文章编号: 1001-0920(2002)05-0545-05

一种新的用于连续值属性离散化的约简算法

刘震宇, 郭宝龙, 杨林耀
(西安电子科技大学 测控工程系, 陕西 西安 710071)

摘 要: 针对在 Nguyen 和 Skowron 的离散化算法中进行启发式约简时会出现某些属性不能进行离散化问题,以及在无核数据集中启发式约简算法计算量比较大等问题,在粗糙集理论和属性频率函数的基础上给出一个新概念——候选核,并提出一种新的用于连续值属性离散化的约简算法——基于候选核的启发式约简算法(简称 BCC)。该算法可以寻找到能对所有属性进行离散化的约简。实验表明,所提出的 BCC 算法能提高大数据集的离散化效果。

关键词: 数据挖掘; 粗糙集理论; 离散化; 约简算法
中图分类号: TP 18 文献标识码: A

A new reduction algorithm for discretization of continuous features

LIU Zhen-yu, GUO Bao-long, YANG Lin-yao
(Department of Measurement Control Engineering, Xidian University, Xi'an 710071, China)

Abstract: There are two problems in the traditional discretization algorithm when heuristic reduction algorithms are used to find the reduction. One is that the reduction discretizing all attributes may not be found. The other is that the heuristic reduction algorithm needs a great deal of time to get the reduction in the data sets without core. To solve the two problems, a new concept called candidate core is given, which is built on the rough set theory and attribute frequency function, and a new heuristic reduction algorithm based on candidate core (named BCC) is presented. This new heuristic reduction algorithm of BCC can find the reduction of the data sets, which discretizes all attributes. The results of experiments show that the new algorithm can improve the performance of discretization for large data sets.

Key words: data mining; rough set theory; discretization; reduction algorithm

1 引 言

连续值属性离散化在机器学习、知识发现和决策等领域有着重要的应用,其中用粗糙集理论研究离散化问题引起了人们的关注[1,2]。20 世纪 90 年代中期, Nguyen 和 Skowron 在粗糙集理论和布尔推理的基础上,提出一种全局有监督的离散化算法[2],简称 NS 算法。NS 算法最重要的意义在于能将离散化问题转变成寻找初始区间约简的问题。但在实际应用中,当 NS 算法面对较多的初始区间时,会出现某些属性不能进行离散化问题,以及获得所

收稿日期: 2001-08-13; 修回日期: 2001-10-16
基金项目: 国家自然科学基金项目(69975015)
作者简介: 刘震宇(1976—),男,湖南宁乡人,硕士生,从事粗糙集、数据仓库和数据挖掘研究; 郭宝龙(1962—),男,陕西西安人,教授,博士生导师,博士,从事神经网络、模式识别和人工智能等研究。

有初始区间的约简需要大量时间等问题。目前,在 NS 算法原理下,较新的基于属性频率函数的启发式约简算法[3](BFF 算法)等能减少约简时间,但不能保证找到对所有属性离散化的初始区间的约简,而且现有算法在无核数据集中计算量很大。

本文通过对 NS 算法存在的问题和 BFF 算法进行分析,提出了候选核的概念,给出一种新的基于候选核的启发式约简算法(简称 BCC 算法)。该算法能很好地解决 NS 算法中约简过程存在的问题,并通过实验验证了 BCC 算法的有效性。

2 NS 离散化算法

2.1 基本概念

定义 **1**(信息系统)[4] 信息系统是一个 4 元组 $S=(U,A,V,f)$。其中,U 是一个非空有限集,称为全域;A 是属性的非空有限集,$A=P\cup Q$,$P\subset A$ 为条件属性集合,$Q\subset A$ 为决策属性集合,都不为空;V 是属性的值域,$V=\cup V_i$,V_i 是各个属性的值域;f 是信息函数,与每条记录及每个属性有关。

定义 **2**(初始区间)[2] 若 S 中有 m 个连续值属性,对于第 i 个连续值属性 $c^i\in P$,值域为 V_i,S 中 c^i 的值集合是 $c^i(U)=\{c^i_1,c^i_2,\cdots,c^i_n\}$,$n$ 为 c^i 在 U 中值的个数。属性 c^i 的初始区间 p^i_k 为 $[c^i_k,c^i_{k+1}]$,$0<k<n$,属性 c^i 的初始区间集合 $p^i=\bigcup_{k=1}^{n-1}p^i_k$,$S$ 的初始区间的集合 $P(S)=\bigcup_{i=1}^{m}p^i$。

2.2 NS 离散化算法基本原理[2]

依据上述定义,首先从 S 中找出各属性的初始区间 p^i,然后根据记录间决策值的不同,得出一个类似于区分矩阵的初始区间矩阵 Ψ;其次将矩阵中所有项表示成合取范式形式,从而构造出区分函数,通过布尔推理获得约简;然后选择一个约简,计算约简中属性 c^i 的初始区间 p^i_k 的中值点为 $0.5\times(c^i_k+c^i_{k+1})$,这些中值点就是各属性的离散点;最后利用离散点对 S 进行离散化。

2.3 NS 离散化算法存在的问题

首先,该算法对区分函数的化简是一个 NP 问题[3]。随着连续值属性的个数、各个属性中初始区间的个数以及记录数的增加,运算量会迅速增加。要从由这么多项的初始区间构成的区分函数中找出所有的约简项,需要花费大量的时间。其次,Nguyen 和 Skowron[4] 指出,可以用启发式约简寻找初始区间的次优约简,同时给出了一些启发式约简算法。但在约简项中有时会出现缺少某个属性的初始区间,使得一些属性不能进行离散化,影响了离散化效果。这是该算法存在的主要问题之一。

3 基于候选核的约简算法

3.1 约 简

NS 算法通过初始区间的概念把离散化问题转变成约简问题,使得在粗糙集中研究离散化问题成为可能。

定义 **3**(约简)[3,5] 若条件属性集合 $\mathrm{Redu}(P,Q)$ 满足以下两个条件:1) $\gamma(P,Q)=\gamma(\mathrm{Redu}(P,Q),Q)$;2) 对于 $\mathrm{Redu}(P,Q)$ 的任意子集,条件 1) 不成立。则 $\mathrm{Redu}(P,Q)$ 可称为信息系统 S 的一个约简,简记为 Redu。其中 $\gamma(P,Q)$ 称为依赖度,是指决策属性 Q 以 $\gamma(P,Q)$ 依赖于条件属性 P,可由下式求出

$$\gamma(P,Q)=\mathrm{card}(\mathrm{Pos}(P,Q))/\mathrm{card}(U) \tag{1}$$

$$\mathrm{Pos}(P,Q)=\bigcup_{Y\in Q^*}\underline{\mathrm{IND}}(P)Y \tag{2}$$

其中,$\mathrm{card}(\cdot)$ 为集合的基数,Q^* 为决策属性的决策类。

约简是原信息系统中属性集合的一个子集,而且该子集与整个属性集合的分类能力相同。信息系统的所有约简可以通过构造区分矩阵,并简化由区分矩阵构造的区分函数得到,但获得所有约简和最小约简都已被证明是 NP-Hard 问题,因此运用启发信息进行约简是必要的。

3.2 属性频率函数和 BFF 算法

文献[3]给出一种新的基于属性频率函数的启发式约简算法,即 BFF 算法。它能在 NS 算法框架下对约简部分作一些改进,使算法适合于寻找最小约简或次优约简,且算法有较快的速度。BFF 算法中的属性频率函数是受以下两个思想的启发而提出的:

1) 属性在区分矩阵中出现次数越多,属性的重要性越大;

2) 属性所在的区分矩阵项越短,属性的重要性越大。

属性频率函数定义为

$$f(a)=f(a)+|P|/|\Psi(x,y)| \tag{3}$$

其中,$|P|$ 是信息系统中条件属性的个数,$|\Psi|$ 是区分矩阵项的长度,a 是一个条件属性。该函数的函数值是信息系统中属性重要程度的一个量度,函数值越大,表明该属性的区分能力越强。

BFF 算法生成约简的过程如下: 首先将区分矩阵 Ψ 按照项长度和项出现的频率进行排序; 然后对排好序的区分矩阵进行遍历, 当 Redu 与区分矩阵的项相交为空时, 从该项中选择属性频率值最大的属性添加到约简中。对区分矩阵遍历结束时, 便可得到该信息系统的一个约简。

3.3　问题分析与结论

BFF 算法与文献[3] 提到的启发式算法一样, 也存在两方面问题: 1) 启发式约简算法的计算时间主要依赖于核的长度, 如果一个数据集无核或核比较小, 则约简计算量非常大, 计算时间很长; 2) 在用启发式约简算法对初始区间进行约简时, 有时找不到能对所有属性离散化的约简。

由属性频率函数的定义可知, 频率函数值越大的属性, 在区分矩阵中出现的频率越高, 其区分能力也越大, 而且经常存在于较短的区分矩阵项中。当用约简Redu 对区分矩阵进行遍历时, 属性频率函数值最大的属性最容易添加到Redu 中, 特别是对于属性和记录的个数都比较多、具有较大的区分矩阵的信息系统。因此有如下结论:

结论 **1**　频率函数值最大的初始区间记为 p_{max}, 即 $f(p_{max}) \geq f(p_j^i), 0 < i \leq m, 0 < j < n^i$。它一般存在于初始区间的约简中。

同理, 频率函数值最大的初始区间在对初始区间的区分矩阵进行遍历时, 也是最容易添加到初始区间的约简中。因此可得如下推论:

推论 **1**　如果 S 中某个属性 c^i 存在频率函数值最大的初始区间 p_{max}, 则该属性是最容易被离散化的。

同样, 根据属性频率函数的定义可知, 频率函数值越小的属性, 在区分矩阵中出现的频率越低, 区分能力也越小。当用约简Redu 对区分矩阵进行遍历时, 这些属性最容易被忽略。同理, 在初始区间对区分矩阵进行遍历时, p_{max}^i 小的属性容易被忽略, 即这类属性最难被离散化, 特别是当这类属性的初始区间频率值远小于其它属性的初始区间频率值时。因此有以下结论:

结论 **2**　如果各属性中频率函数值最大的初始区间记为 p_{max}^i, 即 $f(p_{max}^i) \geq f(p_j^i)$, 从小到大排列得到一个集合 P_{MAX}, 那么 P_{MAX} 前面的初始区间所对应的属性最易在约简后丢失, 特别是频率函数值最小的初始区间所对应的属性。

依据上述结论, 可以对每个属性中最大频率值的初始区间进行排序, 即得到下面引入的候选核的概念, 并根据排序的次序进行启发式约简。这就是本文提出的基于候选核启发式约简算法的基本思想。

3.4　候 选 核

在粗糙集理论中核是一个重要的概念, 记为 Core。

定义 **4**(核)[4]　核是指所有约简的交集, 即

$$\mathrm{Core}(P,Q) = \cap \mathrm{Redu}(P,Q) \tag{4}$$

在区分矩阵中核的意义更为明显。设 $\Psi(i,j) \in \Psi$, Ψ 为区分矩阵, $\Psi(i,j)$ 为区分矩阵中的一项。若 $|\Psi(i,j)| = 1, \forall a \in P, a = \Psi(i,j)$, 则有 $a \in \mathrm{Core}(P,Q)$。如果约简中不包含这个核属性, 则该约简便无法区分产生这一项的 i 和 j 两条记录。

定义 **5**(候选核)　候选核是指各属性的最大频率值初始区间从小到大排列的有序集合, 即

$$\mathrm{Core}' = \{ \sum_{k=1}^{m} p_k \mid p_k \in \sum_{k=1}^{m} p_{max}^i, f(p_{k-1}) < f(p_k) \} \tag{5}$$

候选核中 p_k 从小到大排序, 该次序也是在启发式约简中考虑是否添加到约简中的次序, 即首先考虑将 Core' 中频率值小的 p_k 添加到初始区间的约简中。在实际运用中还发现如下结论及推论:

结论 **3**　如果 p_j^i 满足下面两个条件: 1) $p_j^i \in \mathrm{Core}$; 2) 各个 p_j^i 是不同属性的初始区间。则有 $p_j^i = p_{max}^i \in \mathrm{Core}'$, 且 $\mathrm{Core} \subseteq \mathrm{Core}'$。

推论 **2**　如果 $|\mathrm{Core}| = 1$, 则有 $\mathrm{Core} \subseteq \mathrm{Core}'$。

推论 **3**　如果 Core 中存在属于同一属性的两个初始区间, 则有 $\mathrm{Core} \not\subset \mathrm{Core}'$。

由结论 3 及其两个推论可知, 尽管核与候选核在概念上不同, 但从区分能力角度看, Core 是区分能力最强的初始区间的集合, 候选核是各个属性中区分能力最强的初始区间的集合, 因此这两个概念有一定的联系。当核为空或核满足结论 3 的条件时, 便可认为候选核是核的概念的一个扩展。

3.5　基于候选核的启发式约简算法

下面以候选核概念为基础, 提出一种新的启发式约简算法 —— 基于候选核的启发式约简算法, 简称 BCC 算法。

BCC 算法

输入: 信息系统 $S = (U, A, V, f)$, 初始区间的集合 $\{p_j^i\}, 0 < i \leq m, 0 < j < n^i$;

输出: 初始区间的约简

$$\mathrm{Redu} = \Phi, \quad \mathrm{Core} = \Phi$$
$$\mathrm{Core} = \Phi, \quad f(p_j) = 0$$

计算初始区间的区分矩阵 Ψ, 并计算各个初始区间的频率函数 $f(p_j^i)$;

对初始区间的区分矩阵 Ψ 进行合并, 并排序;

生成核 Core 和候选核 Core', 删去 Core' 中与 Core 所包含的初始区间属于同一属性的初始区间;

令 Redu = Core

For Ψ 中的每项 $\Psi(i,j)$ do

If($\Psi(i,j) \cap$ Redu = = Φ)

If ($\Psi(i,j) \cap$ Core' = = Φ)

选择 $\Psi(i,j)$ 中 $f(p_j^i)$ 最大初始区间 p_j^i

Redu = Redu $\cup \{p_j^i\}$

If ($\{p_j^i\} \cap$ Core' = = p_k^i)

Core' = Core' - p_k^i

Endif

// 如果候选核中含有与初始区间 p_j^i 属于同一属性的初始区间 p_k^i, 则从候选核中删去 p_k^i //

Else

选择 Core' 中与 $\Psi(i,j)$ 相交的 $f(p_j^i)$ 最小的初始区间 p_j^i

Redu = Redu $\cup \{p_j^i\}$

Core' = Core' - p_j^i

Endif

Endif

EndFor

If (Core' $\neq \Phi$)

Redu = Redu $\cup$ Core'

Endif

Return Redu

该算法在约简 Redu 与区分矩阵中的项相交为空时, 并不仅仅将该项中频率值最大的初始区间添加到约简 Redu 中, 而考虑它与候选核相交是否为空。如果出现它们相交为空的情况, 则将该项中频率值最大的初始区间添加到约简 Redu 中, 并检查候选核中是否有与新添加的初始区间属于同一属性的初始区间, 如果有, 则从候选核中删除; 如果该项与候选核相交不为空, 则在它们相交产生的交集中选取频率值最小的初始区间添加到约简 Redu 中, 同时在候选核中删除该初始区间。需要注意的是: 在算法的最后, 当出现条件 Core' $\neq \Phi$ 时, 得到的约简已不是严格定义的约简, 这种情况出现得较少, 一般出现在初始区间只有唯一约简的情况下。

可以看出, 由于提出了候选核的概念, 保证了每个属性都有初始区间, 即克服了无核对启发式约简算法的影响。本文提出的基于候选核的启发式约简算法减少了运算量, 提高了计算速度。

4 实验与讨论

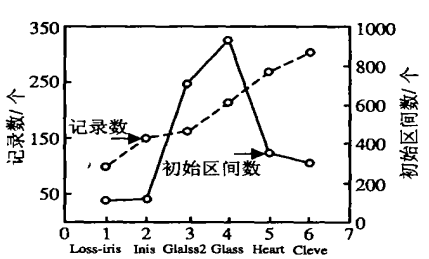
本实验分别对用 BFF 和本文提出的 BCC 约简的 NS 算法, 以及局部离散化中经典的最小信息熵离散化算法(RMEP 算法) 进行比较。

实验采用 MLC + + 工具箱中的 RMEP 算法进行离散化, 其它两种算法用 C + + 语言编写。离散化后用 RSL[6] 中提供的粗糙集算法对数据进行测试, 实验采用的数据集是 UCI 中的 Iris, Glass 和 Cleve 等数据集[5]。

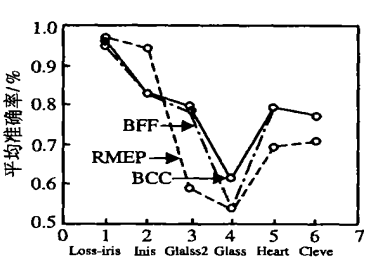
实验对用 BCC 约简的 NS 算法、用 BFF 算法的 NS 算法和 RMEP 算法在最好准确率、平均准确率和依赖度等指标上进行比较。离散化区间数和初始区间数可由定义得出; 依赖度可由式(1) 求出, 该值越大, 说明离散化后信息丢失越少; 最好准确率和平均准确率是指利用 RSL 中提供的粗糙集算法[6], 从训练集中获取规则后对测试集进行判断所获得的最好准确率和平均准确率。实验结果如图 1 所示。

通过实验发现:

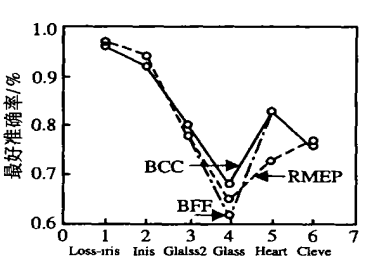
1) RMEP 算法不太理想, 因为大多数数据集都不能对全部属性离散化, 这是导致离散化效果下降的重要原因。例如在 Glass2 中的 9 个连续值属性中, 就有 4 个不能进行离散化, 使得它的平均准确率低



(a) 数据集的记录数和初始区间



(b) 平均准确率



(c) 最好准确率比较

图 1 实验结果

于其它两种算法约0.2。从实验结果可以看出,当不能对数据集中全部属性进行离散化时,离散化效果会下降,例如在 Glass 和 Glass2 中,BFF 算法得到的是缺少一个属性的约简,所以它的准确率低于 BCC 算法。可见,在大数据集中能否对所有属性进行离散化,将对离散化后的效果产生一定的影响。

2) 本文提出的 BCC 算法的离散化效果比较好,而且离散化效果相对于基于频率函数的算法,在 Glass 和 Glass2 两个初始区间较大的数据集中有一定的提高。

3) 由图 1(c)可见,各种算法的准确率并没有随记录数有规律地排列,而是随初始区间数的增加而减少。另外,在依赖度等指标上的实验也验证了 BCC 算法的有效性。可见,当数据量较大时,用 BCC 算法约简的 NS 算法能保持较好的离散化效果。

5 结　　语

本文通过对 NS 算法的研究,在粗糙集理论和属性频率函数的基础上提出了候选核和基于候选核的 BCC 算法,解决了该离散化算法存在的某些属性不能离散化等问题。实验证明,该算法具有较好的离散化效果。本文在研究中总结出的一些规律对于这类算法具有一定的实际意义。

参考文献(References):

[1] Dougherty J, Kohavi R, Sahami M. Supervised and unsupervised discretization of continuous features[A]. *Proc of 12th ICML*[C]. Morgan: Kauffman Publishers, 1995. 194-202.

[2] Nguyen H S. Discretization of real value attributes: A boolean reasoning approach[D]. Warsaw: University of Poland, 1997.

[3] Hu Keyun. Research on concept lattice and rough set based data mining methods[D]. Beijing: Tsinghua University, 2001.

[4] Hu X H. Knowledgediscovery in databases: An attribute-oriented rough set approach[D]. Regina: University of Regina, 1995.

[5] Ronny Kohavi, Dan Sommerfield. MLC++: Machine learning library in C++ [CP/DK]. http://www.sgi.com/technology/mlc, 2001-06-01.

[6] M Gawrys, J Sienkiewicz. rough set library user's manual [CP/DK]. ftp://ftp.ii.pw.edu.pl/pub/Rough/, 1993-09-16/2001-05-08.

(上接第 544 页)

参考文献(References):

[1] Burton H Lee. Embedded internet system: Poised for takeoff[J]. *IEEE Internet Computing*, 1998, 14(2): 24-29.

[2] 赵海, 陈飞鸣. Embedded Internet 的体系结构及其 DNDC 模型的实现[J]. 东北大学学报, 1999, 20(3): 257-260.
(Zhao Hai, Chen Feiming. Embedded internet architecture and establish of the ONDC module[J]. *J of Northeastern University*, 1999, 20(3): 257-260.)

[3] Garvey A, Lesser V. Design to time real-time scheduling[J]. *IEEE Trans on Systems, Man and Cybernetics*, 1993, 23(6): 58-67.

[4] Elliotte Rusty Harold. XML 实用大全[M]. 北京: 中国水利水电出版社, 2000. 83-85.

[5] Ready J F VRTX: A real-time operating system for embedded microprocessor applications[J]. *IEEE Micro*, 1986, 6(4): 8-17.

[6] P Paulin, M Cornero, C Liem, et al. Trends in embedded systems technology[A]. Hardware/Software Codesign, Kluwer Academic[C]. Publishers, 1996.

[7] 赵海. 现场总线网络通信模型中的逻辑链路层[J]. 东北大学学报, 1996, 17(6): 230-234.
(Zhao H. Logic link layer in the communication model of fieldbus network [J]. *J of Northeastern University*, 1996, 17(6): 230-234.)

[8] 陈功富, 韩贤东. 计算机网设计与实现[M]. 北京: 人民邮电出版社, 1994. 146-167.