基于相容粗糙集技术的连续值属性决策树归纳

翟俊海[1,2] 翟梦尧[3] 李胜杰[1,2]
(河北大学数学与计算机学院 保定 071002)[1] (河北省机器学习与计算智能重点实验室 保定 071002)[2]
(河北大学工商学院 保定 071002)[3]

摘 要 决策树是常用的数据挖掘方法,扩展属性的选择是决策树归纳的核心问题。基于离散化方法的连续值决策树归纳在选择扩展属性时,需要度量每一个条件属性的每一个割点的分类不确定性,并通过这些割点的不确定性选择扩展属性,其计算时间复杂度高。针对这一问题,提出了一种基于相容粗糙集技术的连续值属性决策树归纳方法。该方法首先利用相容粗糙集技术选择扩展属性,然后找出该属性的最优割点,分割样例集并递归地构建决策树。从理论上分析了该算法的计算时间复杂度,并在多个数据集上进行了实验。实验结果及对实验结果的统计分析均表明,提出的方法在计算复杂度和分类精度方面均优于其他相关方法。

关键词 相容粗糙集,决策树,扩展属性,割点,统计分析
中图法分类号 TP181 **文献标识码** A

Induction of Decision Tree for Continuous-valued Attributes Based on Tolerance Rough Sets Technique

ZHAI Jun-hai[1,2] ZHAI Meng-yao[3] LI Sheng-jie[1,2]
(College of Mathematics and Computer Science, Hebei University, Baoding 071002, China)[1]
(Key Lab. of Machine Learning and Computational Intelligence, Hebei University, Baoding 071002, China)[2]
(Industrial & Commercial College, Hebei University, Baoding 071002, China)[3]

Abstract Decision tree is a popular data mining method, and it is a crucial problem to select expanded attributes in the induction of decision tree. The uncertainty of each cut of each continuous-valued attributes needs to be measured during the selection of expanded attributes for induction of decision tree based on discretion method, and the computational time complexity is very high. In order to deal with this problem, a method of induction of decision tree for continuous-valued attributes based on tolerance rough sets technique was proposed. The method consists of three stages. First expanded attributes are selected with tolerance rough sets technique, and then the optimal cut of the expanded attribute is found, and the sample set is partitioned by the optimal cut, finally the decision tree can be generated recursively. We analysed the computational time complexity of the algorithm in theory and conducted some experiments on multiple database. The experimental results and the statistical analysis of the results demonstrate that the proposed method outperforms other related methods in terms of computational complexity and classification accuracy.

Keywords Tolerance rough sets, Decision trees, Expanded attributes, Cuts, Statistical analysis

1 引言

决策树[1]算法是一种典型的归纳学习算法,它采用贪心策略从训练样例中构造决策树。描述训练样例的条件属性大致分为3种:符号值属性、连续值属性和混合值属性。ID3算法[2]是处理符号值属性的著名决策树归纳学习算法,它是由Quinlan于1986年首先提出的。ID3算法以信息增益作为选择扩展属性(根结点)的标准,并递归地生成决策树。对于连续值属性,一般是基于离散化思想来构建决策树,具有代表性的工作包括Quinlan提出的C4.5算法[3]和Breima等人提出的分类回归树算法(Classification and Regression Tree,简称CART)[4]。C4.5算法是对ID3算法的改进,它以信息增益比率作为选择扩展属性的标准,既能处理离散值属性,又能处理连续值属性。另外,C4.5算法还克服了ID3算法倾向于选取取值较多的属性的缺点,对属性取值有缺失和错误的训练样例具有很好的健壮性。CART算法用Gini-Index作为扩展属性选取的标准[4]。Fayyad在文献[5]中对连续值属性决策树学习算法进行了详细的综述,并证明了分割信息熵极小化的分割点一定是在边界点处取到。

基于离散化思想的连续值决策树归纳学习算法要度量每

到稿日期:2012-01-26 返修日期:2012-08-22 本文受国家自然科学基金项目(61170040),河北省自然科学基金项目(F2010000323,F2011201063,F2012201023),河北大学自然科学基金项目(2011-228)资助。
翟俊海(1964—),男,博士,副教授,CCF会员,主要研究方向为机器学习与模式识别,E-mail:mczjh@hbu.cn;翟梦尧(1990—),女,主要研究方向为机器学习与模式识别;李胜杰(1985—),男,硕士生,主要研究方向为机器学习。

一个属性的每一个割点的不确定性，因此其计算复杂度高。王熙照等在文献[6]中引入了非平衡割点的概念，改进了Fayyad的工作，从理论上证明了分割信息熵极小的割点不仅在边界点取得，而且一定是非平衡割点。这样只需度量非平衡割点的不确定性，降低了算法的计算复杂度。赵慧琴[7]根据常用的决策树归纳学习算法属性选取标准所共有的特性，提出了一般广义熵函数的概念，在此基础上，提出了基于分割一般广义熵的连续值属性的决策树归纳方法，并证明了广义熵最小的割点也一定是非平衡割点。

对于一个给定的数据集，假设它包含 n 个样例，每个样例用 m 个属性描述。最坏情况下每个属性有 $n-1$ 个非平衡割点。基于非平衡割点的连续值属性决策树归纳算法的计算时间复杂度为 $O(m(n\log_2 n+n^2))$。本文提出了一种基于相容粗糙集技术的连续值属性决策树归纳算法，它可将计算时间复杂度降低为 $O(mn^2+n\log_2 n)$。粗糙集理论是一种完全数据驱动的粒计算方法，它从不同的粒层次刻画待研究的对象，能很好地刻画决策属性与条件属性之间的依赖关系。决策属性依赖某个条件属性的程度越高，说明该条件属性对决策越重要。所以，用决策属性对条件属性的依赖度作为启发式来选择决策树的扩展属性是有理论依据的。该算法首先利用相容粗糙集技术选择扩展属性，然后找出该属性的最优割点，分割样例集并递归地构建决策树。实验结果及对实验结果的统计分析均表明，本文提出的方法在计算复杂度和分类精度方面均优于其他相关方法。

2 基础知识

本节给出将要用到的相容粗糙集[8]和连续值属性决策树归纳的基础知识。

2.1 相容粗糙集

Skowron等[8]提出的相容粗糙集理论是经典粗糙集理论的推广，它用相容关系(或相似关系)代替不可分辨关系(或等价关系)，不仅可以发现属性值之间的相似性，还可以消除属性值之间的微小偏差，从而提高系统决策的鲁棒性，也可有效提高决策的效率。在相容粗糙集理论中，按相容关系划分得到的相容类一般不构成对论域的划分，而是构成了对论域的覆盖。

定义 1 四元组 $IS=(U,A\cup D,V,f)$ 称为决策信息系统，简称决策表。其中，$U=\{x_1,x_2,\cdots,x_n\}$ 为对象的非空有限集合，称为论域；$A=\{a_1,a_2,\cdots,a_m\}$ 是描述对象的属性集合；$D=\{d\}$ 是单决策属性(若 U 中的样例分为 p 类，则 $d\in\{1,2,\cdots,p\}$)；V 为属性值域，$f:U\times A\to V$ 为信息函数。

定义 2 给定决策表 $IS=(U,A\cup D,V,f)$，R 是定义在论域 U 上的二元关系，R 称为相容关系(或相似关系)，当且仅当 R 满足下面两个条件：

(1)自反性：$\forall x\in U$，有 xRx；

(2)对称性：$\forall x,y\in U$，若 xRy，则 yRx。

对于给定的决策表 $IS=(U,A\cup D,V,f)$，可以在论域 U 上定义多种相似关系 R 如[9-11]：

$$R_a(x,y)=1-\frac{|a(x)-a(y)|}{|a_{\max}-a_{\min}|} \tag{1}$$

$$R_a(x,y)=\exp\left(-\frac{(a(x)-a(y))^2}{2\sigma_a^2}\right) \tag{2}$$

$$R_a(x,y)=\max\left(\min\left(\frac{a(y)-(a(x)-\sigma_a)}{a(x)-(a(x)-\sigma_a)},\frac{(a(x)+\sigma_a)-a(y)}{(a(x)+\sigma_a)-a(x)},0\right)\right) \tag{3}$$

式中，$a\in A$，σ_a 是属性 a 的方差。$a(x)$ 和 $a(y)$ 分别是样例 x 和 y 在属性 a 上的取值。$a_{\max}\in V_a$ 和 $a_{\min}\in V_a$ 分别是属性 a 的最大值和最小值。此时，称 R_a 是由属性 a 诱导出的相似关系。对于 $\forall P\subseteq A$，可按如下方式定义属性子集 P 诱导出的相似关系 R_P[9-11]。

$$R_{P,\lambda}(x,y)=\frac{\sum_{a\in P}R_a(x,y)}{|P|}\geqslant\tau \tag{4}$$

或

$$R_{P,\lambda}(x,y)=\prod_{a\in P}R_a(x,y)\geqslant\tau \tag{5}$$

式中，$\tau\in[0,1]$ 是相似性阈值。

定义 3 给定决策表 $IS=(U,A\cup D,V,f)$，$P\subseteq A$，R_P 是由属性子集 P 诱导出的相似关系。对于 $\forall x\in U$，定义 x 的 R_P,τ 相似类(或相容类)$[x]_{R_{P,\tau}}$ 为

$$[x]_{R_{P,\tau}}=\{y\mid(y\in U)\wedge(xR_{P,\tau}y)\} \tag{6}$$

定义 4 给定决策表 $IS=(U,A\cup D,V,f)$，$P\subseteq A$，R_P 是由属性子集 P 诱导出的相似关系。$U/D=\{U_1,U_2,\cdots,U_p\}$ 是决策属性 D 对论域 U 的划分。定义 U_i 关于 R_P 的 τ 上近似和 τ 下近似分别为

$$\overline{R}_{P,\tau}(U_i)=\{x\mid(x\in U)\wedge([x]_{R_{P,\tau}}\cap U_i\neq\phi)\} \tag{7}$$

和

$$\underline{R}_{P,\tau}(U_i)=\{x\mid(x\in U)\wedge([x]_{R_{P,\tau}}\subseteq U_i)\} \tag{8}$$

称二元组 $(\overline{R}_{P,\tau}(U_i),\underline{R}_{P,\tau}(U_i))$ 为 τ 相容粗糙集，其中，$i=1,2,\cdots,p$。

定义 5 给定决策表 $IS=(U,A\cup D,V,f)$，$P\subseteq A$，R_P 是由属性子集 P 诱导出的相似关系。称

$$POS_{R_{P,\tau}}(D)=\bigcup_{U_i\in U/D}\underline{R}_{P,\tau}(U_i) \tag{9}$$

为 P 相对于 D 的 τ 正域。

定义 6 给定决策表 $IS=(U,A\cup D,V,f)$，$P\subseteq A$，R_P 是由属性子集 P 诱导出的相似关系。称

$$\gamma_{P,\tau}(D)=\frac{|POS_{R_{P,\tau}}(D)|}{|U|} \tag{10}$$

为 D 相对于 P 的 τ 依赖度。

2.2 基于信息熵的连续值属性决策树归纳

假设样例集中的每个样例用 m 个属性描述，基于信息熵的连续值属性决策树归纳方法分为三步。首先，对每个属性的每个非平衡割点计算其信息熵，信息熵最小的割点为该属性的局部最优割点。第二步，从 m 个局部最优割点中找出信息熵最小的全局最优割点，该全局最优割点所对应的属性被选为扩展属性。第三步，用全局最优割点分割样例集，递归地构建决策树，下面给出相关的基本概念及算法。

定义 7 给定决策表 $IS=(U,A\cup D,V,f)$，对样例 x_j $(1\leqslant j\leqslant n)$ 在属性 a_i $(1\leqslant i\leqslant m)$ 上的取值由小到大进行排序，为描述方便，假定排序后的样例依然用 x_j $(1\leqslant j\leqslant n)$ 表示。x_j 和 x_{j+1} 的中值称为属性 a_i 的割点，记为 c_{ji} $(1\leqslant j<n)$。若割点 x_j 和 x_{j+1} 具有不同的类别，则称 c_{ji} 为非平衡割点，否则称

c_{ji}为平衡割点。

定义 8 属性 a_i 的割点 c_{ji} $(1\leqslant j<n)$的信息熵定义为：

$$Entr(c_{ji})=\frac{|U_{c_{ji}}^{<}|}{|U|}Entr(U_{c_{ji}}^{<})+\frac{|U_{c_{ji}}^{>}|}{|U|}Entr(U_{c_{ji}}^{>}) \quad (11)$$

式中，$U_{c_{ji}}^{\leqslant}$表示由属性 a_i 的值小于 c_{ji} 的样例构成的集合；$U_{c_{ji}}^{>}$表示由属性 a_i 的值大于 c_{ji} 的样例构成的集合；$Entr(S)$表示集合 S 的熵。

定义 9 称满足如下条件的割点为属性 a_i 的最优割点，也称为局部最优割点。

$$c_{j^*i}=\underset{1\leqslant j<n}{\operatorname{argmin}}Entr(c_{ji}) \quad (12)$$

定义 10 称满足如下条件的割点为全局最优割点。

$$c_e=\underset{1\leqslant i\leqslant m}{\operatorname{argmin}}(c_{j^*i}) \quad (13)$$

基于信息熵的连续值属性决策树归纳方法描述如下。

算法 1 基于信息熵的连续值属性决策树归纳算法

输入：决策表 IS=(U,A∪D,V,f)

输出：决策树

Step1 For(每个属性 a_i $(1\leqslant i\leqslant m)$)

Step2 对样例 x_j $(1\leqslant j\leqslant n)$在属性 a_i上 的取值由小到大进行排序，为描述方便，假定排序后的样例依然用 x_j $(1\leqslant j\leqslant n)$表示。

Step3 For(每个割点 c_{ji} $(1\leqslant j<n)$)

Step4 按式(11)计算属性 a_i 的割点 c_{ji}的信息熵。

Step5 按式(12)计算属性 a_i 的局部最优割点 $c_{j^*i}=\underset{1\leqslant j<n}{\operatorname{argmin}}Entr(c_{ji})$。

Step6 按式(13)计算全局最优割点 $c_e=\underset{1\leqslant i\leqslant m}{\operatorname{argmin}}Entr(c_{j^*i})$，属性 a_e 为扩展属性。

Step7 用 c_e 分割样例集 U 为 $U_{c_e}^{\leqslant}$和 $U_{c_e}^{>}$，重复 Step1－Step6，递归地构建决策树。

显然，Step1 的计算时间复杂度为 $O(m)$，Step2 的计算时间复杂度为 $O(n\log_2 n)$，Step3－Step5 的计算时间复杂度均为 $O(n)$，Step6 的计算时间复杂度为 $O(m)$。因此，算法 1 的计算时间复杂度即为该算法的计算时间复杂度：$O(m(n\log_2 n+n^2))$。

3 基于相容粗糙集技术的连续值属性决策树归纳

本文提出的基于相容粗糙集技术的连续值属性决策树归纳方法分为三步。首先，利用式(10)计算每个条件属性相对于决策属性的相容依赖度，相容依赖度最大的属性被选为扩展属性，即决策树的根结点。第二步，利用式(11)计算扩展属性割点的信息熵，信息熵最小的割点为最优割点。第三步，用最优割点分割样例集，递归地构建决策树。下面给出具体的算法。

算法 2 基于相容粗糙集技术的连续值决策树归纳算法

输入：决策表 IS=(U,A∪D,V,f)，相似性阈值 τ

输出：决策树

//e 依赖度最大的属性下标，CurD 当前最大依赖度值

Step1 For(每个属性 a_i $(1\leqslant i\leqslant m)$)

Step2 For(每个样例 x_j $(1\leqslant j\leqslant n)$)

Step3 按式(6)计算样例 x_j 关于属性 a_i 的 τ 相似类$[x_j]_{a_i,\tau}$。

Step4 按式(9)计算属性 a_i 的 d 正域 $POS_{a_i,\tau}(d)$。

Step5 按式(10)计算属性 a_i 的重要度 $\gamma_{a_i,\tau}(d)$。

Step6 If(属性 a_i 的重要度>CurD)

Step7 CurD=属性 a_i 的重要度

Step8 e=i

Step9 以 a_e 为扩展属性，对样例 x_j $(1\leqslant j\leqslant n)$在属性 a_e 上的取值由小到大进行排序，为描述方便，假定排序后的样例依然用 x_j $(1\leqslant j\leqslant n)$表示。

Step10 按式(11)计算属性 a_e 的割点的信息熵 $Entr(c_{je})$ $(1\leqslant j<n)$。

Step11 按式(12)计算属性 a_e 的最优割点 c_{j^*e}。

Step12 用 c_{j^*e} 分割样例集 U 为 $U_{c_{j^*e}}^{\leqslant}$ 和 $U_{c_{j^*e}}^{>}$，重复 Step1－Step11，递归地构建决策树。

显然，Step1 的计算时间复杂度为 $O(m)$，Step2 和 Step3 的计算时间复杂度均为 $O(n)$，Step4－Step8 的计算时间复杂度均为 $O(1)$，Step9 的计算时间复杂度为 $O(n\log_2 n)$，Step10－Step11 的计算时间复杂度均为 $O(n)$。因此，算法 2 的计算时间复杂度为 $O(mn^2+n\log_2 n)$。

4 实验结果及统计分析

为了进一步验证本文算法的有效性，我们在 9 个 UCI 数据集[12]上进行了两组实验，并对实验结果进行了统计分析。实验所用的 9 个 UCI 数据集分别是 iris，Cancer，Heart，Pima，Wine，Diabetes，Vehicle，Ionosphere 和 Hepatitis。实验采用十折交叉验证的方法，实验环境是 PC 机，双核 1.86G CPU，2G 内存，Windows XP 操作系统，Matlab 7.1 实验平台。

实验 1 与 C4.5 算法和文献[7]的算法相比较

实验中将本文提出的算法与 C4.5 算法和文献[7]中的算法进行了比较。首先对属性值进行归一化处理，用式(1)计算两个样例关于条件属性的相似性，实验结果见表 1。从表 1 可以看出，本文提出的方法在 7 个数据集上的测试精度高于 C4.5 和文献[7]的方法。在 8 个数据集上的所用时间低于 C4.5 和文献[7]的方法。因此与 C4.5 和文献[7]的方法相比，本文提出的方法更有效。

表 1 3 种决策树归纳方法的实验结果

数据集	文献[7]的算法		本文算法		C4.5 算法	
	Accuracy (%)	Time (s)	Accuracy (%)	Time (s)	Accuracy (%)	Time (s)
Iris	96.55	0.144	95.33	**0.097**	96.00	0.111
Cancer	94.22	2.580	**94.21**	**2.213**	93.86	2.686
Heart	74.07	0.425	**74.32**	0.416	73.55	**0.404**
Pima	70.46	1.202	**72.43**	**1.023**	71.69	1.303
Wine	91.56	0.678	**93.89**	**0.673**	91.94	0.687
Diabetes	73.22	0.698	73.51	**0.574**	**73.64**	0.932
Vehicle	67.88	0.537	**68.24**	**0.423**	46.39	0.517
Ionosphere	85.46	1.247	**86.33**	**1.032**	85.61	1.646
Hepatitis	80.12	0.239	**81.58**	**0.201**	78.44	0.227

实验 2 相似性参数对测试精度的影响

在这个实验中，我们探讨相似性参数 τ 对测试精度的影响，并对参数的取值给出一个实验性的指导。为清楚起见，将测试精度随参数 τ 变化的曲线显示在两个图中。图 1 给出了在 iris，Cancer，Wine，Ionosphere 和 Hepatitis 5 个数据集上测试精度随参数 τ 变化的曲线。图 2 给出了在 Pima，Diabetes，Vehicle 和 Heart 4 个数据集上测试精度随参数 τ 变化的曲线。从图 1 可以看出，对 iris，Cancer 和 Wine 3 个数据集，参数 τ 取[0.90，1.00]区间中的值比较合适。对 Ionosphere 数据集，参数 τ 取[0.85，1.00]区间中的值比较合适。而对 Hepatitis 数据集，参数 τ 取[0.85，0.89]区间中的值比较合

适。从图 2 可以看出，对 Diabetes 和 Heart 这两个数据集，参数 τ 取[0.90,1.00]区间中的值比较合适。而对 Pima 和 Vehicle 这两个数据集，参数 τ 取[0.94,1.00]区间中的值比较合适。实验 1 中参数 τ 的取值就是依据这些原则确定的。

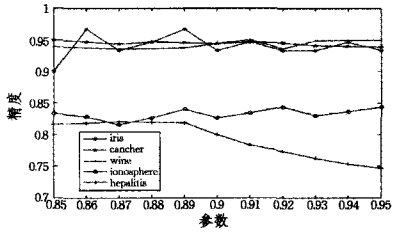

图 1　在 iris 等 5 个数据集上测试精度与相似性参数 τ 之间的关系

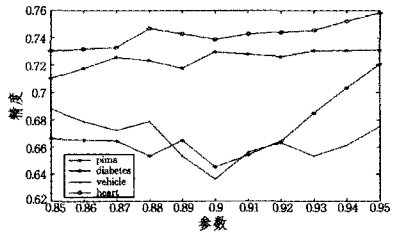

图 2　在 pima 等 4 个数据集上测试精度与相似性参数 τ 之间的关系

另外，为了更进一步验证本文所提算法的有效性，用 Wilcoxon 检验(简称 W-检验)和成对 T 检验(简称 T-检验)对实验结果进行了统计分析[13]。首先，对文献[7]的算法和本文算法在每个数据集上运行十折交叉验证 10 次，对每次十折交叉验证的结果求平均，得到两个 10 维的统计量 $X_i(i=1,2)$，分别对应文献[7]的算法和本文算法。然后，通过计算 MATLAB 函数 *ranksum*(X_1, X_2)进行 Wilcoxon 检验，再通过计算 MATLAB 函数 *ttest2*(X_1, X_2)进行成对 T 检验。结果见表 2，从表 2 的 P 值和 H 值可以看出，本文算法从统计意义上优于文献[7]中的算法。

表 2　文献[7]中的算法与本文算法实验结果的统计分析

数据集	W-检验		T-检验
	P 值	H 值	H 值
Iris	1.40e-003	1	1.23e-005
Wine	1.34e-003	1	2.91e-003
Pima	4.91e-003	1	4.29e-003
Diabetes	3.20e-004	1	6.30e-004
Vehicle	7.65e-004	1	1.84e-004
Ionosphere	3.09e-003	1	1.99e-003
Hepatitis	1.26e-003	1	1.05e-003
Cacher	2.80e-004	1	1.80e-004
heart	2.65e-003	1	1.86e-003

表 3　C4.5 算法与本文算法实验结果的统计分析

数据集	W-检验		T-检验
	P 值	H 值	H 值
Iris	1.18e-003	0	2.30e-003
Wine	2.74e-003	1	1.10e-002
Pima	1.00e-004	1	9.91e-005
Diabetes	7.53e-003	1	2.11e-003
Vehicle	5.80e-004	1	2.96e-005
Ionosphere	5.70e-003	1	2.70e-004
Hepatitis	4.60e-003	1	1.90e-004
Cacher	3.75e-003	1	2.11e-003
heart	6.30e-004	1	5.90e-004

对 C4.5 算法和本文算法进行了类似的统计分析，结果见表 3，从而进一步证实了本文算法的有效性。

结束语　扩展属性的选择标准是决策树归纳学习的核心问题。基于离散化方法构建连续值属性决策树需要度量每一个属性的每一个非平衡割点的不确定性，计算时间复杂度高。针对这一问题，本文提出了一种基于相容粗糙集技术的连续值属性决策树归纳方法。该方法首先用相容粗糙集技术找出最优属性，然后计算该属性的最优非平衡割点，递归地构建决策树。实验结果及对实验结果的统计分析均表明，本文提出的方法在测试精度和运行时间上均优于其他相关方法。

参 考 文 献

[1] Mitchell T M. Machine Learning [M]. 北京：机械工业出版社，2003:55-73

[2] Quinlan J R. Induction of Decision Tree [J]. Machine Learning, 1986,1(1):81-106

[3] Wu X D, Kumar V, Quinlan J R, et al. Top 10 algorithms in data mining [J]. Knowledge and Information Systems, 2008, 14(1): 1-37

[4] Breiman L, Friedman J H, Olshen R A, et al. Classification and Regression Tree [M]. Wadsworth International Group, 1984

[5] Fayyad U M, Irani K B. On the handling of continuous-valued attributes in decision tree generation [J]. Machine Learning, 1992,8(1):87-102

[6] 王熙照，洪家荣. 区间值属性决策树学习算法 [J]. 软件学报，1998,9(8):637-640

[7] Wang Xi-zhao, Zhao Hui-qin, Wang Shuo. The Study of Unstable Cut-point Decision Tree Generation Based-on the Partition Impurity [C]// Proceeding of 2009 International Conference on Machine Learning and Cybernetic. Baoding, China, 2009, 4: 1891-1897

[8] Skowron A. Tolerance approximation spaces [J]. Fundamenta Informaticae, 1996, 27(2/3): 245-253

[9] Macparthalain, Shen Q. On rough sets, their recent extensions and applications [J]. The Knowledge Engineering Review, 2010, 25(4): 365-395

[10] Parthalain N, Shen Q, Jensen R. A distance measure approach to exploring the rough set boundary region for attribute reduction [J]. IEEE Transactions on Knowledge and Data Engineering, 2010, 22(3): 305-317

[11] Parthalain N, Shen Q. Exploring the boundary region of tolerance rough sets for feature selection [J]. Pattern Recognition, 2009, 42(5): 655-667

[12] Blake C L, Merz C J. UCI Repository of Machine Learning Databases. http://www.ics.uci.edu/~mlearn/MLRepository.html, 2011

[13] Janez D. Statistical Comparisons of Classifiers over Multiple Data Sets [J]. Journal of Machine Learning Research, 2006, 7(1): 1-30