不协调决策表中基于对象的近似约简

翟翠红 秦克云
(西南交通大学数学学院 成都 610031)

摘 要 对不协调决策表中基于对象的近似约简问题展开研究。首先,给出近似约简的判定定理;其次,通过区分矩阵与区分函数给出约简的计算方法;最后,通过实例验证,决策表中基于对象的近似约简与整体约简相比,不仅可以获得更简洁的知识,并且在实际生活中具有较好的应用价值。

关键词 决策表,近似约简,区分矩阵,区分函数
中图法分类号 TP18 **文献标识码** A

Approximate Reduction for Objects in Inconsistent Decision Tables

ZHAI Cui-hong QIN Ke-yun
(College of Mathematics, Southwest Jiaotong University, Chengdu 610031, China)

Abstract This paper studied the approximate reduction for the objects in the inconsistent decision tables. Firstly, the judgment theorems with respect to the approximate reduction were obtained. Secondly, the calculation method of the approximate reduction was given by discernibility matrices and functions. Finally, Compared to the overall reduction for a decision table, the reduction for objects in the decision table not only can get more concise knowledge, but also has good application value in real life.

Keywords Decision tables, Approximate reduction, Discernibility matrices, Discernibility functions

1 引言

粗糙集理论是一种处理不精确、不确定和模糊知识的数学工具[1,2],已被成功地应用于人工智能、数据挖掘、模式识别与智能信息处理[3]等领域,并越来越引起国内外学术界的关注。其主要思想就是在保持分类能力不变的前提下,通过知识约简导出问题的决策或分类规则[4]。通过知识约简,可以去掉不必要的属性,不仅使问题得到简化,而且更有利于把握问题的本质。

人们根据实际问题的背景,提出了多种整体属性约简方法,例如基于粗糙正域的约简[5]、基于包含度理论的约简、基于信息熵的约简等。整体约简在某些问题中得到了很好的应用,但它具有一定的局限性。很多情况下不能用整体约简删除的知识,可以通过对象约简进行化简从而获得更简洁的知识。2001年Kryszkiewicz在文献[6]中提出不协调决策表中基于对象的广义决策约简(generalized decision reduction)与近似约简(approximate reduction)的概念,并讨论了它们的基本性质,但并未给出具体的计算方法。可以证明基于对象的广义决策约简与文献[7]中的基于对象的相对约简是同一种约简。因此,本文对基于对象的近似约简问题展开研究,给出近似约简的判定定理,并通过区分矩阵与区分函数[8,9]给出约简的计算方法。最后通过实例展现基于对象约简与整体约简相比具有的优越性及在实际生活中的应用。

2 基本概念

定义 1[6] 一个信息系统是一个四元组 $S=(U,AT,V,f)$,其中

(1)U 是非空有限集合,称为论域,其中元素称为对象;

(2)AT 是非空有限集合,其中元素称为属性;

(3)$V=\bigcup_{a\in A} V_a$,V_a 是属性 a 的取值构成的集合,称为 a 的值域;

(4)$f:U\times AT\rightarrow V$ 称为信息函数,它为每个对象的每个属性赋予一个信息值,且对于任意 $x\in U,a\in AT$,有 $f(x,a)\in V_a$。

设 $S=(U,AT,V,f)$是信息系统。对于任意 $a\in AT$,由 a 可以确定论域 U 上的一个等价关系,称为由 a 确定不可区分关系,记为 $ind(a)$ 或 $SIM(\{a\})$,定义为:对于任意 $x,y\in U$,$(x,y)\in ind(a)$ 当且仅当 $f(x,a)=f(y,a)$。

因此,信息系统本质上就是数据库,每一个属性决定一个知识。类似地,对于任意 $A\subseteq AT$,由 A 确定不可区分关系 $ind(A)$ 为:对于任意 $x,y\in U$,$(x,y)\in ind(A)$ 当且仅当对于任意 $a\in A$,$f(x,a)=f(y,a)$。

信息系统的知识为分类 $U/ind(AT)=\{[x]_{ind(AT)}:x\in$

到稿日期:2013-05-10 返修日期:2013-08-25 本文受国家自然科学基金项目(61175044,61175055),中央高校基础研究基金项目(SWJTU11ZT29)资助。

翟翠红(1987—),女,硕士生,主要研究方向为粗糙集理论与方法,E-mail:865031165@163.com;秦克云(1962—),男,教授,博士生导师,主要研究方向为多值逻辑、粗糙集理论与方法。

$U\}$，其中概念的直观含义为：$[x]_{ind(AT)}:\bigwedge_{a\in AT}f(x,a)$。

定义 2 一个决策表是一个四元组 $DT=(U,A\cup\{d\},V,f)$，其中 U,A,V,f 的意义同定义 1，d 称为决策属性。

定义 3[8] $DT=\{U,AT\cup\{d\},V,f\}$ 是决策表，定义函数 $\partial_A:U\to P(V_d)$，$A\subseteq AT$，为 $\partial_A(x)=\{d(y)\mid y\in[x]_A\}$。称 ∂_A 为 DT 中的广义决策函数。如果 $\forall x\in U,|\partial_{AT}(x)|=1$，则 DT 是协调的（确定的）；否则它是不协调的（不确定的）。

对于决策表 $DT=(U,AT\cup\{d\},V,f)$，设 $\alpha_{AT}(x,y)$ 是满足 $(x,y)\notin SIM(\{a\})$ 的属性 $a\in AT$ 的集合。因此，如果 $(x,y)\in SIM(AT)$，则 $\alpha_{AT}(x,y)=\emptyset$。令 $\vee\alpha_{AT}(x,y)$ 是一个布尔表达式。如果 $\alpha_{AT}(x,y)=\emptyset$，则 $\vee\alpha_{AT}(x,y)=1$；否则 $\vee\alpha_{AT}(x,y)$ 是包含在 $\alpha_{AT}(x,y)$ 中的属性所对应变量的析取。

Δ 是信息系统的区分函数，若 $\Delta=\bigwedge_{(x,y)\in U\times U}\vee\alpha_{AT}(x,y)$。

$\Delta(x)$ 是信息系统中对象 x 的区分函数，若 $\Delta(x)=\bigwedge_{y\in U}\vee\alpha_{AT}(x,y)$。

Δ^* 是决策表 DT 的区分函数，若 $\Delta^*=\bigwedge_{(x,y)\in U\times\{z\in U|d(z)\notin\partial_{AT}(x)\}}\vee\alpha_{AT}(x,y)$。

$\Delta^*(x)$ 是决策表 DT 中对象 x 的区分函数，若 $\Delta^*(x)=\bigwedge_{y\in\{z\in U|d(z)\notin\partial_{AT}(x)\}}\vee\alpha_{AT}(x,y)$。

定义 4[6,8] 设 $DT=\{U,AT\cup\{d\},V,f\}$ 为一决策表，$A\subseteq AT,x\in U$，A 是关于 x 的一个相对（广义决策）约简，如果 A 是满足条件 $\partial_A(x)=\partial_{AT}(x)$ 的极小集。

定理 1[7] 设 $DT=\{U,AT\cup\{d\},V,f\}$ 为一决策表，$x\in U,A\subseteq AT$，则 $\Delta^*(x)$ 的极小析取范式中的所有合取子式恰为关于对象 x 的所有相对约简。

3 近似约简

定义 5[6] 设 $DT=\{U,AT\cup\{d\},V,f\}$ 为一决策表，$A\subseteq AT,x\in U$，A 是关于对象 x 的一个近似约简（approximate reduction），如果 A 是满足条件 $\forall y\in[x]_A,\partial_A(y)=\partial_{AT}(y)=\partial_{AT}(x)$ 的极小集。

定理 2 设 $DT=\{U,AT\cup\{d\},V,f\}$ 为一决策表，$A\subseteq AT,x\in U$，$\forall y\in[x]_A,\partial_A(y)=\partial_{AT}(y)=\partial_{AT}(x)$ 成立的充分必要条件为 $\forall y=\in[x]_A,\partial_A(y)=\partial_{AT}(y)$。

证明：（必要性）显然成立。

（充分性）若 $\forall y\in[x]_A,\partial_A(y)=\partial_{AT}(y)$ 成立，由 $x\in[x]_A$ 知，$\partial_A(x)=\partial_{AT}(x)$。又因为 $\forall y\in[x]_A,\partial_A(y)=\partial_A(x)$，故 $\forall y\in[x]_A,\partial_A(y)=\partial_{AT}(y)=\partial_{AT}(x)$ 成立。

由定理 2 知，$\forall y\in[x]_A,\partial_A(y)=\partial_{AT}(y)=\partial_{AT}(x)$ 与 $\forall y\in[x]_A,\partial_A(y)=\partial_{AT}(y)$ 等价，故关于对象 x 的近似约简有下述等价定义。

定义 5′ 设 $DT=\{U,AT\cup\{d\},V,f\}$ 为一决策表，$A\subseteq AT,x\in U$，A 是关于对象 x 的一个近似约简（approximate reduction），如果 A 是满足条件 $\forall y\in[x]_A,\partial_A(y)=\partial_{AT}(y)$ 的极小集。

由定理 1 知，$\forall x\in U,\partial_A(x)=\partial_{AT}(x)$ 与 $\forall u\in\{z\in U|d(z)\notin\partial_{AT}(x)\}$，当 $\alpha_{AT}(x,u)\neq\emptyset$ 时，$A\cap\alpha_{AT}(x,u)\neq\emptyset$ 等价，由此可得如下推论。

推论 1 设 $DT=\{U,AT\cup\{d\},V,f\}$ 为一决策表，$A\subseteq AT,x\in U,[x]_A=\{x_1,x_2,\cdots,x_n\}$，若 $\forall x_i\in[x]_A(1\leqslant i\leqslant n)$，$\forall u\in\{z\in U|d(z)\notin\partial_{AT}(x_i)\}$ 当 $\alpha_{AT}(x_i,u)\neq\emptyset$ 时，$A\cap\alpha_{AT}(x_i,u)\neq\emptyset$，则称 A 是基于对象 x 的近似协调集。若 A 是近似协调集，且 A 的任何真子集不是近似协调集，则称 A 是基于对象 x 的近似约简。

下面介绍计算基于对象的近似约简的方法。

由定理 1 我们知道，可以用区分函数 $\Delta^*(x)$ 求得对象 x 的所有相对约简 $A_1,A_2,\cdots,A_n$。

设 $[x]_{A_1}=\{x,x_{11},x_{12},\cdots,x_{1l_1}\}$，

$[x]_{A_2}=\{x,x_{21},x_{22},\cdots,x_{2l_2}\}$，

$\vdots \quad \vdots$

$[x]_{A_n}=\{x,x_{n1},x_{n2},\cdots,x_{nl_n}\}$。

记

$\Delta_1(x)=A_1\wedge\Delta^*(x_{11})\wedge\cdots\wedge\Delta^*(x_{1l_1})$，

$\Delta_2(x)=A_2\wedge\Delta^*(x_{21})\wedge\cdots\wedge\Delta^*(x_{2l_2})$，

$\vdots \quad \vdots$

$\Delta_n(x)=A_n\wedge\Delta^*(x_{n1})\wedge\cdots\wedge\Delta^*(x_{nl_n})$。

假设 $\Delta_1(x),\Delta_2(x),\cdots,\Delta_n(x)$ 的极小析取范式分别为 $\Delta_1^*(x),\Delta_2^*(x),\cdots,\Delta_n^*(x)$，令 $\overline{\Delta^*(x)}=\Delta_1^*(x)\vee\Delta_2^*(x)\vee\cdots\vee\Delta_n^*(x)$，这时可以得到下面的定理 3。

定理 3 设 $DT=\{U,AT\cup\{d\},V,f\}$ 为一决策表，$x\in U$，则 $\overline{\Delta^*(x)}$ 的极小析取范式中的所有合取子式恰为关于对象 x 的所有近似约简。

证明：设 A 是 $\overline{\Delta^*(x)}$ 的极小析取范式中的一个合取子式，则由推论 1 知要证 A 是关于对象 x 的近似约简，只需证明 A 是满足条件 $\forall x'\in[x]_A,\forall u\in\{z\in U|d(z)\notin\partial_{AT}(x')\}$，$\alpha_{AT}(x',u)\neq\emptyset$ 时，$A\cap\alpha_{AT}(x',u)\neq\emptyset$ 的极小集。

(1)因为 $\overline{\Delta^*(x)}=\bigvee_{i=1}^{n}\Delta_i^*(x)$，若 A 是 $\overline{\Delta^*(x)}$ 的极小析取范式中的一个合取子式，则存在 $1\leqslant i\leqslant n$，使得 A 是 $\Delta_i^*(x)$ 中的一个合取子式，且 $A_i\subseteq A$（因为 $\Delta_i(x)=A_i\wedge\Delta^*(x_{i1})\wedge\Delta^*(x_{i2})\wedge\cdots\wedge\Delta^*(x_{il_i})$）。下证 $\forall x'\in[x]_A,\forall u\in\{z\in U|d(z)\notin\partial_{AT}(x')\}$，当 $\alpha_{AT}(x',u)\neq\emptyset$ 时，$A\cap\alpha_{AT}(x',u)\neq\emptyset$。若存在 $x'\in[x]_A,u\in\{z\in U|d(z)\notin\partial_{AT}(x')\}$，使得 $\alpha_{AT}(x',u)\neq\emptyset$ 时 $A\cap\alpha_{AT}(x',u)=\emptyset$。令 A 中所有布尔变量取值为 1，其余为 0。由 A 是 $\Delta_i^*(x)$ 中的一个合取子式知 $\Delta_i^*(x)$ 的值为 1。注意到 $x'\in[x]_A\subseteq[x]_{A_i}$，故 $\alpha_{AT}(x',u)$ 的值为 0，即 $\Delta_i^*(x)$ 的值为 0，矛盾。

(2)A 是满足(1)的极小集。设 $A=\{b_1,\cdots,b_m\}$ 且 $\overline{\Delta^*(x)}$ 的极小析取范式为：$\theta_1=(b_1\wedge\cdots\wedge b_m)\vee(a_{11}\wedge\cdots\wedge a_{1m_1})\vee\cdots\vee(a_{k1}\wedge\cdots\wedge a_{km_k})$。

若存在 $A_1=\{b_{i1},\cdots,b_{ij}\}\subset A$ 满足 $\forall x'\in[x]_{A_1},\forall u\in\{z\in U|d(z)\notin\partial_{AT}(x')\}$，当 $\alpha_{AT}(x',u)\neq\emptyset$ 时，$A_1\cap\alpha_{AT}(x',u)\neq\emptyset$。令 $\theta_2=(b_{i1}\wedge\cdots\wedge b_{ij})\vee(a_{11}\wedge\cdots\wedge a_{1m_1})\vee\cdots\vee(a_{k1}\wedge\cdots\wedge a_{km_k})$，于是 $\theta_1\leqslant\theta_2$。若 $\theta_1=0$，则 $(b_1\wedge\cdots\wedge b_m)=(a_{11}\wedge\cdots\wedge a_{1m_1})=\cdots=(a_{k1}\wedge\cdots\wedge a_{km_k})=0$。

因 $\theta_1=0$，故存在 $x'\in[x]_A\subset[x]_{A_1},\forall u\in\{z\in U|d(z)\notin\partial_{AT}(x')\}$，使得 $\alpha_{AT}(x',u)\neq\emptyset$ 且 $\vee\alpha_{AT}(x',u)=0$（若 $\forall x'\in[x]_A,\vee\alpha_{AT}(x',u)\neq0$，则 $\theta_1\neq0$）。即 $\forall a\in\alpha_{AT}(x',u),v(a)=0$。又 $A_1\cap\alpha_{AT}(x',u)\neq\emptyset$，故 A_1 存在布尔变量取值为 0，

从而 $b_{i1}\wedge\cdots\wedge b_{ij}=0,\theta_2=0$。于是 $\theta_1=\theta_2$，此与 θ_1 的极小性矛盾。

由(1)、(2)知，A 是关于对象 x 的近似约简。

另一方面，若 A 是关于对象 x 的近似约简，则 $\forall x'\in[x]_A,\forall u\in\{z\in U|d(z)\notin\partial_{AT}(x')\},\alpha_{AT}(x',u)\neq\emptyset$ 时，$A\cap\alpha_{AT}(x',u)\neq\emptyset$。因为存在 $1\leqslant i\leqslant n$，使得 $A_i\subseteq A$ 即 $[x]_A\subseteq[x]_{A_i}$，故在极小析取范式 $\Delta_i^*(x)$ 中含有 A 中若干元素构成的合取子式，又 $\overline{\Delta^*(x)}=\bigvee_{i=1}^{n}\Delta_i^*(x)$，从而在 $\overline{\Delta^*(x)}$ 的极小析取范式中含有 A 中若干元素构成的合取子式，再由 A 的极小性，即知此合取子式恰为 A。

定理 3 给出了近似约简的判定定理。由判定定理 3，下面给出用区分函数与区分矩阵求决策表中基于对象的近似约简的例子。

例 1　利用区分函数与区分矩阵计算不协调决策表(表 1)中对象的近似约简。

表 1　不协调决策表

条件属性				决策属性
病人	头痛	肌肉痛	体温	流感
e_1	是	是	正常	否
e_2	是	是	高	是
e_3	是	是	很高	是
e_4	否	是	正常	否
e_5	否	否	高	否
e_6	否	是	很高	是
e_7	否	否	高	是
e_8	否	是	很高	否

上表为关于某些病人的决策表，其中 $U=\{e_1,e_2,\cdots,e_8\}$，$C=\{$头痛，肌肉痛，体温$\}$，$D=\{$流感$\}$。

令 $c_1=$头痛，$c_2=$肌肉痛，$c_3=$体温；$d_1=$是，$d_2=$否，则

$U/C=\{\{e_1\},\{e_2\},\{e_3\},\{e_4\},\{e_5,e_7\},\{e_6,e_8\}\}$

$U/D=\{\{e_2,e_3,e_6,e_7\},\{e_1,e_4,e_5,e_8\}\}$

表 2 中给出了 $\alpha_{AT}(x,y)$ 的值，其中 $x\in U,y\in\{z\in U|d(z)\notin\partial_{AT}(x)\}$。

表 2　区分矩阵

x\y	e_1	e_2	e_3	e_4	e_5	e_6	e_7	e_8
e_1		c_3	c_3			c_1 c_3	c_1 c_2 c_3	
e_2	c_3			c_1 c_3	c_1 c_2			c_1 c_3
e_3	c_3			c_1 c_3	c_1 c_2 c_3			c_1
e_4		c_1 c_3	c_1 c_3			c_3	c_2 c_3	
e_5								
e_6								
e_7								
e_8								

由

$$\Delta^*=\bigwedge_{(x,y)\in U\times\{z\in U|d(z)\notin\partial_{AT}(x)\}}\vee\alpha_{AT}(x,y),$$

$$\Delta^*(x)=\bigwedge_{y\in\{z\in U|d(z)\notin\partial_{AT}(x)\}}\vee\alpha_{AT}(x,y)$$

我们得到

$$\Delta^*=c_3(c_1\vee c_3)(c_1\vee c_2\vee c_3)(c_1\vee c_2)c_1(c_2\vee c_3)=c_1c_3$$

$$\Delta^*(e_1)=c_3(c_1\vee c_3)(c_1\vee c_2\vee c_3)=c_3$$

$$\Delta^*(e_2)=c_3(c_1\vee c_3)(c_1\vee c_2)=c_3(c_1\vee c_2)$$

$$\Delta^*(e_3)=c_3(c_1\vee c_3)(c_1\vee c_2\vee c_3)c_1=c_1c_3$$

$$\Delta^*(e_4)=(c_1\vee c_3)c_3(c_2\vee c_3)=c_3$$

$$\Delta^*(e_5)=\Delta^*(e_6)=\Delta^*(e_7)=\Delta^*(e_8)=1$$

上面已经得到 U 中所有对象的相对约简，下面用定理 3 的方法分别对每个对象求近似约简。

(1)基于对象 e_1 的相对约简为 $A_1=\{c_3\}$，由 $[e_1]_{A_1}=\{e_1,e_4\}$ 知 $\Delta_1^*(e_1)=A_1\wedge\Delta^*(e_4)=c_3\wedge c_3=c_3$，故 $\overline{\Delta^*(e_1)}=\Delta_1^*(e_1)=c_3$。因此基于对象 e_1 的近似约简为 $\{c_3\}$。

(2)基于对象 e_2 的相对约简有 $A_1=\{c_1,c_3\},A_2=\{c_2,c_3\}$。由 $[e_2]_{A_1}=\{e_2\},[e_2]_{A_2}=\{e_2\}$ 知 $\Delta_1^*(e_2)=A_1=c_1\wedge c_3$，所以 $\Delta_2^*(e_2)=A_2=c_2\wedge c_3$。故 $\overline{\Delta^*(e_2)}=\Delta_1^*(e_2)\vee\Delta_2^*(e_2)=(c_1\wedge c_3)\vee(c_2\wedge c_3)$。所以基于对象 e_2 的近似约简为 $\{c_1,c_3\},\{c_2,c_3\}$。

(3)基于对象 e_3 的相对约简有 $A_1=\{c_1,c_3\}$。由 $[e_3]_{A_1}=\{e_3\}$ 知 $\Delta_1^*(e_3)=A_1=c_1\wedge c_3$，故 $\overline{\Delta^*(e_3)}=\Delta_1^*(e_3)=c_1\wedge c_3$。所以对象 e_3 的近似约简为 $\{c_1,c_3\}$。

(4)基于对象 e_4 的相对约简有 $A_1=\{c_3\}$。由 $[e_4]_{A_1}=\{e_1,e_4\}$ 知 $\Delta_1^*(e_4)=A_1\wedge\Delta^*(e_1)=c_3\wedge c_3=c_3$，$\overline{\Delta^*(e_4)}=\Delta_1^*(e_4)=c_3$。因此基于对象 e_4 的近似约简为 $\{c_3\}$。

(5)基于对象 e_5 的相对约简有 $A_1=\{c_1\},A_2=\{c_2\},A_3=\{c_3\}$。由 $[e_5]_{A_1}=\{e_4,e_5,e_6,e_7,e_8\},[e_5]_{A_2}=\{e_5,e_7\},[e_5]_{A_3}=\{e_2,e_5,e_7\}$ 知

$$\Delta_1^*(e_5)=A_1\wedge\Delta^*(e_4)\wedge\Delta^*(e_6)\wedge\Delta^*(e_7)\wedge\Delta^*(e_8)=c_1\wedge c_3$$

$$\Delta_2^*(e_5)=A_2\wedge\Delta^*(e_7)=c_2$$

$$\Delta_3^*(e_5)=A_3\wedge\Delta^*(e_2)\wedge\Delta^*(e_7)=c_3\wedge(c_3\wedge(c_1\vee c_2))=c_3\wedge(c_1\vee c_2)$$

$$\overline{\Delta^*(e_5)}=\Delta_1^*(e_5)\vee\Delta_2^*(e_5)\vee\Delta_3^*(e_5)=(c_1\wedge c_3)\vee c_2\vee(c_3\wedge(c_1\vee c_2))=(c_1\wedge c_3)\vee c_2$$

故基于对象 e_5 的近似约简有 $\{c_2\},\{c_1,c_3\}$。

(6)基于对象 e_6 的相对约简有 $A_1=\{c_1\},A_2=\{c_2\},A_3=\{c_3\}$。由 $[e_6]_{A_1}=\{e_4,e_5,e_6,e_7,e_8\},[e_6]_{A_2}=\{e_1,e_2,e_3,e_4,e_6,e_8\},[e_6]_{A_3}=\{e_3,e_6,e_8\}$ 知

$$\Delta_1^*(e_6)=A_1\wedge\Delta^*(e_4)\wedge\Delta^*(e_5)\wedge\Delta^*(e_7)\wedge\Delta^*(e_8)=c_1\wedge c_3,$$

$$\Delta_2^*(e_6)=A_2\wedge\Delta^*(e_1)\wedge\Delta^*(e_2)\wedge\Delta^*(e_3)\wedge\Delta^*(e_4)\wedge\Delta^*(e_8)=c_2\wedge c_3\wedge(c_3\wedge(c_1\vee c_2))\wedge(c_1\wedge c_3)\wedge c_3=c_1\wedge c_2\wedge c_3$$

$$\Delta_3^*(e_6)=A_3\wedge\Delta^*(e_3)\wedge\Delta^*(e_8)=c_3\wedge(c_1\wedge c_3)=c_1\wedge c_3$$

$$\overline{\Delta^*(e_6)}=\Delta_1^*(e_6)\vee\Delta_2^*(e_6)\vee\Delta_3^*(e_6)=(c_1\wedge c_3)\vee(c_1\wedge c_2\wedge c_3)\vee(c_1\wedge c_3)=c_1\wedge c_3$$

故基于对象 e_6 的近似约简为 $\{c_1,c_3\}$。

(7)基于对象 e_7 的相对约简有 $A_1=\{c_1\},A_2=\{c_2\},A_3=\{c_3\}$。由 $[e_7]_{A_1}=\{e_4,e_5,e_6,e_7,e_8\},[e_7]_{A_2}=\{e_5,e_7\},[e_7]_{A_3}=\{e_2,e_5,e_7\}$ 知

$$\Delta_1^*(e_7)=A_1\wedge\Delta^*(e_4)\wedge\Delta^*(e_5)\wedge\Delta^*(e_6)\wedge\Delta^*(e_8)=c_1\wedge c_3$$

$\Delta_2^*(e_7)=A_2\wedge\Delta^*(e_5)=c_2$

$\Delta_3^*(e_7)=A_3\wedge\Delta^*(e_2)\wedge\Delta^*(e_5)=c_3\wedge(c_3\wedge(c_1\vee c_2))=c_3\wedge(c_1\vee c_2)$

$\overline{\Delta^*(e_7)}=\Delta_1^*(e_7)\vee\Delta_2^*(e_7)\vee\Delta_3^*(e_7)=(c_1\wedge c_3)\vee c_2\vee(c_3\wedge(c_1\vee c_2))=(c_1\wedge c_3)\vee c_2$

故基于对象 e_7 的近似约简有$\{c_1,c_3\}$,$\{c_2\}$。

(8)基于对象 e_8 的相对约简有 $A_1=\{c_1\}$,$A_2=\{c_2\}$,$A_3=\{c_3\}$。由$[e_8]_{A_1}=\{e_4,e_5,e_6,e_7,e_8\}$,$[e_8]_{A_2}=\{e_1,e_2,e_3,e_4,e_6,e_8\}$,$[e_8]_{A_3}=\{e_3,e_6,e_8\}$知

$\Delta_1^*(e_8)=A_1\wedge\Delta^*(e_4)\wedge\Delta^*(e_5)\wedge\Delta^*(e_6)\wedge\Delta^*(e_7)=c_1\wedge c_3$

$\Delta_2^*(e_8)=A_2\wedge\Delta^*(e_1)\wedge\Delta^*(e_2)\wedge\Delta^*(e_3)\wedge\Delta^*(e_4)\wedge\Delta^*(e_6)=c_2\wedge c_3\wedge(c_3\wedge(c_1\vee c_2))\wedge(c_1\wedge c_3)\wedge c_3=c_1\wedge c_2\wedge c_3$

$\Delta_3^*(e_8)=A_3\wedge\Delta^*(e_3)\wedge\Delta^*(e_6)=c_3\wedge(c_1\wedge c_3)=c_1\wedge c_3$

$\overline{\Delta^*(e_8)}=\Delta_1^*(e_8)\vee\Delta_2^*(e_8)\vee\Delta_3^*(e_8)=(c_1\wedge c_3)\vee(c_1\wedge c_2\wedge c_3)\vee(c_1\wedge c_3)=c_1\wedge c_3$

故基于对象 e_8 的近似约简有$\{c_1,c_3\}$。

由例1的计算结果容易看出,基于对象的近似约简结果不仅简洁,而且可以获得多种约简方式,打破了整体约简的局限性。同时,关于对象的近似约简具有很大的应用价值。以例1中的决策表 DT 为例,如果采取整体约简,则医生在诊断与下药时对 DT 中的每一个病例都要考虑头痛和体温两个病症。但如果采用对象约简便可以针对不同病人快速准确地诊断与下药,避免不必要的病症对医生的工作产生干扰,同时也可以减少对药品的浪费。

结束语 本文首先对粗糙集的基本理论知识进行介绍,针对不协调目标信息系统中基于对象的近似约简问题展开研究;给出近似约简的等价定义和判定定理,通过区分矩阵与区分函数给出近似约简的计算方法;并通过实例分析整体约简与基于对象的近似约简各自的优势及适用的场景。本文提出的基于区分矩阵与区分函数的算法简单易于理解,且能够找出任一对象的所有近似约简,但该方法时间复杂度为 $O(|A|^2|U|^2)$,其中$|A|$是条件属性个数,$|U|$是记录个数[10],故用于海量数据约简时效率偏低。因此我们下一步将对此方法进行改进,以扩大其使用范围。

参 考 文 献

[1] Pawlak Z. Rough sets[J]. International Journal of Computer and Information Science, 1982, 11: 341-356

[2] Pawlak Z. Rough Sets: Theoretical Aspects of Reasoning about Data[M]. Boston: Kluwer Academic Publishers, 1991

[3] Zhang X H, Zhou B, Li P. A general frame for intuitionistic fuzzy rough sets[J]. Information Sciences, 2012, 216: 34-49

[4] 张文修,梁怡,吴伟志. 信息系统与知识发现[M]. 北京:科学出版社,2003

[5] 杜卫锋,秦克云. 决策表正域约简区分函数条件的改进[J]. 计算机工程与应用,2006,42(20):16-18

[6] Kryszkiewicz M. Comparative study of alternative types of knowledge reduction in inconsistent systems[J]. International journal of intelligent systems, 2001, 16: 105-120

[7] 张文修,吴伟志,梁吉业,等. 粗糙集理论与方法[M]. 北京:科学出版社,2001

[8] Skowron A, Rauszer C. The discernibility matrices and functions in information systems. Intelligent Decision Support: Handbook of Applications and Advances to Rough Sets Theory[M]. Dordrecht: Kluwer Academic Publishers, 1992: 331-362

[9] Yao Y Y, Zhao Y. Discernibility matrix simplication for constructing attribute reducts[J]. Information Sciences, 2009, 179(7): 867-882

[10] 蒋云良,杨章显,刘勇. 不协调信息系统快速属性分布约简方法[J]. 自动化学报,2012,38(3):382-388

(上接第 231 页)

[2] Gandomi A H, Yang Xin-she, Alavi A H. Mixed variable structural optimization using Firefly Algorithm[J]. Computers & Structures, 2011, 89(23): 2325-2336

[3] Sayadi M K, Hafezalkotob A, Naini S G J. Firefly-inspired algorithm for discrete optimization problems: An application to manufacturing cell formation[J]. Journal of Manufacturing Systems, 2013, 32(1): 78-84

[4] Srivatsava P R, Mallikarjun B, Yang Xin-she. Optimal test sequence generation using firefly algorithm Original Research Article[J]. Swarm and Evolutionary Computation, 2013, 8(1): 44-53

[5] Kazem A, Sharifi E, Hussain F K, et al. Support vector regression with chaos-based firefly algorithm for stock market price forecasting[J]. Applied Soft Computing, 2013, 13(2): 947-958

[6] Chandrasekaran K, Simon S P. Network and reliability constrained unit commitment problem using binary real coded firefly algorithm[J]. International Journal of Electrical Power & Energy Systems, 2012, 43(1): 921-932

[7] dos Santos Coelho L, Mariani V C. Firefly algorithm approach based on chaotic Tinkerbell map applied to multivariable PID controller tuning[J]. Computers & Mathematics with Applications, 2012, 64(8): 2371-2382

[8] Horng M-H. Vector quantization using the firefly algorithm for image compression[J]. Expert Systems with Applications, 2012, 39(1): 1078-1091

[9] Senthilnath J, Omkar S N, Mani V. Clustering using firefly algorithm: Performance study[J]. Swarm and Evolutionary Computation, 2011, 1(3): 164-171

[10] Yang Xin-she, Deb S. Eagle strategy using lévy walk and firefly algorithms for stochastic optimization[J]. Studies in Computational Intelligence, 2010, 28(4): 101-111

[11] 刘长平,叶春明. 一种新颖的仿生群智能优化算法:萤火虫算法[J]. 计算机应用研究,2011,28(9):3295-3297

[12] Gandomi A H, Yang X-S, Talatahari S, et al. Firefly algorithm with chaos[J]. Communications in Nonlinear Science and Numerical Simulation, 2013, 18(1): 89-98