维普资讯 http://www.cqvip.com

差别矩阵浓缩及其属性约简求解方法*)

杨 明[1,2] 杨 萍[1]

(南京师范大学数学与计算机学院 南京 210097)[1]

(南京航空航天大学计算机科学与工程系 南京 210016)[2]

摘 要 属性约简是粗糙集理论的重要研究内容之一,已出现大量的属性约简算法,其中基于差别矩阵的属性约简算法是高效属性约简算法之一,但这些算法主要针对一致决策表,而对于不一致决策表,某些情况下不能得到属性约简。为此,本文提出改进的差别矩阵及其属性约简求解方法,统一考虑决策表一致和不一致情况两种情况下的属性约简,有效改进经典的基于差别矩阵求解属性约简的不足。同时,为适应大数据集属性约简需要,提出一种新的差别矩阵浓缩策略,以此提高属性约简的效率。

关键词 粗糙集,差别矩阵,核,属性约简

Discernibility Matrix Enriching and Computation for Attributes Reduction

YANG Ming[1,2] YANG Ping[1]

(School of Mathematics and Computer Science, Nanjing Normal University, Nanjing 210097)[1]

(Department of Computer Science & Engineering, Nanjing University of Aeronautics & Astronautics, Nanjing 210016)[2]

Abstract Attributes reduction is one of important parts researched in rough set theory. Thus, many algorithms have been proposed for attributes reduction, in which the algorithms based on discernibility matrix is one of efficiently attributes reduction algorithms. Unfortunately, these algorithms based on discernibility matrix mainly aim at the consistent decision table, and can not get a correct result for an inconsistent decision table in some cases. Therefore, in this paper, we introduce improved discernibility matrix for computing attributes reduction, which gives an unified framework for a consistent or inconsistent decision table, and efficiently improves the drawback of the existing attributes reduction algorithm based on discerniblity matrix. At the same time, a novel method of improved discernibility matrix enriching is proposed for attributes reduction of a very large dataset.

Keywords Rough set, Discernibility matrix, Core, Attributes reduction

1 引言

Rough Set(RS,粗糙集)是一种新的处理不精确、不完全与不相容知识的数学理论[1],近年来该理论在机器学习、数据挖掘及模式识别等多个领域得到了广泛的应用[2,3]。在粗糙集理论中,核和属性约简是其重要研究内容,也是知识获取的关键步骤,因此核[4~7]和属性约简[8~16]研究备受粗集研究者的关注,也取得了很大的进展。现有的属性约简大体上可分为基于差别矩阵或在此上改进的属性约简算法[8~12]、基于正区域的属性约简算法[13,14]及基于启发式的属性约简算法[15,16]三种;其中基于差别矩阵的方法因其简明高效,受到广泛的关注。

经典的基于差别矩阵的求解属性约简算法主要分为两种:一种是首先依据差别矩阵求出核,然后依据核逐步扩展重要属性,直到求出一个属性约简为止[8~10];另一种是依据差别矩阵得到一个差别函数,通过对该差别函数进行化简得到一个析取的合取表达式,然后得到属性约简[11,12]。但这两种属性约简求解算法主要针对一致决策表,而对于不一致决策表,某些情况下不能得到属性约简。为此,本文提出改进的差别矩阵及其属性约简求解方法,统一考虑决策表一致和不一致情况两种情况下的属性约简,有效改进经典的基于差别矩阵求解属性约简的不足。同时,为适应大数据集属性约简需要,提出一种新的差别矩阵浓缩策略,以此提高属性约简的效率。

2 粗糙集的基本概念

粗糙集理论的要点是将分类与知识联系在一起,并用等价类关系形式化表示分类。可理解为:知识是使用等价类 R 对离散空间 U 的划分,记为 $U/R=\{X_1,X_2,\cdots,X_n\}$,称为 X_i 为 U/R 的等价类。为节省篇幅,仅介绍和属性约简及核有关的一些概念,关于粗糙集的其他一些概念可参见文[2,3]。

决策表 DT 是一个四元组 $\langle U,C\cup D,V,f\rangle$,其中,$U$ 是一组对象的非空有限集合,称为论域;设有 n 个对象,则 U 可表示为:$U=\{x_1,x_2,\cdots,x_n\}$,C 为条件属性,D 为决策属性,$V=\bigcup_{a\in(C\cup D)}V_a$,$V_a$ 为属性 a 的值域集;f 是 $U\times(C\cup D)\to V$ 的映射。对 $B\subseteq(C\cup D)$,无差别关系 IND(B)定义为 $\{(x,y)\in U^2 \mid \forall a\in B, f(x,a)=f(y,a)\}$,通过 IND(B)将 U 划分为若干个类 $E_i(1\leqslant i\leqslant|U/\mathrm{IND}(B)|)$。为便于叙述,设条件属性集合 C 中有 m 个属性:$C_1,C_2,C_3,\cdots,C_m$,其值域为有限离散集

*)本课题得到江苏省自然科学基金(BK2005135)、江苏省博士后科研资助计划(05225)资助及江苏省高校自然科学研究项目基金(05KJB520066)。杨 明 博士、教授,主要研究方向为数据挖掘、机器学习、粗集理论与应用。

维普资讯 http://www.cqvip.com

合，并用$|\cdot|$表示集合的基。不失一般性，假设仅有一个决策属性D，其取值范围是$1,2,\cdots,k$。由D导出的等价类构成U的一个划分：$\{\psi_1,\psi_2,\cdots,\psi_k\}$，其中，$\psi_i=\{x\in U: f(x,D)=i\}$，$i=1,\cdots,k$。

在决策表DT中，若一些数据具有相同的条件属性而具有不同的分类，则称这类数据是不一致的，否则为一致的；若两个不同的对象x和y具有相同的条件属性而具有不同的分类，则称x和y为不一致的，否则称x和y为一致的。

定义1 设$X\subseteq U$为论域的一个子集，$P\subseteq C$，X的关于P的下近似为$\underline{P}X=\{x\in U:[x]_P\subseteq X\}$；其中，$[x]_P$表示$U$中所有与$x$在关系IND($P$)下是等价的元素构成的集合。

定义2 设$P\subseteq C$，对划分$\{\psi_1,\psi_2,\cdots,\psi_k\}$的$P$-近似精度为$\gamma_P=\sum_{i=1}^{k}|\underline{P}\psi_i|/|U|$。

定义3 设$P\subseteq C$，若$\gamma_P=\gamma_C$，且不存在$R\subset P$，使得$\gamma_R=\gamma_C$，则称P为C的一个(相对于决策属性D的)属性约简。称满足$\gamma_P=\gamma_C$的条件属性子集P为候选属性约简。所有C的属性约简的交称为C的核(简称核)，记为$Core(C)$。

定义4 如果属性$a\in C$满足$\gamma_{C-\{a\}}<\gamma_C$，则称属性$a$为不可缺少的(indispensable)，否则，称属性为冗余的。

3 经典的差别矩阵及其求属性约简方法的不足

为克服通过求出所有的不可缺少属性来确定核和属性约简方法的不足，一种简洁的利用差别矩阵来确定核方法被Hu学者提出，其中差别矩阵$M=\{m_{ij}\}$定义为：

$$m_{ij}=\begin{cases}\{a\in C: f(x_i,a)\neq f(x_j,a)\}，当\ f(x_i,D)\neq f(x_j,D)时\\ \phi(空集)，其他情况时\end{cases}\quad(1)$$

基于(1)式，文[9]和文[12]分别给出了两种高效的求解属性约简方法，为进一步说明这两种方法的原理，本文给出下面的两个实例。

实例1 表1所示的是一张数据表，其中共有7个元素和5个属性，$C=\{C_1,C_2,C_3,C_4\}$为条件属性集，D为决策属性。

表1 数据表

元素＼属性	C1	C2	C3	C4	D
x1	0	1	1	1	2
x2	2	0	0	1	1
x3	1	1	0	2	2
x4	1	0	2	0	1
x5	2	2	0	1	1
x6	2	1	1	1	2
x7	0	1	1	0	1

实例2 表2所示的是一张二值数据表，其中共有5个元素和4个属性，C={C1,C2,C3}为条件属性集，D为决策属性。

表2 数据表

元素＼属性	C1	C2	C3	C4	D
x1	1	0	1	0	1
x2	1	0	1	0	0
x3	0	0	1	0	1
x4	0	0	1	0	0
x5	1	1	1	0	1

对实例1，由式(1)可得差别矩阵$M1$(由于$M1$为对称矩阵，仅给出下三角阵)如下：

	x1	x2	x3	x4	x5	x6	x7
x1	ϕ						
x2	{C1,C2,C3}	ϕ					
x3	ϕ	{C1,C2,C4}	ϕ				
x4	{C1,C2,C3,C4}	ϕ	{C2,C3,C4}	ϕ			
x5	{C1,C2,C3}	ϕ	{C1,C2,C4}	ϕ	ϕ		
x6	ϕ	{C2,C3}	ϕ	{C1,C2,C3,C4}	{C2,C3}	ϕ	
x7	{C4}	ϕ	{C1,C3,C4}	ϕ	ϕ	{C1,C4}	ϕ

对$M1$，依据文[7]的结论(即差别矩阵$M1$中的所有单个属性构成的属性集为决策表的核)，可得表1的核为{C4}。由文[9]，从核出发，因$\gamma_{C-\{C1\}}=1$，$\gamma_{C-\{C2\}}=1$，$\gamma_{C-\{C3\}}=1$，选择C2进行扩展得{C2,C4}为一个属性约简。由文[12]，可得差别函数为$F(M1)=\{C4\}\wedge\{C2\vee C3\}=\{C4\wedge C2\}\vee\{C4\wedge C3\}$，即可得属性约简{C2,C4}和{C3,C4}。

对实例2，由式(1)可得差别矩阵$M2$(由于$M2$为对称矩阵，仅给出下三角阵)如下：

	x1	x2	x3	x4	x5
x1	ϕ				
x2	ϕ	ϕ			
x3	ϕ	{C1}	ϕ		
x4	{C1}	ϕ	ϕ	ϕ	
x5	ϕ	{C2}	ϕ	{C1,C2}	ϕ

对$M2$，依据文[7]的结论，将得到{C1,C2}为表2的核，这是一个错误的结论，文[4,5]纠正了该结论，并给出正确的求解方法。若依据$M2$得到的核$P=\{C1,C2\}$，由文[9]直接得P为一个属性约简，但P不是一个正确的属性约简，而是一个候选属性约简，但文[9]不会从P中删除任何属性，这是因为P中的属性全是核属性。而由文[12]，可得差别函数$F(M2)=\{C1\}\wedge\{C2\}=\{C1\wedge C2\}$，即可得属性约简{C1,C2}。但由定义2可知，$\gamma_{\{C2\}}=1/5=0.2$，而$\gamma_C=1/5=0.2$，因而可得{C2}为属性约简。

可见，对一致决策表，运用现有的基于差别矩阵的属性约简算法可得到属性约简；但对不一致决策表，在某些情况下不能得到核和属性约简。因此，改进差别矩阵使得快速求解属性约简是本文的主要目标。

4 改进的差别矩阵及其求属性约简方法

4.1 改进差别矩阵与属性约简方法

由定义2知，决策表的近似精度γ_C由其一致对象决定。于是，引入下面的定理1。

定理1 对$P\subset C$，若P为属性约简当且仅当$\bigcup_{i=1}^{k}\underline{P}\psi_i=\bigcup_{i=1}^{k}\underline{C}\psi_i$，且$\forall S\subset P$有$\bigcup_{i=1}^{k}\underline{S}\psi_i\subset\bigcup_{i=1}^{k}\underline{P}\psi_i$成立。

证明：略。

维普资讯 http://www.cqvip.com

由定理1可知，求解一个属性约简 P 的过程就是寻找使得所有一致对象在 P 上也是一致对象的过程，主要关注的是一致对象能否与其它对象区分开来，不必关心任意两个不一致对象之间是否能够区分。

实例2的求解之所以有误，其本质的原因是将两个不一致对象之间的区分引入到差别矩阵中，如：$M2$ 中蕴涵了不一致对象 $x3$ 与 $x2$，$x4$ 与 $x1$ 的区分。不难看出，若在 $M2$ 中消除不一致对象 $x3$ 与 $x2$，$x4$ 与 $x1$ 的区分对求解属性约简的影响，则可得正确的属性约简。

于是，借用文[4,5]求解核的差别矩阵定义，引入下面的改进差别矩阵。

定义5 对给定的决策表 DT，定义改进的差别矩阵 $IM=\{m_{ij}\}$ 为：

$$m_{ij}=\begin{cases}\{a\in C: f(x_i,a)\neq f(x_j,a)\}, & \text{当 } f(x_i,D)\neq f(x_j,D)\text{时，这里，}x_i\in U1,x_j\in U1\\ \{a\in C: f(x_i,a)\neq f(x_j,a)\}, & \text{其中，}x_i\in U1,x_j\in U2'\\ \phi(\text{空集}), & \text{其他情况时}\end{cases}\tag{1}$$

其中，$U1=\bigcup_{i=1}^{k}\underline{C}\psi_i(U)$，$U2=U-U1$，$U2'=delrep(U2)$，这里的函数 $delrep(U2)$ 描述如下：

```
Begin
U2'=φ;
for 任意 x∈U2 do
if 不存在 y∈U2'使得∀a∈C, f(x,a)=f(y,a)且 f(x,D)≠
   f(y,D) then ;
   U2'=U2'∪{x}
   return U2';
end.
```

性质1 对 $U1=\bigcup_{i=1}^{k}\underline{C}\psi_i(U)$，$U2=U-U1$，若 $U2\neq\phi$，则 $|U2'|<|U2|$。

定理2[5] 对于决策表 DT，若记 $IDM(C,IM)=\{m_{ij}\mid m_{ij}\in IM$ 且 m_{ij} 为单个属性$\}$，则有 $IDM(C,IM)=Core(C)$，即当且仅当某个 m_{ij} 为单个属性时，该属性属于核 $Core(C)$。

对一致决策表或不一致决策表，由定理2知，依据改进的差别矩阵可得到正确的核，因此一些现有基于差别矩阵且从核开始的属性约简算法可确保得到的属性约简是正确的。

为节约篇幅，从核出发的属性约简算法可参见文[9]，这里仅给出基于差别函数求解属性约简算法。依据文[3,11,12]介绍的差别函数求解属性约简原理，得到属性约简描述算法如下。

算法1 基于改进差别矩阵的属性约简算法

```
输入:U1=⋃_{i=1}^{k} C_ψi(U), U2=U-U1, U2'=delrep(U2)
输出:一个属性约简 P。
begin
(1) 由定义5建立差别矩阵 IM;
(2) set={m_ij | m_ij(m_ij≠φ)∈IM, 且¬∃m_kl(m_kl≠φ)∈IM 使得 m_kl
    ⊂m_ij};
(3) F=φ; //把 m 中的属性看成变量, m* 为这些变量的并
    for each m∈set do F=F∧m*;
(4) 化简 F 使得具有 m1∨m2∨…∨ms 形式，其中 mi(i=1,2,…,s)
    为属性变量的合取;
(5) 取出某个满意的 mj 作为属性约简;
end
```

对实例2，$U1=\{x5\}$，$U2=\{x1,x2,x3,x4\}$，由算法1的步骤1可得差别矩阵 IM 为：

	$x5$	$x1$	$x4$
$x5$	ϕ	$\{C2\}$	$\{C1,C2\}$

由步骤2可得 set=$\{\{C2\}\}$；由步骤3可得 F=C2，从而得一个属性约简$\{C2\}$。

可见，无论是一致决策表还是不一致决策表，应用改进差别矩阵求得的属性约简满足定义3的要求。

4.2 改进差别矩阵浓缩与属性约简求解

由4.1节可知，无论从核出发求解属性约简还是依据算法1，均须扫描整个差别矩阵，因而影响求属性约简的效率，不能适应大数据集属性约简的求解。因此，本节对算法1进行优化，对差别矩阵浓缩，以此降低存储开销。

由算法1可知，差别矩阵中只有少部分矩阵元素对求解属性约简起作用，若只保留这些矩阵元素将有效降低存储开销。为此，引入下面的定义6和定理3。

定义6 对给定差别矩阵 M，其浓缩差别矩阵定义为 $IME(M)=\{m\mid m(m\neq\phi)\in M$，且不存在 $m'(m'\neq\phi)\in M$ 使得 $m'\subset m\}$。

由定义6，对实例1，$IME(M1)=\{\{C2,C3\},\{C4\}\}$；对实例2，$IME(M2)=\{\{C1\},\{C2\}\}$，$IME(IM)=\{\{C2\}\}$。可见，由定义6可有效地对差别矩阵进行浓缩。

定理3 对决策表 DT，若改进差别矩阵为 IM，则有下列结论成立：

(1)由 IME(IM)可求得核 Core(C)；

(2)F(IME(IM))=F(IM)成立。

证明：(1)因 IME(IM)必定包含 IM 中所有仅含单个属性的矩阵元素，因而由定理2知由 IME(IM)可求得核 Core(C)。

(2)显然。证毕。

由定理3可知，经典的两种求解属性约简算法均可通过浓缩改进差别矩阵得到。于是，算法1的优化属性约简算法描述如下。

算法2 基于浓缩改进差别矩阵的属性约简算法

```
输入:U1=⋃_{i=1}^{k} C_ψi(U), U2=U-U1, U2'=delrep(U2)
输出:一个属性约简 P。
begin
(1) IME=φ;//直接求浓缩改进差别矩阵 IME
    for x_i∈U1 do
      for y_j∈(U1∪U2') do{
        生成 IM 中的元素 m_ij;
        if m_ij≠φ then
          if IME=φ then IME= m_ij; Else{
            删除 IME 中所有包含 m_ij 的元素;
            if IME 中不存在包含于 m_ij 的元素 then IME=IME∪
            m_ij;
            }}
(2)构造差别函数 F(IME); //类似算法1
   化简 F(IME)得到一个属性约简;
end
```

算法2无须存储整个改进差别矩阵，仅须存储一个浓缩改进差别矩阵，因而有效地降低了存储代价。对实例1而言，改进差别矩阵的矩阵元素个数为49，而下三角矩阵元素(不包含对角线元素)个数也有21；但按照算法2，浓缩改进差别矩阵的元素个数为2。可见，算法2可有效降低存储开销。

4.3 算法的复杂度分析

对 $U1=\bigcup_{i=1}^{k}\underline{C}\psi_i$，$U2=U-U1$，$U2'=delrep(U2)$，下面分别对算法1和算法2的空间复杂度和时间复杂度进行分析。

4.3.1 空间复杂度

若改进差别矩阵为 IM，其元素个数为($|U1|*(|U1|+|U2'|)$)，则由 $|U1|*(|U1|+|U2'|)\leqslant|U1|*|U|\leqslant|U|*|U|$ 知，改进差别矩阵一定程度上降低空间复杂度；而其浓缩改进差别矩阵 IME(IM)的个数 $|IME(IM)|\ll(|U1|*(|U1|$

(下转第269页)

维普资讯 http://www.cqvip.com

用眼睛拟合出一条直线。63.2%分位点是近似等于6.5个周，它是1/λ的估值，于是λ的估值是0.154。从顶部的原点出发画一条平行于拟合直线的直线，该直线与在左侧的辅助刻度直线的交点给出γ在图上的估值，它近似等于2.75。有了这些估值，P的寿命均值能用威布尔分布的均值公式估算出来：

$$\mu=\frac{\Gamma(1+1/\gamma)}{\lambda}$$

即$\mu=\Gamma(1.364)/0.154=0.8897/0.154=5.78$周。这里$\Gamma(\gamma)$是著名的Γ函数，定义为

$$\Gamma(\gamma)=\int_0^\infty x^{\gamma-1}e^{-x}\mathrm{d}x$$

注意，$\Gamma(x)$的值能在Abramowitz和Stegun(1964)的文章里找到。

最后，将实验环境下估算的P的寿命$\mu=5.78$周，用下式进行换算求出P的估计寿命：

$$L_P=\mu\times\frac{f_{adversary}}{f_{general}}$$

其中，$f_{adversary}$表示使用对手模式进行安全测试时对P发动攻击的频率；$f_{general}$表示一般网络状况下P遭受攻击的频率。

本次实验中，P每周遭受的测试小组发起的攻击频率为120次/周，根据网络安全状况调查报告的统计数据估计一般网络状况下P遭受攻击的频率约为5天/周×1次/天×(P受到攻击的可能性)50%=2.5次/周，可以计算出P的估计寿命年数为：

$$L_P=\mu\times\frac{f_{adversary}}{f_{general}}=5.78\times\frac{120}{2.5}=277.44(\text{周})\approx5.34(\text{年})$$

当然，在估算过程中我们用了一些网络统计结果和经验数据，上述实验用的对手是250个已知的攻击类型，对于未知的新的攻击类型无法加以测试，这些因素都可能导致最后估算出来的软件寿命数据不够精确。但是，由于应用软件的寿命本身就不是一个精确值，它不仅受软件本身抗攻击性能的影响，而且主要受到组织业务变动或机构调整的影响。上述估算结果与我们的经验数据拟合较好，可以作为软件购买组织进行该软件的投入—产出成本分析或者软件生产组织估算软件维护成本的一种新简易方法。

结论 在上述数据分析中，模型所发挥的作用应该随着讨论的问题以及针对问题所采用的方法的不同而不同，模型和有关统计方法的复杂性将随着应用的不同而有很大不同，一个应用软件很可能要受若干"失效"原因中任一个的影响，等等。所以，在涉及软件寿命的问题中，数据的产生过程经常是难于弄清楚的，这也是使问题复杂化的一个因素。也正是由于问题的复杂性，我们才不再划分各种网络攻击对应用软件寿命的不同影响，抽象到更高的层次，把所有攻击及其形形色色的新变种统统放到一个攻击环境里来进行研究。

另外，大多数安全bug的隐蔽本质使得应用软件需要特殊的、专门的安全测试。目前还没有算法(即使是探索性的)能够用来描述已知的各种攻击。新的攻击类型或变种不断被发现，人为地预测这些攻击的工作原理简直是不可能的。我们下一步的研究方向是应用机器学习的相关算法，研究从已知攻击特征预测出新攻击变种的方法，以提高软件对防范攻击的自学习能力和免疫能力。全方位的安全防护措施结合软件对防范攻击的自学习能力可能成为我们增强软件安全性的真正出路。

参考文献

1 Howard M, LeBlanc D. Writing Secure Code. Microsoft Press, 2002
2 Motwani R, Raghavan P. Randomized Algorithms. Cambridge University Press, 1995
3 McGraw G, Viega J. 软件安全十大原则. http://www.nap.edu/catalog/6161.html
4 Thompson H H, Chase S G. 红队程序安全测试. 软件研发, 2004, 2(7)
5 Whittaker J, Thompson H H. How to Break Software Security: Effective Techniques for Security Testing. Addison Wesly, 2003
6 Ross S M. Stochastic Processes. John Wiley, 1983
7 Foster I, Kesselman C. The Grid: Blueprint for a New Computing Infrastructure. Morgan Kaufmann, 1998, Chapter 3
8 Lawless J F. Statistical models and methods for lifetime data. John Wiley, 1983

(上接第183页)

+ |U2′|))，其中"≪"表示远远小于。

4.3.2 时间复杂度

对不一致决策表，若不一致对象较多，则算法1和算法2的时间复杂度低于经典的基于差别函数求解属性约简算法，算法的性能有所改进。

可见，算法2可有效地降低存储空间代价，一定程度提高求解属性约简的效率。当然，在实际应用中，用户可根据问题的规模，选择相应的属性约简算法。

结束语 本文提出改进的差别矩阵及其属性约简求解方法，统一考虑决策表一致和不致情况两种情况下的属性约简问题，克服经典的差别矩阵引起的求解属性约简算法不足。同时，提出一种新的差别矩阵浓缩策略，以此提高属性约简的效率，为进行大数据集属性约简提供一条有效的途径。

参考文献

1 Pawlak Z. Rough sets [J]. International Journal of Information and Computer Science, 1982, 11(5): 341～356
2 Pawlak Z. Rough set approach to multi-attribute decision analysis [J]. European Journal of Operational Research, 1994, 72: 443～459
3 刘清. Rough集及Rough推理[M]. 北京：科学出版社，2001
4 杨明，孙志挥. 改进的差别矩阵及其求核方法[J]. 复旦大学学报，2004，43(5)：865～868
5 杨明. 一种基于改进差别矩阵的核增量式更新算法[J]. 计算机学报，2006，29(3)：407～413
6 杨明，杨萍. 一种基于垂直分布的多决策表全局属性核求解算法[J]. 控制与决策，2006(录用待发)
7 Hu X H, Cercone N. Learning in relational databases: A rough set approach [J]. Computational Intelligence: An International Journal, 1995, 11(2): 323～338
8 Jelonek J, Krawiec K, Slowinski R. Rough set reduction of attributes and their domains for neural networks [J]. Computational Intelligence, 1995, 11(2): 339～347
9 叶东毅. Jelonek属性约简算法的一个改进[J]. 电子学报，2000，28(12)：81～82
10 Wang Jue, Wang Ju. Reduction algorithm based on discernibility matrix the ordered attributes method [J]. Journal of Computer Science and Technology, 2001, 16(6): 489～504
11 常犁云，王国胤，吴渝. 一种基于Rough Set理论的属性约简及规则提取方法[J]. 软件学报，1999，10(11)：1206～1211
12 Jensen R, Shen Qiang. Semantics-Preserving Dimensionality Reduction: Rough and Fuzzy-Rough-Based Approaches. IEEE Transactions on Knowledge and Engineering, 2004, 16(12): 1457～1471
13 刘少辉，盛秋戬，吴斌，等. Rough集高效算法的研究. 计算机学报，2003，26(5)：524～529
14 Guan J W, Bell D A. Rough computational methods for information systems [J]. Artificial Intelligences, 1998, 105 (1-2): 77～103
15 苗夺谦，胡桂荣. 知识约简的一种启发式算法. 计算机研究与发展，1998，36(6)：681～684
16 Woblewski J. Finding minimal reducts using genetic algorithm: [ICS Research Report]. 16/95. 1995. Warsaw University of Technology