的启发式离散化技术, 该技术定义了一个新的离散化标准, 根据数据本身的特性选择最合适的断点组合, 解决了CACC的缺陷. 此外, HDT基于粗糙集理论中变精度粗糙集模型, 提出一种新的不一致率衡量标准, 有效地控制了离散化过程中产生的信息丢失, 允许数据存在一定的分类错误. 在15个UCI数据集上的实验结果表明, 与存在的方法相比, HDT技术显著地提高了J4.8决策树和SVM分类器的学习精度.

参考文献(References)

[1] Dougherty J, Kohavi R, Sahami M. Supervised and unsupervised discretization of continuous feature[C]. Proc 12th Int Conf on Machine Learning. 1995: 194-202.

[2] Kerber R. ChiMerge: Discretization of numeric attributes[C]. Proc 9th National Conf on Artificial Intelligence. AAAI Press, 1992: 123-128.

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

[4] Tay E H, Shen L. A modified Chi2 algorithm for discretization[J]. IEEE Trans on Knowledge and Data Engineering, 2002, 14(3): 666-670.

[5] Su C T, Hsu J H. An extended Chi2 algorithm for discretization of real value attributes[J]. IEEE Trans on Knowledge and Data Engineering, 2005, 17(3): 437-441.

[6] 谢宏, 程浩忠, 牛东晓. 基于信息熵的粗糙集连续属性离散化算法[J]. 计算机学报, 2005, 28(9): 1570-1574.
(Xie H, Cheng H Z, Niu D X. Discretization of continuous attributes in rough set theory based on information entropy[J]. Chinese J of Computers, 2005, 28(9): 1570-1574.)

[7] Fayyad U, Irani K. Multi-interval discretization of continuous-valued attributes for classification learning[C]. Proc 13th Int Joint Conf on Artificial Intelligence. San Mateo: Morgan Kaufmann, 1993: 1022-1027.

[8] 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.

[9] 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.

[10] Jin R M, Breitbart Y, Muoh C. Data discretization unification[J]. Knowledge and Information System, 2008: 115-142.

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

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

[13] Hettich S, Bay S D. The UCI KDD archive[DB/OL]. 1999. http://kdd.ics.uci.edu/.

[14] Witten I H. Data mining: Practical machine learning tools and techniques with java implementations[M]. San Francisco: Morgan Kaufmann, 2000.

[15] Hsu C W, Lin C J. A comparison of methods for multiclass support vector machines[J]. IEEE Trans on Neural Networks, 2002, 13(2): 415-425.

[16] 孟庆生. 信息论[M]. 西安: 西安交通大学出版社, 1986: 18-30.
(Meng Q S. Information theory[M]. Xi'an: Xian Jiaotong University Press, 1986: 18-30.)

[17] Ziarko W. Variable precision rough set model[J]. J Computer and System Science, 1993, 46: 39-59.

[18] Pawlak Z. Rough sets[J]. Int J of Computer and Information Sciences, 1982, 11(5): 341-356.

[19] Weka 3 Data mining software in Java[DB/OL]. http://www.cs.waikato.ac.nz/ml/weka, 2007.

[20] Weiss S M, Kulikowski C A. Computer systems that learn: Classification and prediction methods from statistics, neural nets[C]. Machine Learning and Expert Systems. San Mateo: Morgan Kaufmann, 1990.

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