基于决策的剥离式连续属性离散化算法*)

潘 巍[1] 李晋川[2] 王阳生[3] 杨宏戟[4]

(首都师范大学信息工程学院 北京 100037)[1] (四川大学 成都 610065)[2]

(中国科学院自动化研究所模式识别国家重点实验室 北京 100080)[3]

(Software Technology Research Laboratory, De Montfort University, Leicester, LE1 9BH, England)[4]

摘 要 针对粗糙集理论只能处理离散数据的局限,提出了基于决策的剥离式连续属性离散化方法,一改传统的候选断点集合的获取方法,直接通过分析连续属性在各决策类的取值范围和计算属性重要度,完成对连续属性的初步离散。此外,本文提出候选断点集的推移原则,可逐步减小候选断点集的范围。由于每次都是针对尚不能明确分类的样本进行细化,因此随着候选断点集的减少和明确分类样本的增加,系统能够迅速收敛,并且离散化后的决策表总是相容的,这与目前很多离散方法不考虑决策相容性相比,能够最大限度地保留系统的有用信息。本文提出的离散化方法是领域独立的,不需要领域知识,可应用于不同领域的连续属性的离散化。

关键词 粗糙集理论,属性离散化,候选断点,决策相容性

A New Algorithm of Discretization of Consecutive Attributes Based on the Decision in Rough Sets

PAN Wei[1] LI Jin-Chuan[2] WANG Yang-Sheng[3] YANG Hong-Ji[4]

(Institute of Information Engineering, Capital Normal University, Beijing 100037)[1] (Sichuan University, Chengdu 610065)[2]

(Institute of Automation, Chinese Science Academies, Beijing 100080)[3]

(Software Technology Research Laboratory, De Montfort University, Leicester, LE1 9BH, England)[4]

Abstract Proposed a new algorithm of discretization of consecutive attributes based on the decision according to the limitation that Rough Set Theory can only deal with the discrete attributes in database. Unlike traditional methods, the initial candidate points are obtained by analyzing the distribution ranges of consecutive attributes in each decision sort and computing their attribution significances. At the same time, proposed some rules of decreasing candidate points in order to increasee the velocity of system convergence. Using the algorithm, the decision table after discretization will be always consistent and can reserve useful information as much as possible. Finally, the algorithm is field-independent and can be used in different fields without any additional information.

Keywords Rough set theory, Attribute discretization, Candidate point, Decision consistency

1 引言

粗糙集理论是波兰数学家 Z. Pawlak 在 1982 年提出的一种数据分析理论,其基本原理是在模拟人类对不明确问题的认知和处理方式的基础上,对现有的不完全知识进行约简和求核,并以此作为决策依据,对观测、度量到的某些不精确数据进行分类。由于对不确定性的描述相对客观,目前,粗糙集理论已被成功地应用于机器学习、故障诊断、决策分析、过程控制、模式识别、数据挖掘等领域,并取得了成功。

遗憾的是,由于粗糙集理论处理的是具有离散属性值数据的集合,而在多数情况下,同一个数据库中既包含离散属性,又包含连续属性,因此需要对连续属性数据进行某种方式的离散化。离散化的实质是把落在连续属性取值区间的某个子区间上的数据点看作是不可分辨的,用同一个代码代替。离散化必然会造成部分信息的损失,并可能遇到相当繁琐的计算,不同的离散化方法的计算复杂性和信息损失量是不相同的,因此研究使信息损失尽可能少,计算量尽可能小的离散化方法是重要的,这也是粗糙集理论的一个应用研究方向。

2 基于决策的剥离式连续属性离散化算法

连续属性离散化过程本质上就是采用一定的断点集合对决策系统的连续属性进行划分。为了提高系统的聚类能力,增强系统对噪音的鲁棒性,应该采用尽可能少的断点来完成划分过程,从这一角度来说,在保证系统分类能力的前提下,用最小的结果断点集合对系统进行的离散化就是基于粗糙集理论的最优离散化[1]。

基于粗糙集理论的离散化方法大致可分为两类,其中一类直接把其他学科中的离散化方法借用到粗糙集理论中来,很少或者不考虑粗糙集理论的特性,如等距划分法、等频划分法、Navie Scaler 算法等,这此算法不需要额外的参数,直接根据信息表进行离散化,但它们一次仅考虑单个属性,得到的信息表可能引入新的冲突,因此这一类离散化算法效果并不突出。另一类方法是结合粗糙集理论解决离散化问题,例如 Skowron 提出的布尔代数与 RS 理论相结合的算法[2,3],此外

*)基金项目:国家 863 高技术研究发展计划项目(编号:2003AA114020)。潘 巍 博士,讲师,研究方向为模式识别,信息融合。李晋川 博士,副教授,研究方向为软件仿真。王阳生 研究员,博士生导师,研究方向为模式识别。杨宏戟 博士生导师,英国 De Montfort 大学软件技术研究室主任,研究方向为软件工程。

还有基于断点重要性[1,4,5]、粗糙熵[6,7]、属性重要性[8]的算法等。从领域知识上分，又可分为基于领域知识[9,10]和独立于领域知识的离散化方法[1,11]。

对于一个协调的决策表而言(如果不协调，则先要转换成协调决策表后再进行离散)，如果条件属性的划分较粗(值域较小)，可能导致划分后的决策表不相容；如果划分较细，则可能使划分后的决策表中仍含有很多冗余信息，降低约简效率。所以，我们对连续属性化的目标是，在尽量减少信息损失的前提下寻找使得约简效率最高的划分。我们受到最优决策规则提取方法的启发，提出了一种领域独立的基于决策的剥离式连续属性离散化算法，即每一轮的断点都要依据决策属性进行选择，并尝试用最少的属性对样本进行分类，如果对样本不能完全覆盖，才会考虑增加新的属性参与离散。断点确定后，剥离出已能明确分类的样本，将剩余的样本送入下一轮的断点选择，如此循环，直到决策表完全相容或满足需要的条件。剩余的样本实际上就是在离散化后出现决策矛盾的分类，换言之，系统无法根据现有离散数据对其进行正确分类，需要加入新的断点对剩余样本进行更深的细化。

2.1 候选断点集合的选择

数据离散的过程是很直观的，选取一定的断点就能够实现离散。为了生成连续属性 c 的候选断点集合，通常的做法是：对 c 的所有值进行排序得$\{v_1, v_2, \cdots, v_m\}$，任取 $E_i \in (v_i, v_{i+1}), i=1,2,\cdots,m-1$，$\{E_i\}$就构成 c 的一个候选断点集合。随机地选取候选断点进行离散，并根据某种原则例如粗糙熵[7]、未确知测度[8]、不相容度[12]等判断离散后的系统分类性能，当满足要求后数据离散的过程结束。系统进行离散需要的时间与候选断点集的大小成正比。因此，本文倾向于更小的候选断点集合，或者说根据需要增加候选断点，而不是一开始就确定一个庞大的数量。

本文用一个简单的图表说明候选断点集合的选择过程。设决策表 $S=\langle U, C\cup D, V, f\rangle$，$Y_i$ 是根据 D 划分的等价类，$i=1,2,\cdots$决策种类，$Y_i \cap Y_j = \Phi$。在图 1 中，共有 4 个决策种类，$[\mathrm{Min}_i, \mathrm{Max}_i]$分别是连续属性 c 在 Y_i 上的取值区间，从图中可以很直观地看出，c 在 Y_1 上的取值区间$[\mathrm{Min}_1, \mathrm{Max}_1]$没有与其它区间重叠，表明根据此区间的数据能够明确样本 $x_j (x_j \in U)$ 对 Y_1 的归属关系，将该区间的所有数据聚成一类不会造成决策冲突。同理，$[\mathrm{Min}_2, \mathrm{Min}_3), (\mathrm{Max}_2, \mathrm{Max}_4]$能分别明确样本 x_j 对 Y_2、Y_4 的归属关系，至于其它区间的数据，仅根据连续属性 c 无法明确样本 x_j 的归属，需要结合决策表中的其它属性进行联合划分。经过第一轮的离散化，可以得到 c 的明确断点集合：$\{\mathrm{Min}_2, \mathrm{Min}_3, \mathrm{Max}_2\}$和候选断点集合区间：$(\mathrm{Min}_3, \mathrm{Max}_2)$。可以看到，候选断点集合是在第一轮离散过后产生的，如有必要，可用于第二轮的断点选择，显然它的范围要比基于全值域的候选断点集合小得多。

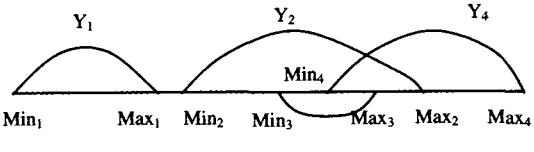

图 1 连续属性 c 的候选断点集合选取示例

2.2 属性重要度 μ

按照上述方法可以得到连续属性 c 对 Y_i 的重要度 μ_{ci} 及所有决策类的重要度 μ_c，属性重要度 μ 的定义为

$$\mu_{ci} = \frac{d_i}{\mathrm{Max}_i - \mathrm{Min}_i}, \mu_c = \sum_{i=1}^{m} \mu_{ci} \quad (1)$$

其中 d_i 是 c 在 Y_i 上没有与 $Y_j (i \neq j)$ 重叠的所有取值区间之和。属性重要度 μ_{ci} 描述了仅根据 c 就可以明确样本 x_j 对 Y_i 的归属问题的能力，$\mu_{ci}=1$ 时可得到决策规则 $x_j \in U, v_{x_j c} \in [\mathrm{Min}_i, \mathrm{Max}_i] \Rightarrow x_j \in Y_i$，$v_{x_j c}$ 是样本 x_j 在属性 c 上的取值。$\mu_{ci}=0$ 时表示仅根据 c 无法明确任何样本 x_j 对 Y_i 的归属。

由大到小对属性重要度 μ_{ci} 和 μ_c 进行排序，把可以明确分类的样本从决策表中剥离出来，如果根据某一属性，发现可以剥离的样本数为 0，说明在本轮离散过程中，该属性在各决策类上的取值区间是冗余的，不必对其进行分类。剥离结束后，如果决策表不为空，则将剩余样本形成新的决策表，参与下一轮的离散。

2.3 候选断点的推移原则

每轮离散结束后所有产生决策矛盾的样本都会形成新的决策表，重新针对剩余的决策类计算属性重要度，并根据属性重要度由大到小依次对样本进行剥离。如果仅根据单一属性无法继续对样本细化，则需要对候选断点集合进行推移，进一步缩减候选断点集合。为此，本文制定了三大推移基本原则：一是单次推移所产生的新断点应尽量少，二是尽可能把决策值相同的相邻断点合为一类，三是如果相邻断点之间的决策值各不相同，应尽量把可明确分类的相邻断点合为一类，以便于下一次的再分类。

设连续属性 c 的候选断点集合为$\{v_i\}$，$i=1,2,\cdots,m$，$v_i < v_j$，$i<j$，m 为候选断点数，推移算法如下：

1)遍历当前决策表中每一个样本的属性 c 值和对应的决策值，记入 Candidate [k]. value，Candidate [k]. decision [t]。k 用于计数不同 c 值的个数，初始为 0，每遇到一个新值，$k=k+1$，t 用于标记该 c 值所对应的决策值，Candidate [k]. decision [t]的维数即决策种类数，初始均为 0，例如：若 c 值对应的决策值为 3，则相应的 decision [2]=1(从 0 开始计数)。

2)对 Candidate 排序，使 Candidate [i]. value<Candidate [j]. value，$i<j$，Candidate. decision 应同时做相应调整。

3)取 Candidate [0]. decision [j]，$j=1,2,\cdots$决策种类数，如果只有 1 个是"1"，其它的为"0"，说明 Candidate [0]有明确的分类能力，Candidate [0]. label＝相应的决策类；如果有多个为"1"的情况，说明 Candidate [0]没有明确的分类能力，Candidate [0]. label＝－1，计数器 Count＝1，Count 用于显示连续推移能力。

4)开始推移。推移过程中，除非必要，否则不希望出现一个连续值即为一个类的情况，所以会预先设定一个推移阈值 T_{count}，通常可取 T_{count} ＝候选断点数/3，随着细化的深入，T_{count} 逐渐变小。Tmpcandidate 用于存放推移过程中的可疑断点，初始为－1。

推移的原则很简单，就是把独立候选断点加入相邻的连续断点集：

(1)if 相比较的候选断点标记相同，then Count＝Count＋1；

(2)else if Count$<T_{\mathrm{count}}$，Count＝Count＋1；

(3)else if Candidate [$i-1$]. label＝－1，认为当前断点可疑，保留待查，令 Temp＝Count，Tmpcandidate＝i，Count＝1，$i=i+1$，继续推移计算，当出现两者标记不同且后者标记不为－1 时，如果 Count$<T_{\mathrm{count}}$，取消 Tmpcandidate，新断点为 i，如果 Count$<T_{\mathrm{count}}$，Tmpcandidate 和 i 都成为新断点；如果后者标记为－1 且 Count$<T_{\mathrm{count}}$，取消 Tmpcandidate，Count＝Count＋Temp，如果 Count$>T_{\mathrm{count}}$，Tmpcandidate 和 i

都成为新断点。

(4)else if Candidate[i+1].label=−1,认为当前断点可疑,保留待查,其方法与(3)相同。

(5) else i 成为新的断点,Count=1。

当前属性完成候选断点推移后,用新的断点对当前决策表进行剥离,剥离后的剩余样本组成新的决策表,用于下一个属性的推移。所有属性完成推移后,如果决策表中仍有剩余样本,则缩小 T_{count},将连续属性在决策表中的数据作为候选断点集再次进行推移,直到决策表为空,离散化过程全部结束。

3 实验

IRIS数据集是在分类问题中被广泛运用于测试的数据集,该数据集有4个连续条件属性 sepal_length,sepal_width,petal_length,petal_width 和3个决策分类 Iris-setosa(类别1),Iris-versicolor(类别2),Iris-virginica(类别3),各属性在3种决策分类上的取值范围、属性重要度 μ_{ci}、μ_c、第一轮离散断点如表1所示。

从表1可以发现,根据属性重要度,petal_length 和 petal_width 要比 sepal_length 和 sepal_width 重要,它们都可以实现对 Iris-setosa(类别1)的完全分类,因此,可随机选择一个作为决策规则。需要指出的是,我们用于判断属性重要度的方法在文[8]中得到了验证,刘开第等人用未确知测度与聚类神经网络相结合的方法分析 Iris 各属性的重要性,其结果与我们的类似,但我们的方法更简单快捷。

由表1,可得到第一轮的离散结果:petal_length:[1,3),[3,4.5),[4.5,5.2),[5.2,6.9],petal_width:[0.1,1.4)[1.4,1.9),[1.9,2.5],sepal_length 和 sepal_width 的所有数据暂时分为一类。

经过第一轮离散后,新的决策表中只剩余26个样本,按属性重要度依次对各属性的候选断点集进行推移,得到新的断点集,第二次离散结束后,决策表为空,离散过程结束。此时,各属性的离散数据为:sepal_length{6,6.4},sepal_width{3.1},petal_length{3,4.5,5,5.2},petal_width{1.4,1.6,1.9}。

本文比较了 Teghem、蔡智[7]、Naïve Scaler 算法、Semi Navie Scaler 算法及本文方法对 IRIS 数据集进行离散后的系统分类能力,结果如表2所示。

采用本文方法,经过第一轮离散过程后,即能以最少的断点数达到较低的误判数。由于只是进行简单的排序、计算属性重要度和剥离明确分类的样本,计算量远小于其它方法。此时,如果不考虑误判样本,对离散后的决策表提取决策规则,本文得到8条长度为2的决策规则,在以上各种方法中,规则数是最少最短的,充分遵循了使得约简效率最高的离散原则。

在对产生冲突的26个样本进行第二次离散化后,决策表中的数据全部相容,断点数虽有所增加,但也大大小于 Semi Naïve 算法,总体分类性能在几种方法中具有很强的优势。我们用文[8]提供的实验数据(20个样本,6个连续属性,3个决策属性)用同样的方法进行离散,也取得了良好的分类效果。

小结 针对粗糙集理论只能处理离散数据的局限,提出了领域独立的基于决策的剥离式连续属性离散化方法,一改传统的候选断点集合的获取方法,直接通过分析连续属性在各决策类的取值范围和计算属性重要度,完成对连续属性的初步离散,此时生成的候选断点集中只含有尚不能明确分类的数据,数量得到有效的控制。此外,本文提出候选断点集的推移原则,可逐步减小候选断点集的范围。由于每次都是针对尚不能明确分类的样本进行细化,因此随着候选断点集的减少和明确分类样本的增加,系统能够迅速收敛,并且离散化后的决策表总是相容的,这与目前很多离散方法不考虑决策相容性相比,能够最大限度地保留系统的有用信息。本文提出的离散化方法是领域独立的,不需要领域知识,可应用于不同领域的连续属性的离散化。

表1 Iris数据集中各属性对决策的重要度

	sepal_length			sepal_width			Petal_length			petal_width		
类别	1	2	3	1	2	3	1	2	3	1	2	3
取值范围	4.3～5.8	4.9～7	4.9～7.7	2.3～4.4	2～3.4	2.2～3.8	1～1.9	3～5.1	4.5～6.9	0.1～0.6	1～1.8	1.4～2.5
μ_{ci}	0.4	0	0.25	0.29	0.14	0	1	0.71	0.75	1	0.5	0.64
μ_c	0.65			0.43			2.46			2.14		
离散断点	无			无			3,4.5,5.2			1.4,1.9		

表2 不同离散方法对IRIS数据集的离散性能比较

	Teghem	粗糙熵	Naïve	Semi Naïve	本法(第1轮)	本法(第2轮)
平均断点数	3	2	3	14	1.25	2.5
误判数	15	16	37	0	8	0

参考文献

1 赵军,王国胤,吴中福,等.基于粗集理论的数据离散化方法.小型微型计算机系统,2004,25(1):60～64

2 Nguyen H S. Discretization problem for rough sets methods. In: Proc. of 1st International Conference on Rough Sets and Current Trends in Computing (RSCTC'98), 1998. 545～552

3 Nguyen H S, Skowron A. Quantization of real values attributes, roughset and Boolean reasoning approaches. In: Proc. of 2nd Joint Annual Conference on Information Science, 1995. 34～37

4 Xiang Xin-jian, Stolle M. An algorithm of discretization of continuous attributees in rough sets based on cluster. Journal of Zhejiang University of Science and Technology, 2003,15(3):154～157

5 Zhao Jun, Wang Guo-yin, Wu Zhong-fu, et al. New algorithms for data discretization based on rough set theory. Journal of Chongqing University (Natural Science Edtion), 2002,25(3):18～21

6 沈东升.一种连续属性离散化的新算法.漳州师范学院学报(自然科学版),2003,16(4):27～30

7 蔡智,王煦法,蔡庆生.基于粗糙集理论的连续属性离散化算法研究.计算机科学,2001,28(5.专刊):39～41,22

8 刘开第,王义闹,庞彦军.基于聚类神经网络的连续属性离散化方法.计算机科学,2001,28(5.专刊):136～137,168

9 Slowinski R. Rough classification of HSV patients. Intelligent Decision Support, Kluwer: Roman Slowinski, 1992. 77～94

10 Hu X H, Cercone N. Learning in relational databases: A rough set approach. International Journal of Computational Intelligence, 1995,11(3):323～338

11 陈遵德,张荣进.基于K均值与粗集理论结合的模式分类方法.计算机科学,2001,28(5.专刊):64～66

12 苗夺谦.Rough Set理论中连续属性的离散化方法.自动化学报,2001,27(3):296～302