**DOI**:10.11992/tis.201606005
**网络出版地址**:http://www.cnki.net/kcms/detail/23.1538.TP.20160808.0830.008.html

# 面向成组对象集的增量式属性约简算法

钱进[1,2],朱亚炎[1]
(1.江苏理工学院 计算机工程学院,江苏 常州 213015; 2. 南京信息工程大学 江苏省大数据分析技术重点实验室,江苏 南京 210044)

**摘　要**:现实世界中数据集都是动态变化的,非增量式属性约简方法从头重新计算原始数据集,而且未考虑先前约简结果中的信息,将耗费大量的时间和空间。为此,讨论了动态数据环境下约简的不变性,提出了一种面向成组对象集的增量式属性约简算法,利用先前约简中信息来快速获取强传承性的约简,从而提高增量式学习算法效率。最后,将该算法与非增量式约简方法和面向单个对象的增量式约简方法在 UCI 数据集和人工数据集上进行了相关比较。实验结果表明,面向成组对象的增量式属性约简算法能够快速处理动态数据,具有较好的约简传承性。

**关键词**:粗糙集;属性约简;成组对象集;约简传承性;增量式学习
**中图分类号**: TP181　**文献标志码**:A　**文章编号**:1673-4785(2016)04-0496-07

**中文引用格式**:钱进,朱亚炎. 面向成组对象集的增量式属性约简算法[J]. 智能系统学报, 2016, 11(4): 496-502.
**英文引用格式**:QIAN Jin, ZHU Yayan. An incremental attribute reduction algorithm for group objects[J]. CAAI Transactions on Intelligent Systems, 2016, 11(4): 496-502.

# An incremental attribute reduction algorithm for group objects

QIAN Jin[1,2], ZHU Yayan[1]
(1. School of Computer Engineering, Jiangsu University of Technology, Changzhou 213015, China; 2. Jiangsu Key Laboratory of Big Data Analysis Technology /B-DAT, Nanjing University of Information Science & Technology, Nanjing 210044, China)

**Abstract**: Real-world datasets change in size dynamically. Non-incremental attribute reduction methods usually need to re-compute source data when obtaining a new reduction without considering the information in the existing reduction, which consumes a great deal of computational time and storage space. Therefore, in this paper, some reduction invariance properties for dynamic datasets are discussed. An incremental attribute reduction algorithm for group objects using the previous reduction is proposed to quickly update a reduction with high inheritance rate and thus improve the efficiency of incremental learning. Finally, the incremental approach proposed is compared with an existing incremental attribute reduction algorithm for a single object, the non-incremental attribute reduction algorithms on the UCI, and synthetic datasets. Experimental results show that this incremental attribute reduction algorithm for group objects can deal with dynamic data rapidly, as it has better inheritance of reduction.

**Keywords**: rough set theory; attribute Reduction; group objects; inheritance rate of Reduct; incremental learning

属性约简是 Rough 集理论中的核心问题之一,也是知识获取的关键步骤。目前,许多学者已提出了一些有效的属性约简算法[1-4],如基于正区域的属性约简算法、基于差别矩阵的属性约简算法、基于信息熵的属性约简算法等,但这些算法都是针对静态的决策表,不适合处理动态的信息系统。现实世界是不断变化的,数据会源源不断地添加到原始决

**收稿日期**:2014-06-02.　**网络出版日期**:2014-08-08.
**基金项目**:江苏省自然科学基金项目(BK20141152);教育部人文社会科学研究青年基金项目(15YJCZH129);江苏省青蓝工程项目;江苏省大数据分析技术重点实验室开放基金项目(KXK1402); 江苏理工学院校级大学生创新项目(KYX15017).
**通信作者**:钱进. E-mail:qjqjlqyf@163.com.

策表中,一般不希望将原有的决策表和新产生的增量数据整合成一个新的决策表进行属性约简,因为这样会对原有数据不断地进行重复的计算。因此,如何利用原决策表中所含的信息并结合增量数据来进行属性约简成为粗糙集理论新的挑战。

数据的动态变化主要有 3 种情况:1)属性集保持不变而对象不断增加[5-8];2)对象集保持不变而属性集不断增加[9];3)对象集和属性集同时增加[10]。本文着重研究第 1 种情况的增量式属性约简问题,尤其研究适合大规模数据集的约简问题。文献[5]提出了基于正区域的属性约简增量式更新算法,提高了属性约简算法效率;文献[6]提出了基于差别矩阵的属性约简增量式更新算法;文献[7]提出了不使用可辨识矩阵的增量式核更新算法以及属性约简算法;文献[8]针对现有增量式属性约简算法中存在的约简传承性差以及不完备现象,提出了基于标记可辨识矩阵的增量式属性约简算法。然而,这些算法不适宜解决每次增加批量对象的问题。文献[11]提出了面向成组对象集的 3 种增量式信息熵属性约简算法;文献[12]充分利用先前约简中信息和计数排序算法快速更新批量对象的约简,降低计算复杂度;文献[13-14]探讨了混合属性约简算法以及利用 MapReduce 进行面向大规模数据集的属性约简方法。

为提高增量式学习算法效率[15]和约简传承性,本文构建了面向成组对象的增量式属性约简算法,利用原始决策表的一个候选约简来快速地更新新增决策表的约简,这样既提高了约简的传承性,又有效地利用了原有知识,提高了增量式学习算法效率。

## 1　粗糙集概念

下面简要介绍本文主要用到的一些 Rough 集的基本概念[1,9,11,13-14]。

**定义 1**　四元组 $S=\langle U,C\cup D,V,f\rangle$ 是一个决策表,其中 $U=\{x_1,x_2,\cdots,x_n\}$ 表示对象的非空有限集合,称为论域;$C$ 表示条件属性的非空有限集,$D$ 表示决策属性的非空有限集,$C\cap D=\varnothing$;$V=\bigcup_{a\in C\cup D}V_a$,$V_a$ 是属性 $a$ 的值域;$f:U\times(C\cup D)\to V$ 是一个信息函数,它为每个对象赋予一个信息值,即 $\forall a\in C\cup D$,$x\in U$,有 $f(x,a)\in V_a$;每一个属性子集 $R\subseteq C\cup D$ 决定了一个二元不可区分关系 IND($R$):

$$\mathrm{IND}(R)=\{(x,y)\in U\times U\mid \forall a\in R, f(x,a)=f(y,a)\}$$

关系 IND($R$)构成了 $U$ 的一个划分,用 $U/\mathrm{IND}(R)$ 表示,简记为 $U/R$。$U/R$ 中的任何元素 $[x]_R=\{y\mid \forall a\in R, f(x,a)=f(y,a)\}$ 称为等价类。不失一般性,假设决策表 $S$ 仅有一个决策属性 $D=\{d\}$,其决策属性值映射为 1,2,…,$k$,由 $D$ 导出的 $U$ 上划分记为 $U/D=\{D_1,D_2,\cdots,D_k\}$,其中,$D_i=\{x\in U\mid f(x,D)=i\}$,$i=1,2,\cdots,k$。

**定义 2**　在决策表 $S=\langle U,C\cup D,V,f\rangle$ 中,对于每个决策类 $D_i\in U/D$ 和不可区分关系 $A\subseteq C$,$D_i$ 的下近似集与上近似集分别可以由 $A$ 的基本集定义如下:

$$\underline{\mathrm{apr}}_A(D_i)=\cup\{x\in U\mid [x]_A\subseteq D_i\}$$

$$\overline{\mathrm{apr}}_A(D_i)=\cup\{x\in U\mid [x]_A\cap D_i\neq\varnothing\}$$

**定义 3**　在决策表 $S=\langle U,C,D,V,f\rangle$ 中,$\forall A\subseteq C$,正区域 $\mathrm{POS}_A(D)$ 和边界域 $\mathrm{BND}_A(D)$ 定义为

$$\mathrm{POS}_A(D)=\bigcup_{1\leqslant i\leqslant k}\underline{\mathrm{apr}}_A(D_i)$$

$$\mathrm{BND}_A(D)=\bigcup_{1\leqslant i\leqslant k}(\overline{\mathrm{apr}}_A(D_i)-\underline{\mathrm{apr}}_A(D_i))=U-\mathrm{POS}_A(D)$$

**定义 4**　在决策表 $S$ 中,一个属性集 $\mathrm{Red}\subseteq C$ 是 $C$ 的 $D$-约简,如果

1) $\mathrm{POS}_{\mathrm{Red}}(D)=\mathrm{POS}_C(D)$;

2) $\forall a\in\mathrm{Red}$,$\mathrm{POS}_{\mathrm{Red}-\{a\}}(D)\neq\mathrm{POS}_{\mathrm{Red}}(D)$。

**定义 5**　在决策表 $S$ 中,$A\subseteq C$,$\forall c\in A$,在正区域下属性 $c$ 重要性定义为

$$\mathrm{Sig}^{\mathrm{inner}}(c,A,D)=\gamma_A(D)-\gamma_{A-\{c\}}(D)$$

式中:$\gamma_A(D)=\dfrac{|\mathrm{POS}_A(D)|}{|U|}$。

**定义 6**　在决策表 $S$ 中,$A\subseteq C$,$\forall c\in C-A$,在正区域下属性 $c$ 重要性定义为

$$\mathrm{Sig}^{\mathrm{outer}}(c,A,D)=\gamma_{A\cup\{c\}}(D)-\gamma_A(D)$$

**定义 7**　设Red为决策表 $S$ 的候选属性约简,NewRed 为新增样本之后的新约简,则单次增量式约简的传承率(inheritance rate, IR)定义为

$$\mathrm{IR}=\frac{|\mathrm{Red}\cap\mathrm{NewRed}|}{\min(|\mathrm{Red}|,|\mathrm{NewRed}|)}$$

假设进行了 $w$ 次增量式约简,则平均传承率(inheritance rate average,IRA)定义为

$$\mathrm{IRA}=\sum_{i=1}^{w}\frac{\mathrm{IR}_i}{w}$$

在约简过程中,传承率越高则约简集的变化越小,对原始规则集的影响将越小。如果传承率为 1,说明新增的对象不影响原始的规则集;如果传承率为 0,则新的约简集与原来的约简集完全不同,这时需全部更新所有规则。

## 2 面向成组对象集的增量式属性约简算法

给定决策表 $S = \langle U, C \cup D, V, f \rangle$，一个约简 $\text{Red} \subseteq C$。新对象 $y$ 加入到决策表 $S$ 中，$U' = U \cup \{y\}$，将此时的新决策表标记为 $S'$。一种最简单的属性约简增量式更新算法是直接计算 $S'$ 的约简。显然，这种方法的属性约简效率比较低下，因为需要重复计算原始决策表 $S$ 中的等价类。为此，如何在已有的约简Red的基础上快速更新约简则成为本文主要研究内容。为此，如何在已有的约简Red的基础上快速更新约简则成为本文主要研究内容。为方便讨论，假设 $U/\text{Red} = \{X_1, X_2, \cdots, X_m\}$，$U/C = \{X'_1, X'_2, \cdots, X'_t\}$，用 $\text{POS}^U_{\text{Red}}$ 表示决策表 $S$ 由约简 Red导出的正区域，用 $\text{BND}^U_{\text{Red}}$ 表示决策表 $S$ 由约简 Red导出的边界域，即 $U - \text{POS}^U_{\text{Red}}$。

当新对象 $y$ 加入到 $S$ 中，主要分为两类情况：

1) $y$ 无法用Red区分，当且仅当 $\exists x \in U$ 使得 $\forall a \in \text{Red}[f(x,a) = f(y,a)]$；

2) $y$ 可以用Red区分，当且仅当 $\forall x \in U$ 使得 $\exists a \in \text{Red}[f(x,a) \neq f(y,a)]$。

对于第2种情况，显然 $\text{POS}^{U'}_{\text{Red}} = \text{POS}^U_{\text{Red}} \cup \{y\}$，则Red是新决策表 $S'$ 的约简。对于第1种情况，还不能完全判断出正区域的变化，需要对上述情况进一步细分，分为以下4种情况。

①若 $\exists X_i \in \text{BND}^U_C[y \in X_i]$，则 $\text{POS}^{U'}_{\text{Red}} = \text{POS}^{U'}_C$；

②若 $\exists X_i \in \text{POS}^U_C[y \in X_i]$，分为2种情况：$a)$ $\forall x \in X_i[f(x,d) = f(y,d)]$，则 $\text{POS}^{U'}_{\text{Red}} = \text{POS}^U_{\text{Red}} \cup \{y\}$，而 $\text{POS}^{U'}_C = \text{POS}^U_C \cup \{y\}$，即 $\text{POS}^{U'}_{\text{Red}} = \text{POS}^{U'}_C$；$b)$ $\exists x \in X_i[f(x,d) \neq f(y,d)]$，则 $\text{POS}^{U'}_C = \text{POS}^U_C - X_i$，而 $\text{POS}^{U'}_{\text{Red}} = \text{POS}^U_{\text{Red}} - X_i$，则 $\text{POS}^{U'}_C = \text{POS}^{U'}_{\text{Red}}$。总之，$\text{POS}^{U'}_{\text{Red}} = \text{POS}^{U'}_C$。

③若 $[y]_C \notin U/C \wedge [y]_{\text{Red}} \in U/\text{Red}$，若 $\exists x \in \text{POS}^U_{\text{Red}}$ 使得 $\forall z \in [x]_{\text{Red}}[f(z,d) = f(y,d)]$，则 $\text{POS}^{U'}_{\text{Red}} = \text{POS}^U_{\text{Red}} \cup \{\text{y}\}$，而 $\text{POS}^{U'}_C = \text{POS}^U_C \cup \{\text{y}\}$，则 $\text{POS}^{U'}_{\text{Red}} = \text{POS}^{U'}_C$；

④当 $[y]_C \notin U/C \wedge [y]_{\text{Red}} \in U/\text{Red}$，若 $\exists x \in \text{POS}^U_{\text{Red}}$ 使得 $\exists z \in [x]_{\text{Red}}[f(x,d) \neq f(y,d)]$，则 $\text{POS}^{U'}_{\text{Red}} = \text{POS}^U_{\text{Red}} - [x]_{\text{Red}}$，而 $\text{POS}^{U'}_C = \text{POS}^U{}_C \cup \{\text{y}\}$，即 $\text{POS}^{U'}_{\text{Red}} \neq \text{POS}^{U'}_C$。

根据上述分析，可以得出以下定理。

**定理 1** 给定决策表 $S$ 和新增对象 $y$，Red是决策表 $S$ 的约简，若 $\exists X_i \in \text{BND}^U_C[y \in X_i]$，则Red是新决策表 $S'$ 的约简。

**证明** 若Red为决策表 $S$ 的约简，则 $\text{POS}^U_{\text{Red}} = \text{POS}^U_C$。又 $\exists X_i \in \text{BND}^U_C[y \in X_i]$，则 $\text{POS}^{U'}_C = \text{POS}^U_C$ 和 $\text{POS}^{U'}_{\text{Red}} = \text{POS}^U_{\text{Red}}$，从而 $\text{POS}^{U'}_C = \text{POS}^{U'}_{\text{Red}}$，故Red为决策表 $S'$ 的约简。

**定理 2** 给定决策表 $S$ 和新增对象 $y$，Red是决策表 $S$ 的约简，若 $\exists X_i \in \text{POS}^U_C[y \in X_i]$ 且 $\forall z \in X_i[\forall a \in \text{Red}[f(z,a) = f(y,a)] \Rightarrow f(z,d) = f(y,d)]$，则Red是新决策表 $S'$ 的约简。

**证明** 若Red为决策表 $S$ 的约简，则 $\text{POS}^U_{\text{Red}} = \text{POS}^U_C$。又 $\exists X_i \in \text{POS}^U_{\text{Red}}[y \in X_i]$ 且 $\forall z \in X_i[\forall a \in \text{Red}[f(z,a) = f(y,a) \Rightarrow f(z,d) = f(y,d)]$，则 $\text{POS}^{U'}_C = \text{POS}^U_C \cup \{y\}$ 和 $\text{POS}^{U'}_{\text{Red}} = \text{POS}^U_{\text{Red}} \cup \{y\}$，从而 $\text{POS}^{U'}_{\text{Red}} = \text{POS}^{U'}_C$，故Red为决策表 $S'$ 的约简。

**定理 3** 给定决策表 $S$ 和新增对象 $y$，Red是决策表 $S$ 的约简，$y \notin U/C \wedge y \in U/\text{Red}$，若 $\exists z \in X_i[f(z,\text{Red}) = f(y,\text{Red})] \wedge f(z,d) \neq f(y,d)$，则Red不是决策表 $S'$ 的约简。

**证明** 若Red为决策表 $S$ 的约简，则 $\text{POS}^U_{\text{Red}} = \text{POS}^U_C$。又 $y \notin U/C$，则 $\text{POS}^{U'}_C = \text{POS}^U_C \cup \{y\}$。$[y]_{\text{Red}} \in U/\text{Red}$，若 $\exists z \in X_i[f(x,\text{Red}) = f(y,\text{Red})] \wedge f(x,d) \neq f(y,d)$，显然 $\text{POS}^{U'}_{\text{Red}} \neq \text{POS}^U_{\text{Red}} \cup \{y\}$，故Red不是决策表 $S'$ 的约简。

下面给出面向单个对象的增量式属性约简算法，如算法1所示。

**算法 1** 面向单个对象的增量式属性约简算法

**输入** 一个决策表，$S$；一个新增对象，$y$；一个约简，$\text{Red}_U$；

**输出** 一个新的约简，$\text{Red}_{U \cup \{y\}}$。

1) $A = \text{Red}_U$；

2) 计算 $U/A = \{A_1, A_2, \cdots, A_r\}$，$U/C = \{X_1, X_2, \cdots, X_t\}$，计算 $\text{POS}^U_C$ 和 $\text{POS}^U_A$ // $y$ 属于旧的正区域

3) 如果 $y \in \text{POS}^U_C$，则转到9)；// $y$ 属于旧的边界域

4) 如果 $y \in \text{BND}^U_C$，则转到9)；// $y$ 属于新的正区域对象

5) 如果 $y \notin U/A$，则转到9)；否则 $A'_h = A_h \cup \{y\}$；

6) 判断 $A'_h$ 一致性，若 $A'_h$ 中对象的决策一致，则转到9)；// $y$ 与其他对象在 $A$ 属性集上产生冲突

7) While ( $|\text{POS}^{A'_h}_A| \neq |\text{POS}^{A'_h}_C|$ )

{ For each attribute $c \in C - A$

计算 $\text{sig}^{\text{outer}}_{U'}(c, A, D)$；

$\text{sig}^{\text{outer}}_{U'}(a, A, D) = \max(\text{sig}^{\text{outer}}_{U'}(c, A, D))$；

$A = A \cup \{a\}$；

8)　For each attribute $c \in A$

　{　计算 $\mathrm{sig}^{\mathrm{inner}}_{U'}(c,A,D)$;

　　如果 $\mathrm{sig}^{\mathrm{inner}}_{U'}(c,A,D)=0$,则 $A=A-c$;

9) $\mathrm{Red}_{U\cup\{y\}}=A$, 输出 $\mathrm{Red}_{U\cup\{y\}}$。

**复杂度分析**　算法的时间复杂度主要集中在 2)、7)和 8)。利用文献[13]中计数排序算法和简化决策表处理方式,2)的时间复杂度为 $O(|C||U|)$,7)的时间复杂度为 $O(|C||U|)$,8)的最坏时间复杂度为 $O(|C|^2(|U/C|+1))$,故整个算法时间复杂度为 $max(O(|C||U|), O(|C|^2(|U/C|+1)))$,空间复杂度为 $O(|U|)$。

若每次只增加一个对象,使用算法 1 来计算约简,其计算效率较差。为此,本文主要考虑当每次增加一批对象时如何进行属性约简。给定一个决策表 $S$, $A\subseteq C$, $U/A=\{A_1,A_2,\cdots,A_r\}$。假设 $U^{\Delta}$ 是新增的对象集, $U^{\Delta}/A=\{A_1^{\Delta},A_2^{\Delta},\cdots,A_{r'}^{\Delta}\}$,那么 $(U\cup U^{\Delta})/A=\{A'_1,A'_2,\cdots,A'_h,A_{h+1},\cdots,A_r,A_{h+1}^{\Delta},\cdots,A_{r'}^{\Delta}\}$,其中 $A'_i=A_i\cup A_i^{\Delta}(i=1,2,\cdots,h)$。当成批增加对象时,我们主要考虑 $U\cup U^{\Delta}$ 中 $A$ 不能区分的对象集,进一步说就是考虑 $A'_i$ 中对象集以及 $A_{h+1}^{\Delta},\cdots,A_{r'}^{\Delta}$ 中 $C$ 能够区分而 $A$ 不能区分的对象集。下面给出面向成组对象集的增量式属性约简算法,如算法 2 所示。

**算法 2**　面向成组对象集的增量式属性约简算法(*GIAR*)

**输入**　一个决策表, $S$;一个候选约简, $\mathrm{Red}_U$;新增对象集, $U^{\Delta}$;

**输出**　一个新的约简, $\mathrm{Red}_{U\cup U^{\Delta}}$。

1) $A=\mathrm{Red}_U$, $U'_{\mathrm{bnd}}=\varnothing$;

2)计算 $U/A=\{A_1,A_2,\cdots,A_r\}$, $U^{\Delta}/A=\{A_1^{\Delta},A_2^{\Delta},\cdots,A_{r'}^{\Delta}\}$和 $(U\cup U^{\Delta})/A=\{A'_1,A'_2,\cdots,A'_h,A_{h+1},\cdots,A_r,A_{h+1}^{\Delta},\cdots,A_{r'}^{\Delta}\}$;

3) $U/C=\{X_1,X_2,\cdots,X_t\}$, $U^{\Delta}/C=\{X_1^{\Delta},X_2^{\Delta},\cdots,X_{t'}^{\Delta}\}$和 $(U\cup U^{\Delta})/C=\{X'_1,X'_2,\cdots,X'_{h'},X_{h'+1},\cdots,X_r,X_{h'+1}^{\Delta},\cdots,X_{t'}^{\Delta}\}$;

4)如果 $h=0$ 和 $h'=0$,转 5;

5)计算 $\mathrm{POS}_A^{U^{\Delta}}$ 和 $\mathrm{POS}_C^{U^{\Delta}}$,

如果 $\mathrm{POS}_A^{U^{\Delta}}=\mathrm{POS}_C^{U^{\Delta}}$,转到 10);否则转到 6)。

//新增对象集中约简不能区分的矛盾对象

6) $U'_{\mathrm{bnd}}=U^{\Delta}-\mathrm{POS}_A^{U^{\Delta}}$;

7)*for* $i=1$ to $h$

{如果 $A'_i$ 不一致,则 $U'_{\mathrm{bnd}}=U'_{\mathrm{bnd}}\cup A'_i$;}

//累加同一等价类中约简不能区分的矛盾对象

8)计算 $\mathrm{POS}_A^{U'_{\mathrm{bnd}}}$ 和 $\mathrm{POS}_C^{U'_{\mathrm{bnd}}}$;

9)While $(|\mathrm{POS}_A^{U'_{\mathrm{bnd}}}|\neq|\mathrm{POS}_C^{U'_{\mathrm{bnd}}}|)$

{For each attribute $c\in C-A$

　　计算 $\mathrm{sig}^{\mathrm{outer}}_{U'_{\mathrm{bnd}}}(c,A,D)$;

$\mathrm{sig}^{\mathrm{outer}}_{U'_{\mathrm{bnd}}}(a,A,D)=\max(\mathrm{sig}^{\mathrm{outer}}_{U'_{\mathrm{bnd}}}(c,A,D))$;

//若这样的属性有多个,则任选一个;

$A=A\cup\{a\}$;}

10)For each attribute $c\in A$

{　计算 $\mathrm{sig}^{\mathrm{inner}}_{U\cup U^{\Delta}}(c,A,D)$;

　如果 $\mathrm{sig}^{\mathrm{inner}}_{U\cup U^{\Delta}}(c,A,D)=0$,则 $A=A-\{c\}$;}

11) $\mathrm{Red}_{U\cup U^{\Delta}}=A$, 输出 $\mathrm{Red}_{U\cup U^{\Delta}}$。

**复杂度分析**　算法的时间复杂度主要集中在 2)、3)、9)和 10)。利用文献[13]中计数排序算法和简化决策表处理方式,2)的时间复杂度为 $O(|A||U\cup U^{\Delta}|)$, 3)的时间复杂度为 $O(|C||U\cup U^{\Delta}|)$, 9)的时间复杂度为 $O(|C||U'_{\mathrm{bnd}}|)$, 10)的最坏时间复杂度为 $O(|C|^2|[U\cup U^{\Delta}]/C|)$,故整个算法时间复杂度为 $\max(O(|C||U'_{\mathrm{bnd}}|), O(|C|^2|[U\cup U^{\Delta}]/C|))$,空间复杂度为 $O(|U\cup U^{\Delta}|)$。

**例 1**　表 1 给出一个决策表 $S$ 和新增对象集 $U^{\Delta}$,决策表 $S$ 包含 3 个约简 $\{c_2c_1\}$、$\{c_2c_3\}$ 和 $\{c_3c_4\}$,分别计算约简的变化情况,如表 2 所示。

**表 2**　原始决策表 $S$ 和新增对象集 $U^{\Delta}$

**Table 2**　**Original decision table $S$ and new added object set $U^{\Delta}$**

| $U$ | $c_1$ | $c_2$ | $c_3$ | $c_4$ | d | $U^{\Delta}$ | $c_1$ | $c_2$ | $c_3$ | $c_4$ | $d$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $x_1$ | 1 | 0 | 1 | 0 | 2 | $y_1$ | 2 | 1 | 0 | 0 | 1 |
| $x_2$ | 0 | 1 | 0 | 1 | 2 | $y_2$ | 1 | 0 | 1 | 0 | 4 |
| $x_3$ | 1 | 1 | 1 | 1 | 3 | $y_3$ | 0 | 1 | 0 | 1 | 2 |
| $x_4$ | 1 | 0 | 1 | 0 | 1 | $y_4$ | 0 | 2 | 1 | 1 | 1 |
| $x_5$ | 0 | 2 | 2 | 1 | 2 | $y_5$ | 0 | 1 | 1 | 1 | 2 |
| $x_6$ | 1 | 2 | 0 | 0 | 2 | $y_6$ | 0 | 1 | 1 | 2 | 1 |

表 2 约简变化与约简传承性比较

Table 2 Comparison of *Red*uct change and *Red*uct inheritance rate

| NewObject | Case | $\mathrm{Red}_U$ | $\mathrm{Red}_{U'}$ | IR | $\mathrm{Red}_U$ | $\mathrm{Red}_{U'}$ | IR | $\mathrm{Red}_U$ | $\mathrm{Red}_{U'}$ | IR |
|---|---|---|---|---|---|---|---|---|---|---|
| 21001 | 1) | $c_2c_1$ | $c_2c_1$ | 1 | $c_2c_3$ | $c_2c_1$ | 0.5 | $c_3c_4$ | $c_3c_4c_1$ | 1 |
| 10104 | ① | $c_2c_1$ | $c_2c_1$ | 1 | $c_2c_3$ | $c_2c_3$ | 1 | $c_3c_4$ | $c_3c_4$ | 1 |
| 01012 | ② | $c_2c_1$ | $c_2c_1$ | 1 | $c_2c_3$ | $c_2c_3$ | 1 | $c_3c_4$ | $c_3c_4$ | 1 |
| 02111 | ② | $c_2c_1$ | $c_2c_1$ | 1 | $c_2c_3$ | $c_2c_3$ | 1 | $c_3c_4$ | $c_3c_4c_1$ | 1 |
| 01112 | ③ | $c_2c_1$ | $c_2c_1$ | 1 | $c_2c_3$ | $c_2c_1$ | 0.5 | $c_3c_4$ | $c_3c_4c_1$ | 1 |
| 01121 | ④ | $c_2c_1$ | $c_2c_1c_3$ | 1 | $c_2c_3$ | $c_2c_3c_1$ | 1 | $c_3c_4$ | $c_3c_4$ | 1 |
| 合计 | | 平均传承率 | | 1 | 平均传承率 | | 0.83 | 平均传承率 | | 1 |

## 3 实验验证

为了评价所提出的增量式约简算法效率和约简传承性,使用 Windows 7 操作系统,2.4 GHz 处理器和 16 GB 内存的计算机和 Visual C#2012 实现了相关实验。由于所提出的约简算法和经典的约简算法仅能够处理离散型属性,先采用 Rosetta 软件(http://www.lcb.uu.se/tools/rosetta)填充缺省值,并将数值型属性连续值离散化;然后,分别在 4 个来自 UCI Repository 机器学习公共数据集和 2 个人工数据集进行实验。每个数据集仅有 1 个决策属性。人工数据集 Dataset1 每个属性值为 1~5;而人工数据集 Dataset2 每个属性值为 1~9。表 3 描述了 6 个数据集特性。原始数据集的 50%作为基本数据集,剩下 50%数据集的 20%、40%、60%、80%和 100%作为 5 个增量数据集,非增量式属性约简算法(NIAR)、面向单个属性的增量式属性约简算法(SIAR)和面向成组数据集的属性增量式约简算法(GIAR)实验结果如图 1 所示。

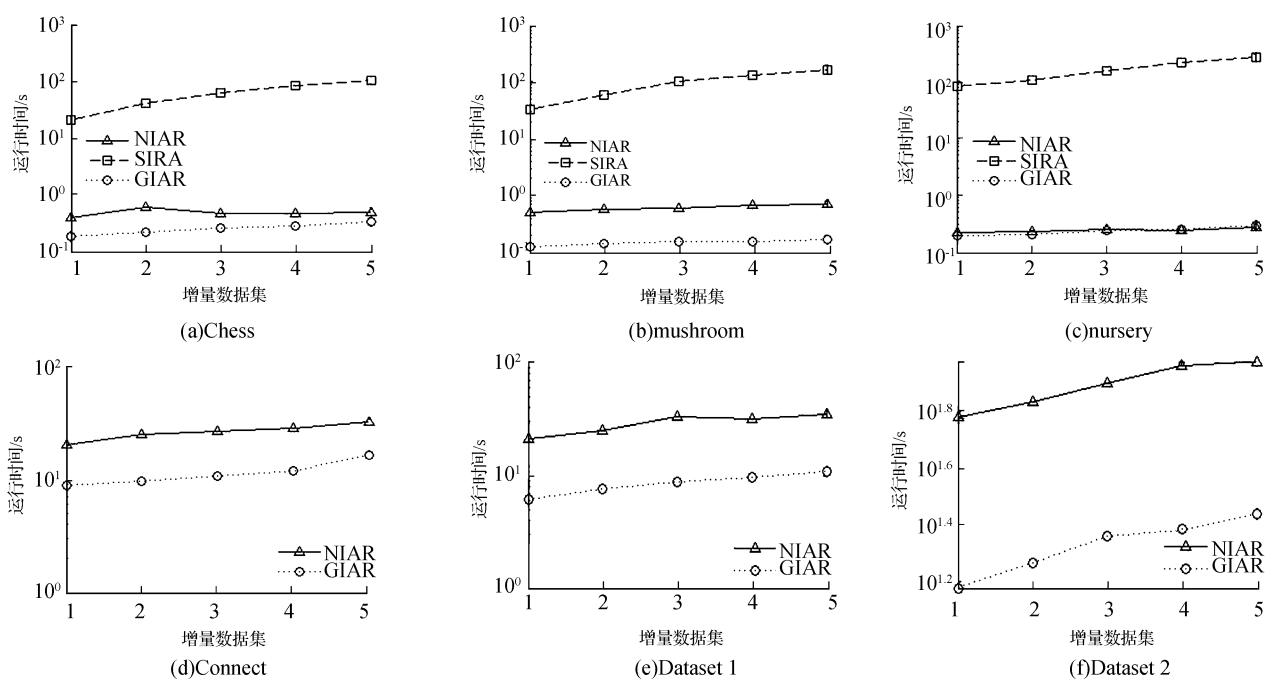



图 1 增量式约简算法和非增量式约简算法运行时间比较

Fig.1 Comparison of incremental and non-incremental *Red*uction algorithms on running time

由于面向单个属性的增量式约简算法(SIAR)对大规模数据集运行时间太长,图 1(e)-(f)未标出。从图 1 可以看出,SIAR 算法比 GIAR 算法运行时间更长,而 GIAR 算法的运行时间明显少于 NIAR 算法,特别对于较大数据集,算法的效果越明显。实验结果表明,所提出的面向成组对象集的增量式约

简算法是可行的。

表 3　6 个数据集特性

**Table 3　A description of six data sets**

| 序号 | 数据集 | 对象数 | 属性个数 | 类别数 |
|---|---|---|---|---|
| 1 | Chess | 3 196 | 36 | 2 |
| 2 | mushroom | 8 124 | 22 | 2 |
| 3 | nursery | 12 960 | 9 | 5 |
| 4 | Connect | 67 557 | 42 | 3 |
| 5 | Dataset1 | 200 000 | 30 | 5 |
| 6 | Dataset2 | 500 000 | 30 | 9 |

下面比较约简长度与约简的传承性。图 2 给出了 6 个数据集在不同增长比例下的约简长度。在 4 个数据集上,约简不变,则只需要生成新增数据集的规则,原始规则集不必重新生成,而在另外两个数据集上,约简长度稍微增长,这主要因为新增对象与原始数据集引起冲突,需要另外的属性集来细分原始对象,这时除了生成新规则集外,还需要修改部分原始规则。

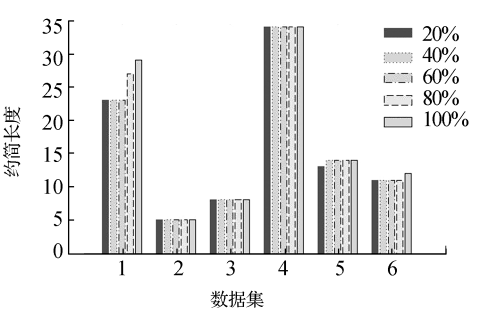



图 2　约简长度比较

**Fig.2　Comparison of *Red*uct length**

表 4 给出了约简的传承性。从表 4 可以看出,利用先前约简中信息所得到的新约简结果变化不大,约简传承性较好。

表 4　约简传承性比较

**Table 4　Comparison of *Red*uct inheritance rate**　%

| 数据集 | 20% | 40% | 60% | 80% | 100% |
|---|---|---|---|---|---|
| Chess | 100 | 100 | 95.45 | 95.45 | 95.45 |
| mushroom | 100 | 100 | 100 | 100 | 100 |
| nursery | 100 | 100 | 100 | 100 | 100 |
| Connect | 100 | 100 | 100 | 100 | 100 |
| Dataset1 | 100 | 100 | 100 | 100 | 100 |
| Dataset2 | 100 | 100 | 90.9 | 90.9 | 100 |

## 4　结论

在数据挖掘中,属性约简仅仅是数据预处理中一个过程,挖掘规则才是最终的输出。因此,充分利用先前约简中信息不仅能够快速得到约简,而且更容易地利用已有知识进行规则更新。本文所提出的面向成组对象集的增量式约简方法就是充分利用先前约简中信息来快速更新约简,不仅具有较高的约简传承率,而且可以快速进行增量式学习,具有良好的实用性。作者下一步将利用 Map *Red*uce 进一步研究大规模数据集增量式属性约简方法。

## 参考文献:

[1]PAWLAK Z. Rough sets[J]. International journal of computer & information sciences, 1982, 11(5): 341-356.

[2]SKOWRON A, RAUSZER C. The discernibility matrices and functions in information systems[M]//SLOWINSKI R. Intelligent Decision Support, Handbook of Applications and Advances of the Rough Sets Theory. Netherlands: Springer, 1992: 311-362.

[3]苗夺谦, 胡桂荣. 知识约简的一种启发式算法[J]. 计算机研究与发展, 1999, 36(6): 681-684.
MIAO Duoqian, HU Guirong. A heuristic algorithm for reduction of knowledge [J]. Journal of computer research and development, 1999, 36(6): 681-684.

[4]王国胤, 于洪, 杨大春. 基于条件信息熵的决策表约简[J]. 计算机学报, 2002, 25(7): 759-766.
WANG Guoyin, YU Hong, YANG Dachun. Decision table reduction based on conditional information entropy [J]. Chinese journal of computers, 2002, 25(7): 759-766.

[5]HU Feng, WANG Guoyin, HUANG Hai, et al. Incremental attribute reduction based on elementary sets[M]//SLEZAK D, WANG Guoyin, SZCZUKA M, et al. Proceedings of the 10th International Conference on Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing. Berlin Heidelberg: Springer, 2005: 185-193.

[6]杨明. 一种基于改进差别矩阵的属性约简增量式更新算法[J]. 计算机学报, 2007, 30(5): 815-822.
YANG Ming. An incremental updating algorithm for attribute reduction based on improved discernibility matrix[J]. Chinese journal of computers, 2007, 30(5) 815-822.

[7]冯少荣, 张东站. 一种高效的增量式属性约简算法[J]. 控制与决策, 2011, 26(4): 495-500.
FENG Shaorong, ZHANG Dongzhan. Effective incremental algorithm for attribute reduction[J]. Control and decision, 2011, 26(4): 495-500.

[8]尹林子, 阳春华, 王晓丽, 等. 基于标记可辨识矩阵的增量式属性约简算法[J]. 自动化学报, 2014, 40(3): 397-404.
YIN Linzi, YANG Chunhua, WANG Xiaoli, et al. An in-

cremental algorithm for attribute reduction based on labeled discernibility matrix[J]. Acta automatica sinica, 2014, 40(3): 397-404.

[9] SHU Wenhao, SHEN Hong. Updating attribute reduction in incomplete decision systems with the variation of attribute set[J]. International journal of approximate reasoning, 2014, 55(3): 867-884.

[10] CHEN Hongmei, LI Tianrui, LUO Chuan, et al. A Decision-theoretic rough set approach for dynamic data mining [J]. IEEE transactions on fuzzy systems, 2015, 23(6): 1958-1970.

[11] LIANG Jiye, WANG Feng, DANG Chuangyin, et al. A group incremental approach to feature selection applying rough set technique [J]. IEEE transactions on knowledge and data engineering, 2014, 26(2): 294-308.

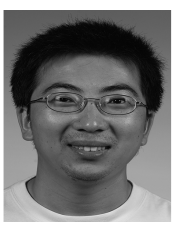
[12] QIAN Jin, YE Feiyue, LV Ping. An incremental attribute reduction algorithm in decision table[C]//Proceedings of the 7th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD). Yantai, China: IEEE, 2010, 4: 1848-1852.

[13] QIAN Jin, MIAO Duoqian, ZHANG Zehua, et al. Hybrid approaches to attribute reduction based on indiscernibility and discernibility relations[J]. International journal of approximate reasoning, 2011, 52(2): 212-230.

[14] QIAN Jin, MIAO Duoqian, ZHANG Zehua, et al. Parallel attribute reduction algorithms using MapReduce [J]. Information sciences, 2014, 279: 671-690.

[15] 康向平, 苗夺谦. 一种基于概念格的集值信息系统中的知识获取方法[J]. 智能系统学报, 2016, 11(3): 287-293.
KANG Xiangping, MIAO Duoqian. A knowledge acquisition method based on concept latticein set-valued information systems[J]. CAAI transactions on intelligent systems, 2016, 11(3): 287-293.

**作者简介:**



钱进,男,1975 年生,副教授,博士,主要研究方向为粗糙集、粒计算、云计算、大数据等。发表学术论文 40 余篇,其中被 SCI、EI 检索 20 余篇。

朱亚炎,男,1994 年生,主要研究方向为粗糙集、云计算等。

# 2017 IEEE 机电一体化与自动化国际会议
# 2017 **IEEE International Conference on Mechatronics and Automation**

The 2017 IEEE International Conference on Mechatronics and Automation (ICMA 2017) will take place in Takamatsu, Kagawa, Japan from August 6 to August 9, 2017. Takamatsu is a small city located at Sikoku which is the smallest island in 4 main islands of Japan. Shikoku contains a lot of temples including Zentsu-ji, where one of the most famous Buddhists, Kukai, was born. As the host city of ICMA 2017, Takamatsu not only provides the attendees with a great venue for this event, but also an unparalled experience in the Japanese history through several historical architectures. You are cordially invited to join us at IEEE ICMA 2017 in Takamatsu. The objective of ICMA 2017 is to provide a forum for researchers, educators, engineers, and government officials involved in the general areas of mechatronics, robotics, automation and sensors to disseminate their latest research results and exchange views on the future research directions of these fields.

**website**: http://2017.ieee-icma.org/