连续属性的频数监督断点离散化技术

林汀辉,史　亮,姜青山
LIN Ting-hui,SHI Liang,JIANG Qing-shan

厦门大学 软件学院,福建 厦门 361005
Software School,Xiamen University,Xiamen,Fujian 361005,China

LIN Ting-hui,SHI Liang,JIANG Qing-shan.Frequency-supervised-breakpoint based discretization algorithm.Computer Engineering and Applications,2009,45(2):128-130.

Abstract: In this paper,a discrete algorithm is proposed based on the frequency supervised breakpoint which is applied to the conditional continuous attributes.The algorithm adopts the idea of frequency supervised breakpoint that is brought forward to generate initial breakpoints.On the basis of the preparatory work,reduction of breaktpoints has been performed.The result obtained shows clearly the breakpoints generated by this algorithm not only are in line with the actual data distribution but also they are more reasonable,refined.

Key words: frequency-supervised-breakpoint;equivalent;discretization

摘　要 提出了一种频数监督断点的离散化算法。该算法利用所提出的频数监督断点思想产生初始断点,并在此基础上进行断点简约。实验结果表明该算法所产生的断点不仅符合实际数据分布,而且更为合理、精练。

关键词 频数监督断点;不可分辨;离散化
DOI:10.3778/j.issn.1002-8331.2009.02.037　　**文章编号**:1002-8331(2009)02-0128-03　　**文献标识码**:A　　**中图分类号**:TP301.6

1　引言

连续属性的最优离散化是一个 NP 完全问题[1-4],作为数据挖掘和机器学习的重要预处理步骤[5],离散化方法的性能将对后续的数据挖掘任务产生直接影响,不好的离散化会导致一些关键信息的丢失或造成沉重的挖掘负荷。

目前已经提出许多种离散化算法[6-10],包括 Naïve Scaler[10]、等宽、等频[2]、信息熵[9]等。文献[2,8,10]是基于粗糙集理论,采用无监督方式从集合角度来分析,虽然能较好地反映数据的统计分布,但不能较好融合属性的知识背景,在集合的划分上可能改变了原有的域值分类,不能较好地保持原有等价关系;文献[3,11-13]虽然考虑了属性之间的关系,但产生断点上属于硬性划分,断点取舍上较为单一,只由属性之间关系决定,未考虑类别信息的比较应用。

针对当前属性离散化方法中断点选取所存在的数量和选取合理性问题,提出一种基于频数监督断点的离散化算法 FSBDA(Frequency-Supervised-Breakpoint based Discretization Algorithm)。该算法通过引入频数监督断点的思想产生初始断点集,并对该初始断点集进行简约,从而完成属性的离散化。

2　离散化算法 FSBDA

FSBDA 算法的设计思想体现了对离散化方法要求的理解,在保持数据在离散过程中的不可分辨关系和在一定程度上保证决策表原有分类结果的不变性的基础上,尽可能降低断点数量,并使所得到的断点不仅符合数据分布,又体现属性的内在知识背景。以下将对算法实现中所涉及到的部分重要概念进行阐述,并给出详细的算法实现流程。

2.1　频数监督断点

断点的产生分为监督离散化和非监督离散化,监督式方法把类别的信息带到离散化的过程。其中 D 值的引入是类别信息应用,属于监督式的方法。D 值根据实际情况的不同可以有好多分类,如可以分故障和非故障;在网络异常数据方面,可分出 Dos 攻击、DNS 攻击、蠕虫病毒等,D 值类别信息在决策表中的决策属性这一列体现,是整个决策表的关键属性。现有的基于频数断点的离散化方法就是基于支持度,断点的选取考虑了属性值出现的次数。但该方法仅从数学角度来处理,属于无监督方法,断点选取的合理性不足。因此,对频数断点的方法进行改进,提出了频数监督断点思想。它结合了 Naive Scaler 的基本算法思想,并在断点产生和简约过程中考虑了 D 值的应用。

不妨设初始数组 A,该数组用来存储一个条件属性 a 里面的值,且已经按照从小到大的顺序排列数组 A'。设 $a(x_1)'$,$a(x_2)'$,…,$a(x_N)'$ 是数组 A' 中的元素,$d(x_1)'$,$d(x_2)'$,…,$d(x_N)'$ 是对应的决策值,一般相邻不同的值且其对应的决策值不相等,则

基金项目:国家自然科学基金(the National Natural Science Foundation of China under Grant No.10771176)。
作者简介:林汀辉(1983-),男,硕士研究生,主研方向:网络信息安全、数据挖掘;史亮,副教授,博士;姜青山,教授,博士生导师。
收稿日期:2008-07-08　　修回日期:2008-11-17

它们之间可以产生一个断点。断点的产生规则如下：

设 $a(x_i)$ 与 $a(x_j)$ 是属性 a 中两个值,其中 $a(x_i)\neq a(x_j)$,$i<j$,$d(x_i)\neq d(x_j)$ 且不存在 $\forall_k\{a(x_i)<a(x_k)<a(x_j)$ and $(i<k<j)\}$。支持度 sup (Y) 定义成论域中 Y 值个数$|Y|$。$sup(a(x_i))$ 和 $sup(a(x_j))$ 对应于 $a(x_i)$ 和 $a(x_j)$ 两个的支持度$|a(x_i)|$和$|a(x_j)|$,则它们之间可以产生一个断点 c_a^y：

$$c_a^y=\frac{V(a(x_i))*|a(x_i)|+V(a(x_j))*|a(x_j)|}{|a(x_i)|+|a(x_j)|} \tag{1}$$

$V(a(x_i))$ 为论域 x_i 的值,按照公式 (1) 可以产生一组初始断点 $V_a=\{c_a^1,c_a^2,c_a^3,\cdots,c_a^{N-1}\}$,$V_a$ 中的断点也是按顺序排列。考察 c_a^i 与 c_a^{i+1} 之间是否有任何 $a(x_i)$ 值存在,如果有,则 c_a^i 与 c_a^{i+1} 断点都被保留 否则,只留下其中一个断点,被保留的断点即为所求的频数监督断点。最后得到一组频数监督断点集合 $V'_a=\{c_a^1,c_a^2,c_a^3,\cdots,c_a^k\}$。

2.2　断点简约

在断点集 V_a 上,考虑断点 c_a^i 的取舍,分别取关于 c_a^i 的上值 *saveup* 与下值 *savedown*,其中 *saveup*,*savedown* $\in a(x_i)$,把原先 $a(x_i)=savedown$ 的值相应改成 $a(x_i)=saveup$,不考虑前面已经处理过的属性,若与现有任何一个 $a(x)=saveup$ 的整条 x_i 记录不矛盾,则进行替换,否则,应把相应的值改回。最终得到关于属性 a 的新值 $Val=\{a'(x_1),a'(x_2),\cdots,a'(x_N)\}$和最终断点集合 $V'_a=\{c_a^{1'},c_a^{2'},c_a^{3'},\cdots,c_a^{k'}\}$。

2.3 FSBDA 算法流程

整个 FSBDA 算法在过程中，关键步骤在于获取频数监督断点和噪声和边缘化处理。假设 $U=\{x_1,x_2,\cdots,x_N\}$为具有 N 个实例的全域 (universe),A 为条件属性集,d 属为决策性,这样决定了一个决策系统为 $S=(U,A,d)$。对每个连续属性 $a\in A$,记 C_a 为属性 a 的断点集合,并赋初值 $C_a=A$。$a(x_i)$ 为对象 x_i $(i=1,2,\cdots,n)$ 所对应的属性 a 的值,$d(x_i)$ 为对象 x_i 所对应的决策属性 d 的值。

FSBDA 算法具体描述如下：

(1) 首先假设条件属性 a 是这次所求的断点属性对象,一开始设置一个存储集合 A 来存储数据表中 $x_1,x_2,\cdots,x_N$ 的 $a(x_i)$ 值,并从小到大排列 A 中的值。根据 $a(x_i)$ 值的不同确定一组初始划分集合 Co,由小到大分别存储,即每个 $a(x_i)$ 值对应一组划分集合 Co。

(2) 初始划分 Co 中的每个集合对应一个支持度 $sup(a(x))$,在满足 $a(x_i)\neq a(x_{i+1})$ 且 $d(x_i)\neq d(x_{i+1})$ 的条件下,通过公式 (1) 得到一个断点 C_a^i。若是满足 $a(x_i)\neq a(x_{i+1})$ 但 $d(x_i)=d(x_{i+1})$ 时,则按照前面方法继续考虑 $a(x_i)$ 与 $a(x_{i+2})$ …的关系,直到产生一个断点为止。

(3) 根据频数监督断点定义,从上到下扫描 a 的论域寻找可能的断点,得断点集合 $V_a=\{c_a^1,c_a^2,c_a^3,\cdots,c_a^k\}$,并把原有的数据表分成 $K+1$ 个划分，对每个划分统计中该划分内出现频数最多的某个 $a(x_i)$ 值,并将该划分内的其他值用该 $a(x_i)$ 值进行替换。

(4) 在断点集 V_a 上进行简约处理,最终得到关于条件属性 a 的新值 $Val=\{a'(x_1),a'(x_2),\cdots,a'(x_N)\}$和 $V'_a=\{c_a^{1'},c_a^{2'},c_a^{3'},\cdots,c_a^{k'}\}$最终断点集合。

3　算法实现及比较

3.1　算法测试

本实验以文献[6]给出的汽轮机故障诊断方面的实例,其中 U为事务的编号,$S1$~$S11$ 表示的是过程征兆,D 则表示的是故障的有无。这里采用的配置是 Intel 公司的 Pentium ® 4 处理器,主频 3.0 GHz,512 MB 内存,系统是 WindowsXP 环境,编程环境是 VS2005。

以其中的一个属性 $S1$ 为例。由于 $S1$ 在全域里面存在多个不同值 (0.2,0.3,0.4,0.5,0.6,0.7,0.8)。根据所提出的频数监督断点的定义,计算得到的 $S1$ 频数监督断点集合为：(0.333,0.45,0.53,0.66,0.75),即将 $S1$ 初始划分成有 6 个集合。在完成 MFA 算法步骤 (3)、(4) 后,属性 $S1$ 最后被离散化为 5 个划分。结果如表 1 所示。

表 1　数据表属性 $S1$ 断点选取结果

事务编号	$S1$
11,12,10,16,17	0.3
9,13,14,15	0.4
3,5,6,7	0.5
8,18,19,20,21	0.7
1,2,4	0.8

3.2　算法比较与结果讨论

为了比较,选取 5 种典型的离散化算法,对每个算法在相同实验环境和用同一样本进行测试,响应时间如表 2 所示。

表 2　$S1$ 上各种离散算法响应时间

算法集	FSBDA	等距划分 (k=5)	等频划分 (k=5)	Naïve Scaler	频数断点法	信息熵中值
响应时间/ms	236	97	99	141	187	307

从表 2 可以看出,等距和等频划分算法比其他算法的响应时间略好一些,这是因为它们是一种硬性划分,实质只是在集合上的操作,未带进知识性的信息。而 MFA 在噪声处理和边缘化的时候需要计算量较大,所以时间会稍微耗费多一些。

断点选择的合理性和断点个数是离散化算法的综合度量标准离散化算法,表 3 给出了 5 种典型算法对于实验样本中$S1$ 属性的离散化后得到的断点数量和分布情况。可以看出,等距和等频属无监督形式,使得断点分布过于平均,不符合实际情况,没有很好地反映数据实际的分布,且 k 值取值有待启发确定。从断点选取的合理性来说,等距和等频的效果不如 Naïve Scaler 和信息熵中值 ;Naïve Scaler 虽然考虑了决策值对断点划分的影响,但断点冗余较大 ;信息熵中值法属于有监督方式,但它是动态变化的,属性个数和域值的不同,使得断点产生上较趋于不稳定,且由于需要进行多个属性重要度计算,时间复杂度往往偏大。相比较而言,FSBDA 算法引入了类别信息 d 值,有监督地产生断点然后取舍，断点的选取较原先频数断点法更为合理。虽然 MFA 在效率没有明显优势,但同 Naïve Scaler 方法相比,不存在本质上劣势,且在断点检测结果更为合理、简练。

表3 S1各种离散算法产生断点数

算法	断点序号							断点个数
	1	2	3	4	5	6	7	
FSBDA	0.333	0.45	0.53	0.75				4
等距划分(k=5)	0.32	0.44	0.56	0.68				4
等频划分(k=5)	0.35	0.50	0.65	0.80				4
Naïve Scaler	0.30	0.35	0.45	0.55	0.60	0.65	0.75	7
频数断点法	0.26	0.357	0.45	0.537	0.66	0.75		6
信息熵中值	0.35	0.45	0.55	0.65	0.75			5

4 结束语

针对当前属性离散化方法中断点选取所存在的数量和选取合理性问题,提出一种基于频数监督断点的离散化算法FSBDA。该算法同现有基于频数断点的离散化方法相比,能较为高效地完成连续属性的离散化,所产生的断点不仅符合实际数据分布,而且更为合理、精练。通过实验测试验证了该算法的性能达到预期效果。下一步的工作将以FSBDA算法为基础,研究多属性离散化技术。

参考文献:

[1] 孙栋.粗糙集及其在数据挖掘中的应用研究[D].西安:西北大学,2006-08.

[2] Liu Huan,Hussain F,Tan C L,et al.Discretization:An enabling technique[J].Data Mining and knowledge Discovery,2002,6:393-423.

[3] Ziarko W,Yao Yi-gu.Rough sets and current trends in computing[C]//Lecture Notes in Computer Science.[S.l.]:Springer,2001.

[4] 梁吉业,曲开社,徐宗本.信息系统的属性约简[J].系统工程理论与实践,2001,21(12):76-80.

[5] 王国胤.Rough集理论与知识获取[M].西安:西安交通大学出版社,2001:24-31.

[6] Liu Xiao-yan,Wang Huai-qing.A discretization algorithm based on a heterogeneity criterion[J].IEEE Transactions on Knowledge and Data Engineering,2005,9(17):1166-1173.

[7] Chac-Ton S,Jyh-Hwa H.An extended Chi2 algorithm for discretization of real value attributes[J].IEEE Transactions on Knowledge and Data Engineering,2005,17(3):437-441.

[8] Kerber R.ChiMerge:discretization of numeric attributes[C]//Proceedings Ninth National Conference on Artificial Intelligence.[S.l.]:AAAI Press,1992:123-128.

[9] 胡逢彬,桂现才.基于相对熵的决策表连续属性离散化算法[J].计算机与信息技术,2006:39-41.

[10] Lenareik A,Piasta Z.Discretization of attributes space intelligent decision support[M].Kluuer:Roman Slowinski,1992:373-389.

[11] 桂现才.基于相对熵的一种属性约简算法[J].计算机工程与应用,2006,42(33):157-159.

[12] 阙夏.连续属性离散化方法研究[D].合肥:合肥工业大学,2006-06.

[13] 焦宁.连续属性离散化算法比较研究[D].合肥:合肥工业大学,2007-08.

(上接60页)

化归档算法δ-MOEA中。通过比较实验,发现δ-MOEA能很好地保持解集的分布性。

参考文献:

[1] Deb K.Multi-objective optimization using evolutionary algorithms[M].Chichester,UK:John Wiley & Sons,2001.

[2] 谢涛,陈火旺,康立山.多目标优化的演化算法[J].计算机学报,2003,26(8):997-1003.

[3] 郑金华.多目标进化算法及其应用[M].北京:科学出版社,2007.

[4] Deb K,Mohan M,Mishra S.A fast multi-objective evolutionary algorithm for finding well-spread pareto-optimal solutions,2003002[R].KanGAL Report,2003.

[5] Laumanns M,Thiele L,Deb K,et al.Combining convergence and diversity in evolutionary multi-objective optimization[J].Evolutionary Computation,2002,10(3):263-282.

[6] Coello Coello C A.Guest editorial:special issue on evolutionary multi-objective optimization[J].IEEE Transactions on Evolutionary Computation,2003,7(2):97-99.

[7] 周育人,闵华清,许孝元,等.多目标演化算法的收敛性研究[J].计算机学报,2004,27(10):1415-1421.

[8] Knowles J D,Corne D.Properties of an adaptive archiving algorithm for storing nondominated vectors[J].IEEE Transactions on Evolutionary Computation,2003,7(2):100-116.

[9] Deb K,Amrit P,Sameer A,et al.A fast and elitist multi-objective genetic algorithm:NSGA-II[J].IEEE Transactions on Evolutionary Computation,2002,6(2):182-197.

[10] Schott J R.Fault tolerant design using single and multicriteria genetic algorithm optimization[D].Aeronautics and Astronautics,Massachusetts Institute of Technology,Cambridge,1995-05.

[11] van Veldhuizen D A,Lamont G B.On measuring multiobjective evolutionary algorithm performance[C]//2000 Congress on Evolutionary Computation,2000,1:204-211.

[12] Schaffer J D.Multiple objective optimization with vector evaluated genetic algorithms[C]//Grefenstette:Proceedings of the First International Conference on Genetic Algorithms and Their Applications,1985:93-100.

[13] Poloni C.Multi-objective optimization by GAs:application to system and component design[C]//Methods in Applied Sciences'96:Invited Lectures and Special Technological Sessions of the Third ECCOMAS Computational Fluid Dynamics Conference and the Second ECCOMAS Conference on Numerical Methods in Engineering.Chichester:Wiley,1996:258-264.

[14] Fonseca C M,Fleming P J.An overview of evolutionary algorithms in multi-objective optimization[J].Evolutionary Computation,1995,3(1):1-16.

[15] van Veldhuizen D A.Multiobjective evolutionary algorithm:classifications,analysis,and new innovations[D].1999.

[16] Deb K.Multi-objective genetic algorithms:problem difficulties and construction of test problems,CI-49/98[R].Department of Computer Science/LS11,University of Dortmund,1998.

[17] Deb K,Thiele L,Laumanns M,et al.Scalable multi-objective optimization test problems[C]//Proceeding of the Congress on Evolutionary Computation(CEC2002),2002:825-830.

[18] Deb K.Multi-objective genetic algorithms:problem difficulties and construction of test problems[J].Evolutionary Computation,1999,7(3):205-230.