文章编号: 1001-0920 (2015) 04-0709-06 **DOI:** 10.13195/j.kzyjc.2014.1205

不一致决策表规则获取的粒计算方法

陈泽华, 张 裕, 谢 刚

(太原理工大学 信息工程学院，太原 030024)

摘 要: 针对不一致决策表的规则获取, 从属性多粒度角度考虑, 按粒度由粗到细将决策表划分成不同的粒度空间, 通过定义相容粒关系矩阵和不相容粒关系矩阵, 并充分挖掘隐含在矩阵中的启发式信息, 实现对不同粒度空间确定性规则和不确定性规则的获取. 最后, 从实例分析以及 UCI 测试对算法进行验证, 并与现有算法进行实验对比, 实验结果与分析表明了所提出算法的可行性和有效性, 而且按此方法获取的规则集的泛化能力更强.

关键词: 不一致决策表；规则获取；多粒度；粒计算

中图分类号: TP273 **文献标志码:** A

GrC method of rule acquisition for inconsistent decision table

CHEN Ze-hua, ZHANG Yu, XIE Gang

(College of Information Engineering，Taiyuan University of Technology，Taiyuan 030024，China．Correspondent：Chen Ze-hua，E-mail：zehuachen@163.com)

Abstract: To acquire the rules of inconsistent decision table, the inconsistent decision table is granulated into different granular spaces from fine to coarse in the perspective of attribute multi-granulation. By defining the consistent granular relation matrix and inconsistent granular relation matrix, as well as mining the heuristic information hidden in the matrices, the certain rules and uncertain rules in different granular space are acquired. Finally, the proposed algorithm is illustrated by an example and verified by UCI test set. The experiment comparison with existing algorithms is done to show the feasibility and effectiveness of the proposed algorithm, which also proves that the acquired rules have better generalized ability.

Keywords: inconsistent decision table；rule acquisition；multi-granulation；granular computing

0 引 言

随着大数据时代的到来, 丰富的数据与贫乏的知识之间的矛盾日益突出, 而实际中获取的数据往往具有不确定性并带有噪声, 研究这类数据的知识获取一直是数据挖掘领域的重要课题. 粗糙集理论[1]可以用确定的方法处理不确定信息, 规则获取[2]是基于粗糙集理论的知识发现的重要研究内容之一. 决策表一般可分为一致决策表和不一致决策表. 在对不一致决策表规则获取[3-6]的研究中, 张文修等[3]首先定义并比较了不协调目标信息系统中的几种不同知识约简, 为不协调目标信息系统的进一步研究指明了方向; 黄兵等[4]在文献 [3] 的基础上, 通过定义相应的决策矩阵并比较它们与条件属性矩阵的关系, 得到了提取系统所有规则的矩阵方法, 该方法虽然直观, 但获取的规则数量太大, 且规则之间存在重复; An 等[5]在研究粒计算方法的基础上提出了基于粒计算的规则获取算法, 算法中融入了在全粒空间中获取规则的思想, 根据每个条件属性形成原子知识粒子, 通过判断每个原子粒子是否属于决策粒来获取规则, 然而, 在全粒空间中获取规则, 算法的复杂性较高, 规则数量大且规则之间有冗余; 张清华等[6]对 An 等[5]的算法进行了改进, 提出基于最大粒的规则获取算法, 通过加入判断原子粒子是否是全粒空间中包含某个决策粒的最大粒子的条件, 从而获取更短且泛化能力更强的决策规则, 但算法仍然是在全粒空间中提取规则, 并没有降低算法的复杂性.

近年来, 粒计算[7]受到越来越多的学者的关注, 其中多粒度[8-12]的研究对粒计算理论及应用的发展起到了推动作用. 经典粗糙集是建立在单个二元关系基础之上的, 而多粒度粗糙集则是建立在多个二元关

收稿日期: 2014-08-03；**修回日期:** 2015-01-28.

基金项目: 国家自然科学基金项目(61402319)；山西省回国留学人员科研项目(2013-031).

作者简介: 陈泽华(1974−), 女, 副教授, 从事粒计算、智能信息处理等研究；张裕(1989−), 男, 硕士生, 从事粒计算、粗糙集理论及应用的研究.

系基础之上. Qian 等[8-10]提出了基于属性的多粒度粗糙集模型, 在此基础上, 重新定义了上下近似、度量精度, 证明了经典粗糙集中的一些性质是多粒度粗糙集的特例, 讨论了多粒度粗糙集的属性约简以及规则提取, 并发展了决策多粒度粗糙集; Yang 等[11]从多粒度角度研究模糊粗糙集, 并重点研究了乐观和悲观多粒度模糊粗糙集; Xu 等[12]提出了一般多粒度粗糙集, 并讨论了其主要性质.

本文研究粒计算方法, 借助多粒度思想, 从属性多粒度角度对决策表进行分析, 按粒度从粗到细将决策表划分成不同的粒度空间, 通过定义相容粒关系矩阵与不相容粒关系矩阵, 并利用启发式算子 RULH_1 和 RULH_2 加快实现不同粒度空间的规则获取. 通过设定规则是否覆盖论域为终止条件, 提高了算法收敛速度. 最后通过理论分析以及实验结果验证了所提出算法的可行性.

1 基本概念

定义 1 决策表[2]可以用如下的一个五元组表示:

$$S=(U,C,D,V,f).$$

其中: U 表示非空有限对象集, 称为论域; C 和 D 分别表示条件属性集和决策属性集, 同时 $C\bigcap D=\phi$; $V=\bigcup\limits_{a\in C\bigcup D}V_a$, V_a 是属性 a 的值域; $f:U\times C\bigcup D\to V$ 是信息函数, 它指定 U 中每个对象的属性值.

若 $B\subseteq C$, 则可以定义一个 U 上的不可区分关系

$$\mathrm{IND}(B)=\\ \{(x,y)\in U\times U|\forall b\in B,f(x,b)=f(y,b)\}.$$

R_C 和 R_D 分别表示由 $\mathrm{IND}(C)$ 和 $\mathrm{IND}(D)$ 导出的等价类, 若 $R_C\subseteq R_D$, 则称该决策表为一致决策表, 否则称为不一致决策表[2].

定义 2 在不一致决策表 $S=(U,C,D,V,f)$ 中, 假设 $U/D=\{D_1,D_2,\cdots,D_n\}$ 表示 U 在 D 下的决策类划分, $U/C=\{C_1,C_2,\cdots,C_m\}$ 表示 U 在 C 下的条件类划分, 称 $\mathrm{POS}_C(D)=\bigcup\limits_{D_i\in U/D}C_*(D_i)$ 为 C 相对于 D 的正域, $C_*(D_i)$ 表示在不可区分关系 $\mathrm{IND}(C)$ 下 D_i 的下近似集.

定义 3 如定义 2 所示, 条件类将对应一组决策类, 决策规则[2]定义为

$$f(x,C_i)\to f(x,D_j).$$

其中: $f(x,C_i)$ 为规则前件, $f(x,D_j)$ 为规则后件. 决策规则对应的可信度[2]为

$$\delta=|C_i\bigcap D_j|/|C_i|.$$

当 $\delta=1$ 时, 该决策规则为确定性规则, 否则为不确定性规则.

2 粒矩阵[13]与粒关系矩阵

定义 4 在不一致决策表 S 中, $U=\{u_1,u_2,\cdots,u_l\}$, $B\subseteq C$, 设 $U/B=\{B_1,B_2,\cdots,B_m\}$ 表示论域 U 在 B 下的条件类划分, $U/D=\{D_1,D_2,\cdots,D_n\}$ 表示论域 U 在 D 下的决策类划分. 定义条件粒矩阵 X 为

$$X=\begin{bmatrix}B_1\\B_2\\\vdots\\B_m\end{bmatrix}=\begin{bmatrix}a_{11}&a_{12}&\dots&a_{1l}\\a_{21}&a_{22}&\dots&a_{2l}\\\vdots&\vdots&\ddots&\vdots\\a_{m1}&a_{m2}&\dots&a_{ml}\end{bmatrix}.\tag{1}$$

其中

$$a_{ik}=\begin{cases}1,\ u_k\in B_i;\\0,\ u_k\notin B_i;\end{cases}\quad 1\leqslant k\leqslant l.$$

同理, 定义决策粒矩阵 Y 为

$$Y=\begin{bmatrix}D_1\\D_2\\\vdots\\D_n\end{bmatrix}=\begin{bmatrix}b_{11}&b_{12}&\dots&b_{1l}\\b_{21}&b_{22}&\dots&b_{2l}\\\vdots&\vdots&\ddots&\vdots\\b_{n1}&b_{n2}&\dots&b_{nl}\end{bmatrix}.\tag{2}$$

其中

$$b_{jk}=\begin{cases}1,\ u_k\in D_j;\\0,\ u_k\notin D_j;\end{cases}\quad 1\leqslant k\leqslant l.$$

定义 5 定义粒关系矩阵

$$\mathrm{GrM}=XY^{\mathrm{T}}=\\ \begin{bmatrix}B_1D_1&B_1D_2&\dots&B_1D_n\\B_2D_1&B_2D_2&\dots&B_2D_n\\\vdots&\vdots&\ddots&\vdots\\B_mD_1&B_mD_2&\dots&B_mD_n\end{bmatrix}=(c_{ij})_{m\times n}.\tag{3}$$

其中: Y^{T} 表示 Y 的转置; GrM 称为条件属性集 B 与决策属性 D 之间的粒关系矩阵, 它反映了条件类 B_i 与决策类 D_j 之间的包含关系; $c_{ij}=\sum\limits_{k=1}^{l}(a_{ik}b_{kj})$ 则反映了 B_i 包含于 D_j 中的对象的个数.

令 $\mathrm{NE}(i)=1$ 表示 GrM 中第 i 行非零元素个数为 1, $\mathrm{NE}(j)=1$ 表示 GrM 中第 j 列非零元素个数为 1.

在计算过程中, 先对不一致决策表 S 进行预处理: 将 S 中的所有记录重新排序, 所有不一致记录置前, 一致记录置后, 得到排序后的不一致决策表 S'. 于是有以下定义和性质.

定义 6 在不一致决策表 $S'=(U,C,D,V,f)$ 中, $B\subseteq C$, B 和 D 对应的条件粒矩阵和决策粒矩阵分别为 X 和 Y, $\mathrm{GrM}=XY^{\mathrm{T}}$, 将 GrM 分为两部分: 一部分由 GrM 中 $\mathrm{NE}(i)=1$ 的行组成, 称为相容粒关系矩阵 TGrM; 另一部分则由 GrM 中 $\mathrm{NE}(i)\neq 1$ 的部分组成, 称为不相容粒关系矩阵 NTGrM.

例如

$$\mathrm{GrM}=\begin{bmatrix}1&1&0\\0&1&2\\1&0&0\\0&2&0\end{bmatrix},$$

则 $\mathrm{NE}(3)=\mathrm{NE}(4)=1$, 于是

$$\mathrm{TGrM}=\begin{bmatrix}1&0&0\\0&2&0\end{bmatrix},\ \mathrm{NTGrM}=\begin{bmatrix}1&1&0\\0&1&2\end{bmatrix}.$$

根据定义 6, 有以下性质.

性质 1　考察 TGrM, 对于任意的 $\mathrm{NE}(i)=1$, 说明条件等价类 B_i 完全包含于决策类 D_j, 此时对应一条确定性规则.

性质 2　考察 TGrM, 若 $\mathrm{NE}(i)=1$ 且 $\mathrm{NE}(j)=1$, 则说明条件等价类 B_i 与决策类 D_j 一一对应, 此时对应一条确定性规则.

对某一决策类而言, 若 TGrM 满足性质 2, 则此时可以使得获取的确定性规则数目最少, 因此在获取规则时应优先考虑性质 2.

注 1　在不一致决策表 S' 中, 记 C 与 D 的粒关系矩阵为 CGrM, 按照定义 6, CGrM 可以分为两部分: 相容粒关系矩阵 TCGrM 和不相容粒关系矩阵 NTCGrM.

性质 3　考察 NTGrM 与 NTCGrM, 如果存在 NTGrM = NTCGrM, 则可以获取不确定性规则.

证明　NTGrM 与 NTCGrM 对应的均为不一致记录, 反映了不一致记录中条件类与决策类的包含关系, 当 NTGrM = NTCGrM 时, 说明去掉属性 $C-B$ 后不一致记录中条件类与决策类的包含关系不变, 对不一致记录而言, $\mathrm{POS}_B(D)=\mathrm{POS}_C(D)$, 因此可以获取不确定性规则. □

下面举例说明性质 3.

例 1　如表 1 所示, $U=\{u_1,u_2,u_3,u_4\}$, $C=\{a,b,c\}$, $D=\{d\}$.

表 1　不一致决策表

U	a	b	c	d
u_1	1	0	1	1
u_2	1	0	1	1
u_3	1	0	1	2
u_4	1	0	2	3

由表 1 计算得到 $\mathrm{CGrM}=\begin{bmatrix}2&1&0\\0&0&1\end{bmatrix}$, 则

$$\mathrm{TCGrM}=[0\ \ 0\ \ 1],\ \mathrm{NTCGrM}=[2\ \ 1\ \ 0].$$

若 $B=\{a,b\}$, 则可得到

$$\mathrm{GrM}=[2\ \ 1\ \ 1],\ \mathrm{NTGrM}=[2\ \ 1\ \ 1],$$

此处 NTGrM ≠ NTCGrM(NTCGrM 中第 3 个元素 0 变为 NTGrM 中的 1), 即去掉属性 c 后多了 1 条不一致记录, 第 4 条记录原本不属于决策属性值为 1 的决策类, 但在不考虑条件属性 c 时被分配到决策属性值为 1 的决策类, 使原本的不一致记录发生了变化, 不满足性质 3.

若考虑 $B=\{a,c\}$, 则

$$\mathrm{GrM}=\begin{bmatrix}2&1&0\\0&0&1\end{bmatrix},$$

$$\mathrm{TGrM}=[0\ \ 0\ \ 1],\ \mathrm{NTGrM}=[2\ \ 1\ \ 0].$$

此时 NTGrM = NCGrM, 满足性质 3 (不一致记录未发生变化), 可以提取不确定性规则

$$a=1\bigwedge c=1\rightarrow d=1|\delta=0.667,$$

$$a=1\bigwedge c=1\rightarrow d=2|\delta=0.333.$$

若 NTGrM 对应多个条件类, 则可以按性质 3 依次对每个条件类进行处理.

性质 1、性质 2 和性质 3 是本文进行规则获取的理论基础. 通过以上分析可知, 通过计算 TGrM 和 NTGrM, 并根据矩阵中包含的信息可以实现确定性规则和不确定性规则的提取.

3　不一致决策表规则获取的粒计算方法

不一致决策表可以看成一个“大”的粒度空间, 本文利用属性多粒度思想, 将决策表按粒度从粗到细划分成不同粒度的“小”的粒度空间, 在小的粒度空间中, 通过计算粒关系矩阵, 并充分挖掘隐含在矩阵中的启发式信息完成对决策表确定性规则和不确定性规则的提取. 为了描述算法, 先给出以下符号定义.

不失一般性, 假设不一致决策表 $S=(U,C,D,V,f)$, $C=\{c_1,c_2,\cdots,c_n\}$, $D=\{d\}$.

ω: 表征决策表粒度空间的粒度, 以条件属性的个数来度量, ω 越大, 粒度越细, $1\leqslant\omega\leqslant n$.

Gr_ω: 决策表按粒度 ω 划分成的粒度空间, 这里以条件属性组合表示, 这样的粒度空间有 c_n^ω 种, 即 $\mathrm{Gr}_\omega=\{\mathrm{Attr}_1,\mathrm{Attr}_2,\cdots,\mathrm{Attr}_{C_n^\omega}\}$.

RULObj: 规则覆盖的论域对象的集合.

RULH_1 和 RULH_2 为如下两个启发式算子:

RULH_1: 对于 $\mathrm{Attr}_i\in\mathrm{Gr}_\omega$, $\mathrm{RULObj}_{\omega-1}$ 表示在粒度 $\omega-1$ 下规则覆盖的论域集合, $\mathrm{RULObj}_{\mathrm{Attr}_i}$ 表示在粒度空间 Attr_i 中规则覆盖的论域集合, 则 $\mathrm{RULH}_1=|\mathrm{RULObj}_{\mathrm{Attr}_i}-\mathrm{RULObj}_{\mathrm{Attr}_i}\bigwedge\mathrm{RULObj}_{\omega-1}|$.

RULH_2: 对于 $\mathrm{Attr}_i\in\mathrm{Gr}_\omega$, 如果 TGrM 中存在 $\mathrm{NE}(i)=1$ 且 $\mathrm{NE}(j)=1$, 则 $\mathrm{RULH}_2=1$.

算法 1　不一致决策表规则获取算法.

输入: 不一致决策表 $S=(U,C,D,V,f)$;

输出: 所有规则集 Rules.

Step 1: 对 S 进行预处理得到 S', 初始化 Rules $=\varnothing$, RULObj $=\varnothing, \omega=1$, 计算 Y、CGrM 和 NTCGrM.

Step 2: 粒度由粗到细, 在粒度 $1 \leqslant \omega \leqslant n$ 的粒度空间中获取规则

$\mathrm{Gr}_\omega=\{\mathrm{Attr}_1, \mathrm{Attr}_2, \cdots\}$ //粒度空间,

$\mathrm{GrM}_\omega=\{\mathrm{GrM}_{\mathrm{Attr}_i} | \mathrm{Attr}_i \in \mathrm{Gr}_\omega\}$ //粒关系矩阵.

其中 $\mathrm{GrM}_{\mathrm{Attr}_i}=X_{\mathrm{Attr}_i} \cdot Y^{\mathrm{T}}$, 并得到

$$\mathrm{TGrM}_\omega=\{\mathrm{TGrM}_{\mathrm{Attr}_i} | \mathrm{Attr}_i \in \mathrm{Gr}_\omega\},$$

$$\mathrm{NTGrM}_\omega=\{\mathrm{NTGrM}_{\mathrm{Attr}_i} | \mathrm{Attr}_i \in \mathrm{Gr}_\omega\},$$

$$\mathrm{RULH1}_\omega=\{\mathrm{RULH1}_{\mathrm{Attr}_i} | \mathrm{Attr}_i \in \mathrm{Gr}_\omega\},$$

$$\mathrm{RULH2}_\omega=\{\mathrm{RULH2}_{\mathrm{Attr}_i} | \mathrm{Attr}_i \in \mathrm{Gr}_\omega\}.$$

若 $\mathrm{RULH1}_\omega \neq \varnothing$ 或 $\mathrm{RULH2}_\omega \neq \varnothing$, 则转 Step 3, 否则转 Step 4.

Step 3: 获取确定性规则.

1) 若 $\mathrm{RULH2}_\omega \neq \varnothing$, 则获取规则到 Rules;

2) 对 $\mathrm{RULH1}_\omega$ 中的元素按大到小进行排序, 并按此顺序获取规则到 Rules.

Step 4: 获取不确定性规则. 对于 $\mathrm{NTGrM}_{\mathrm{Attr}_i} \in \mathrm{NTGrM}_\omega$, 判断矩阵 $\mathrm{NTGrM}_{\mathrm{Attr}_i}$ 与 NTCGrM 是否相等, 若相等, 则获取不确定性规则到 Rules.

Step 5: 计算 RULObj. 若 $U-\mathrm{RULObj} \neq \varnothing$, 则 $\omega=\omega+1$, 并转 Step 2, 否则算法结束.

在同一粒度中, 先根据启发式信息 $\mathrm{RULH}_2=1$ 获取对应的规则, 再根据 RULH_1 获取对应的规则, 保证在同一粒度下获取的规则最少. 粒度由粗到细, 以保证规则获取是在一个尽可能小的条件属性集上进行, 获取的规则必然最简. 另外, 算法中加入判断规则是否覆盖论域的条件, 保证了获取的规则是完备的.

算法的复杂度分析: 算法 1 的核心是计算粒关系矩阵, 并根据矩阵中包含的启发式信息获取规则, 因此算法 1 的时间复杂度主要集中在对粒关系矩阵的计算. 在最坏情况下, 最多需要计算 2^n 次粒关系矩阵, 因此算法 1 的时间复杂度为 $O(2^n)$. 但是, 在实际情况中决策表的数据冗余性很大时, 算法 1 的时间复杂度要远远小于 $O(2^n)$, 并且算法中加入了终止条件, 以保证算法收敛更快. 此外, 算法的空间复杂度主要集中在对矩阵 X 和矩阵 Y 以及粒关系矩阵 GrM 的表示和存储上, 这 3 个矩阵占用的内存在最坏情况下为 $O(|C||U|)$, 虽然算法 1 在最坏情况下的空间复杂度为 $O(|C||U|)$, 但算法 1 处理的是具有稀疏特性的布尔矩阵 (矩阵中只含 0 和 1), 因此, 算法 1 的空间复杂度要比最坏情况好很多. 如果能够提高矩阵处理技巧, 算法 1 的时间和空间复杂度还有进一步降低的空间.

4 实验与分析

4.1 实例说明

例 2 如表 2 所示, $U=\{u_1, u_2, u_3, u_4, u_5, u_6\}$, $C=\{c_1, c_2, c_3, c_4\}$, $D=\{d\}$.

表 2 不一致决策表 S

U	c_1	c_2	c_3	c_4	d
u_1	1	0	0	0	1
u_2	0	1	1	1	2
u_3	0	1	0	0	2
u_4	0	1	1	0	2
u_5	0	1	0	0	1
u_6	0	1	0	0	1

Step 1: 对 S 进行预处理得到 S', 如表 3 所示.

表 3 不一致决策表 S'

U	c_1	c_2	c_3	c_4	d
u_1	0	1	0	0	1
u_2	0	1	0	0	1
u_3	0	1	0	0	2
u_4	1	0	0	0	1
u_5	0	1	1	1	2
u_6	0	1	1	0	2

Rules $=\varnothing$, RULObj $=\varnothing, \omega=1$, 计算 Y、CGrM 和 NTCGrM 如下:

$$Y=\begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix},$$

$$\mathrm{CGrM}=\begin{bmatrix} 2 & 1 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{bmatrix}.$$

于是 $\mathrm{NTCGrM}=[2 \quad 1]$.

Step 2: 粒度由粗到细, 在不同粒度的粒度空间中获取规则.

$$\mathrm{Gr}_\omega=\{c_1, c_2, c_3, c_4\},\ \mathrm{Attr}_i \in \mathrm{Gr}_\omega;$$

$$\begin{cases} \mathrm{GrM}_{c_1}=\begin{bmatrix} 2 & 3 \\ 1 & 0 \end{bmatrix}, \\ \mathrm{NTGrM}=[2 \quad 3], \\ \mathrm{TGrM}=[1 \quad 0]; \end{cases}$$

$$\begin{cases} \mathrm{GrM}_{c_2}=\begin{bmatrix} 2 & 3 \\ 1 & 0 \end{bmatrix}, \\ \mathrm{NTGrM}=[2 \quad 3], \\ \mathrm{TGrM}=[1 \quad 0]; \end{cases}$$

$$\begin{cases} \mathrm{GrM}_{c_3}=\begin{bmatrix} 3 & 1 \\ 0 & 2 \end{bmatrix}, \\ \mathrm{NTGrM}=[3 \quad 1], \\ \mathrm{TGrM}=[0 \quad 2]; \end{cases}$$

$$\begin{cases} \mathrm{GrM}_{c_4} = \begin{bmatrix} 3 & 2 \\ 0 & 1 \end{bmatrix}, \\ \mathrm{NTGrM} = [3 \quad 2], \\ \mathrm{TGrM} = [0 \quad 1]. \end{cases}$$

计算 RULH_1 和 RULH_2, 如表 4 所示.

表 4　RULH_1 和 RULH_2

粒度空间	RULH_1	RULH_2
c_1	1	0
c_2	1	0
c_3	2	0
c_4	1	0

$\mathrm{RULH}_1 \neq \varnothing$, 于是可以获取确定性规则.

Step 3: 获取确定性规则. 如表 4 所示, 不存在 $\mathrm{RULH}_2 = 1$, 于是对 RULH_1 值按大到小排序, 粒度空间 c_3 对应的 RULH_1 值最大, 其次是 c_1、c_2 和 c_4, 根据此排序可以获取确定性规则如下:

$$c_3 = 1 \rightarrow d = 2|\delta = 1,$$

$$c_1 = 1 \rightarrow d = 1|\delta = 1.$$

c_2 和 c_1 能获取的规则都对应第 4 条记录, 因为已对 c_1 获取规则, 所以不再对 c_2 获取规则. 同理, c_3 获取的规则对应第 5 条和第 6 条记录, c_4 获取的规则对应第 5 条记录, 所以不再获取. 此时, 确定性规则都已获取完毕, 故在粒度 $\omega = 2$ 时无需再获取确定性规则.

Step 4: 获取不确定性规则. 由 Step 2 中的结果可知, 对于所有粒度空间, 均有 $\mathrm{NTGrM} \neq \mathrm{NTCGrM}$, 因此在粒度 $\omega = 1$ 时无法获取不确定规则.

Step 5: 通过计算得到

$$\mathrm{RULObj} = \{u_4, u_5, u_6\},$$

$$U - \mathrm{RULObj} = \{u_1, u_2, u_3\} \neq \varnothing,$$

需要继续获取规则. $\omega = \omega + 1 = 2$, 计算过程如下:

$$\mathrm{Gr}_\omega = \{c_1c_2, c_1c_3, c_1c_4, c_2c_3, c_2c_4, c_3c_4\},$$

$$\mathrm{NTGrM}_{c_1c_2} = [2 \quad 3],$$

$$\mathrm{NTGrM}_{c_1c_3} = [2 \quad 1],$$

$$\mathrm{NTGrM}_{c_1c_4} = [2 \quad 2],$$

$$\mathrm{NTGrM}_{c_2c_3} = [2 \quad 1],$$

$$\mathrm{NTGrM}_{c_2c_4} = [2 \quad 2],$$

$$\mathrm{NTGrM}_{c_3c_4} = [3 \quad 1].$$

在上述结果中没有计算 RULH_1 和 RULH_2. 在 Step 3 中已分析出, 在粒度 $\omega = 2$ 时只需获取不确定性规则, $\mathrm{NTGrM}_{c_1c_3} = \mathrm{NTGrM}_{c_2c_3} = \mathrm{NTCGrM}$, 对粒度空间 c_1c_3 或 c_2c_3 获取不确定性规则即可覆盖整个论域. 这里选择 c_1c_3 获取规则如下:

$$c_1 = 0 \bigwedge c_3 = 0 \rightarrow d = 1|\delta = 0.667,$$

$$c_1 = 0 \bigwedge c_3 = 0 \rightarrow d = 2|\delta = 0.333.$$

此时计算

$$\mathrm{RULObj} = \{u_1, u_2, u_3, u_4, u_5, u_6\},$$

$$U - \mathrm{RULObj} = \varnothing.$$

至此规则集已覆盖论域, 计算结束.

4.2　UCI 数据集测试

为了进一步验证算法 1 的可行性, 本文借助 RIDAS[14]测试平台, 从 UCI 数据集中选择 9 组常用数据集, 用以测试算法 1、文献 [6] 中的基于最大粒的规则获取算法 (算法 a)、文献 [14] 中的基于代数观的属性约简算法 (算法 b) 和基于信息熵的属性约简算法 (算法 c). 具体做法是在每个数据集中各选取 50% 数据对每个算法进行测试并获取各自的规则集, 然后利用所获取的规则集对每个数据集整体分别进行识别, 从而得到各自的正确识别率. 表 5 显示了实验结果和对比效果.

表 5　实验结果对比

数据集	规则个数				正确识别率 / %			
	算法 1	算法 a	算法 b	算法 c	算法 1	算法 a	算法 b	算法 c
Haberman	110	86	105	105	76.7	73.2	73.5	73.5
Wine	11	54	86	87	99.0	64.0	54.5	55.1
Glass	61	85	105	100	80.5	57.0	52.3	52.8
Ecoli	72	63	90	90	86.2	84.5	84.8	84.8
Voting	24	27	29	33	96.3	95.9	97.7	95.6
Liver	104	89	119	119	79.0	81.7	77.7	77.7
Balance	158	132	172	172	82.9	77.6	80.8	80.8
Iris	8	7	8	8	97.8	98.7	98.7	98.7
Zoo	11	8	15	11	97.4	97.0	96.0	96.0

如表 5 所示, 在规则个数方面, 算法 1 相比于算法 a、算法 b 和算法 c 都有一定程度的减少. 但是, 在正确识别率上, 算法 1 相比于算法 a、算法 b 和算法 c 都高, 这是由于在处理包含不一致记录的数据集时, 本文从属性多粒度角度出发, 在粒度由粗到细的不同粒度空间中逐层提取规则, 不论得到的是确定性规则还

是不确定性规则, 都能保证规则的数量少, 无重复, 无冗余, 因此算法1能得到更简洁、更少的规则, 规则的泛化能力也更强.

5 结 论

本文在文献[15]的基础上, 提出了一种不一致决策表规则获取的粒计算方法. 首先从属性多粒度角度对决策表进行分析; 然后在不同粒度空间中获取规则; 最后从实例分析和实验验证两个方面说明了算法的可行性和有效性. 所提出的算法有以下特点:

1) 在不同粒度空间中获取规则, 使得复杂问题简单化;

2) 充分利用矩阵简单直观的特点, 定义了相容粒关系矩阵和不相容粒关系矩阵, 并挖掘隐含在矩阵中的启发式信息, 实现对决策规则的快速获取;

3) 粒度由粗到细, 规则获取在例化方向上进行, 保证了规则的最简性, 提高了规则的泛化能力.

算法的不足之处在于, 虽然本文使用的是布尔稀疏矩阵, 但矩阵的运用在一定程度上会增加算法的复杂度. 另外, 在同一粒度空间中获取的规则还有待优化. 因此, 如何降低算法的空间复杂度并进一步对算法进行优化都是值得研究的课题, 相关工作还在继续.

参考文献(References)

[1] Pawlak Z. Rough sets[J]. Int J of Computer and Information Science, 1982, 11(5): 341-356.

[2] 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: 147-152.
(Wang G Y. Rough set theory and knowledge acquisition[M]. Xi'an: Xi'an Jiaotong University Press, 2001: 147-152.)

[3] 张文修, 米据生, 吴伟志. 不协调目标信息系统的知识约简[J]. 计算机学报, 2003, 26(1): 1-7.
(Zhang W X, Mi J S, Wu W Z. The knowledge reduction of uncoordinated target information system[J]. Chinese J of Computers, 2003, 26(1): 1-7.)

[4] 黄兵, 周献中. 不一致决策表中规则提取的矩阵算法[J]. 系统工程与电子技术, 2005, 27(3): 441-445.
(Huang B, Zhou X Z. The inconsistent decision table matrix algorithm for extracting rules[J]. Systems Engineering and Electronics, 2005, 27(3): 441-445.)

[5] An J J, Wang G Y, Wu Y, et al. A rule generation algorithm based on granular computing[C]. Proc of the IEEE Int Conf on Granular Computing. Hong Kong, 2005: 102-107.

[6] 张清华, 王国胤, 刘显全. 基于最大粒的规则获取算法[J]. 模式识别与人工智能, 2012, 25(3): 386-396.
(Zhang Q H, Wang G Y, Liu X Q. Rule acquisition algorithm based on maximal granule[J]. Pattern Recognition and Artificial Intelligence, 2012, 25(3): 386-396.)

[7] Lin T Y. Granular computing: Practices, ries, and future directions[C]. Encyclopedia of Complexity and Systems Science. New York: Springer, 2009: 4339-4355.

[8] Qian Y H, Liang J Y, Yao Y Y, et al. MGRS: A multi-granulation rough set[J]. Information Sciences, 2010, 180(6): 949-970.

[9] Qian Y H, Zhang H, Sang Y, et al. Multigranulation decision-theoretic rough sets[J]. Int J of Approximate Reasoning, 2014, 55(1): 225-237.

[10] Liu X, Qian Y H, Liang J Y. A rule-extraction framework under multigranulation rough sets[J]. Int J of Machine Learning and Cybernetics, 2014, 5(2): 319-326.

[11] Yang X B, Song X N, Dou H L, et al. Multi-granulation rough set: from crisp to fuzzy case[J]. Annals of Fuzzy Mathematics and Informatics, 2011, 1(1): 55-70.

[12] Xu W H, Zhang X, Wang Q. A generalized multi-granulation rough set approach[C]. Bio-Inspired Computer and Application. Berlin: Springer Berlin Heidelberg, 2012: 681-689.

[13] Chen Z H, Xie G, Yan G W. Application of a matrix-based binary granular computing algorithm in RST[C]. Proc of IEEE Int Conf on Granular Computing. Beijing, 2005: 409-412.

[14] Wang G Y, Zheng Z, Zhang Y. RIDAS-A rough set based intelligent data analysis system[C]. Proc of the 1st Int Conf on Machine Learning and Cybernetics. Beijing, 2002: 646-649.

[15] 陈泽华, 张裕, 谢刚. 基于粒计算的最简决策规则挖掘算法[J]. 控制与决策, 2015, 30(1): 143-148.
(Chen Z H, Zhang Y, Xie G. Mining algorithm for concise decision rules based on granular computing[J]. Control and Decision, 2015, 30(1): 143-148.)

(责任编辑: 曹洪武)