doi:10.3969/j.issn.1001-4616.2017.03.007

基于优势-等价关系的属性约简算法

吴婷婷[1],李　艳[1],郭娜娜[1],何　强[2]

(1.河北大学数学与信息科学学院,河北省机器学习与计算智能重点实验室,河北 保定 071002)
(2.北京建筑大学理学院,北京 100044)

[**摘要**]　考虑多标准分类问题,即条件属性具有偏好关系而决策属性是无序的类别,通过在条件属性上引入优势关系而决策属性仍然用等价关系来描述不同的属性. 针对这类信息系统,本文提出了一种基于样例对的矩阵约简算法. 区别于传统的基于辨识矩阵约简方法,该算法在不计算辨识矩阵的前提下,通过选择样例对,来找到辨识矩阵中对约简有用的属性,因此,所提算法能够明显改善计算约简的时间耗费. 进一步,为了处理较大规模的数据,提出了一种近似约简算法,该算法按属性重要性添加属性到约简中,进一步缩短了求取约简的时间. 最后在 UCI 数据集上进行大量的实验与传统的约简算法进行了对比,表明了所提出算法的可行性与有效性.

[**关键词**]　粗糙集,优势-等价关系,属性约简,辨识矩阵,样例对
[**中图分类号**]TP311　[**文献标志码**]A　[**文章编号**]1001-4616(2017)03-0045-07

Attribute Reduction Algorithms Based on Dominance-Equivalence Relations

Wu Tingting[1],Li Yan[1],Guo Nana[1],He Qiang[2]

(1.College of Mathematics and Information Science,Hebei University,Key Lab. of Machine Learning and Computational Intelligence,Baoding 071002,China)
(2.College of Science,Beijing University of Civil Engineering and Architecture,Beijing 100044,China)

Abstract:Considering multiple criteria classification problems,dominance relations and equivalence relations can be respectively introduced to condition attributes and decision attributes to describe different types of data. Based on the dominance-equivalence relations,a novel attribute reduction method based on sample pair selection is developed to deal with this kind of information systems. Instead of calculating the whole discernibility matrix,the proposed method only store the useful attributes for attribute reduction by selecting the discerned sample pairs,and therefore it can significantly improve the time costin attribute reduction. In addition,we propose an approximate reduction algorithm in order to deal with comparative large-scale information systems. This algorithm add attributes based on attribute importance and it's time saving. Finally,the experimental results on UCI data sets demonstrate the feasibility and effectiveness of the proposed algorithms.

Key words:rough set,dominance-equivalence relation,attributes reduction,discernibility matrix,sample pair

粗糙集理论[1-5]是近年发展起来的一种处理不精确性、不确定性和模糊知识的软计算工具,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等领域[6-8]. 经典粗糙集是以完备信息系统为研究对象,以等价关系为基础,把论域划分为互不相交的等价类,划分越细,信息越充分,知识越丰富.

知识约简是粗糙集理论的核心问题之一. 众所周知,知识库中描述知识的属性并不是同等重要的,甚至有些属性是冗余的,知识约简就是在保持知识库分类能力不变的前提下,去除其中不相关或不重要的属性. 通过知识约简可以使知识表示简化,又不丢失基本信息,使人们能够更深入地理解知识并进行决策. 知识约简的研究[9-14]主要是在等价关系下的信息系统中进行,要定义论域上的等价关系,属性值必须是离散的,但是,在实际问题中属性值更多是连续的,其中部分具有偏序关系. 传统的粗糙集方法在处理这类问题时需先将其离散化[15-16],然后再进行处理,这就会导致信息丢失,所以,建立基于优势关系的信息系统有助于处理连续属性或偏序关系的问题. 不少学者对这一问题进行了大量的研究[17-18],在对基于优势关系下的信息系统知识约简的研究已经有了不少成果[19-22]. 针对知识约简,主要方法有两类:一是基

收稿日期:2017-03-18.
基金项目:国家自然科学基金(61170040、61473111)、河北省自然科学基金(F2014201100、A2014201003).
通讯联系人:李艳,博士,教授,研究方向:机器学习. E-mail:ly@hbu.cn

于辨识矩阵的;二是基于重要性度量的. 辨识矩阵中不是所有的属性集都对约简起作用,即有些属性集是冗余的,在求约简时如果遍历整个辨识矩阵并且对其进行处理,会造成空间和时间上的极大浪费. Chen 等人[23]针对这个问题,提出了基于样例对的属性约简方法,并且指出提取样例对信息时,只选择对约简有用的必要的样例对,而其他冗余的样例对不参与运算,从而显著提高了约简的效率. 但是这个工作是建立在等价关系上的,只能处理符号值数据,对连续值和有序值,必须先进行离散化处理.

本文所研究的对象为多准则分类问题中的目标信息系统,即在条件属性上引入偏序关系,在决策属性上引入等价关系的信息系统. 将[23]基于等价关系所提出的约简算法推广到优势-等价关系下,提出相应的基于样例对的矩阵约简算法,从而提高优势关系下属性约简的效率,并且为了处理较大规模的数据,本文提出了一种近似约简算法,该算法依照属性重要性选择属性,在求约简时,更加高效.

1　基本概念

作为研究基础,本节简单介绍几个与本文密切相关的概念.

定义 1(目标信息系统)　四元组 $S=(U,A,V,f)$ 是一个目标信息系统,其中 $U=\{x_1,x_2,\cdots,x_n\}$ 是对象的一个非空集合,称为论域;$A=C\cup D$,$C\cap D=\varnothing$ 是一个有限属性集,其中 C 为条件属性集,D 为决策属性集;V 是属性值的集合;$f:U\times A\to V$,指在每个属性上对 U 中每个对象给出一个 V 中的值,即 $\forall a\in A,x\in U$,$f(x,a)\in V$. 目标信息系统也称为决策表.

定义 2(不可分辨关系和等价类)　设决策表为 $S=(U,C\cup D)$,对于任意一个非空属性子集 $B\subseteq C$,定义如下等价关系为不可分辨关系:

$\mathrm{IND}(B)=\{(x,y)\in U\times U:a(x)=a(y),\forall a\in B\}$,

这个等价关系把 U 划分成等价类的集合,表示为 $U/\mathrm{IND}(B)=\{[x]_B:x\in U\}$,在这里 $[x]_B=\{y\in U:(x,y)\in\mathrm{IND}(B)\}$ 叫做基于等价关系 $\mathrm{IND}(B)$ 的 x 的等价类.

定义 3(划分)　属性子集 B 上的不可分辨关系 $\mathrm{IND}(B)$ 形成 U 上的一个划分,记作 $U/\mathrm{IND}(B)$,简记为 U/B,其中 $[x_i]_R$ 为 R 上的等价类.

定义 4(优势/劣势关系)　设 $S=(U,A,V,f)$ 为一个目标信息系统,$A=C\cup D$,$P\subseteq C$,$x,y\in U$,对于每一个有序属性 $q\in P$,如果满足 $f(x,q)\geqslant f(y,q)$,则称 x 在属性 P 上优于 y,记做 xD_Py,反之为 yD_px,xD_Py 与 yD_px 分别称为定义在 P 上的优势关系、劣势关系.

在此基础上,可以分别定义 U 中一个对象的优势集和劣势集.

定义 5(优/劣势集)　给定 $P\subseteq C$,$x\in U$,定义 $D_P^+(x)=\{y\in U:yD_Px\}$ 表示优于 x 的对象的集合,称作 P 上对象 x 的优势集.

$D_P^-(x)=\{y\in U:xD_Py\}$ 表示劣于 x 的对象的集合,称作 P 上对象 x 的劣势集.

定义 6(上/下近似)　决策属性 D 的值对论域构成一个划分,即 $U/D=cl=\{cl_t,t=1,2,\cdots,n\}$,其中 cl_t 为论域 U 基于决策属性 D 形成的第 t 个等价类,则 cl_t 基于优势集的上下近似记做:

$$\underline{P}(cl_t^{\geqslant})=\{x\in U:D_P^+(x)\subseteq cl_t^{\geqslant}\},\ \overline{P}(cl_t^{\geqslant})=\{x\in U:D_P^-(x)\cap cl_t^{\geqslant}\neq\phi\}.$$

记 $\mathrm{POS}_U(C,D)=\bigcup\limits_{x\in U/D}\underline{P}(cl_t^{\geqslant})$ 为优势关系下条件属性集相对于决策属性集的正域. 正域为 U 中能够确切地划分到 cl_t 中的对象 x 的集合. 去除某些属性后,正域不变,即信息系统的决策能力不变.

定义 7(属性约简)　目标信息系统中的属性并不是同等重要的,属性约简是指可以找到一个较小的属性集 $B\subseteq C$,使得可用 C 描述的对象集合必然可用 B 描述,从而消除冗余属性.

具体定义如下:给定四元组 $S=(U,A,V,f)$ 是一个目标信息系统,$A=C\cup D$,$C\cap D=\varnothing$,其中 C 为条件属性集,D 为决策属性,子集 $B\subseteq C$ 是 C 的一个约简,如果它满足以下两个条件:

(1) $\mathrm{POS}_U(B,D)=\mathrm{POS}_U(C,D)$;

(2) $\forall a\in B,\mathrm{POS}_U(B-\{a\},D)\neq\mathrm{POS}_U(B,D)$.

即 B 为能够保持原有决策系统正域不变的最小的属性子集.

定义 8(属性的重要性)　优势关系下属性的重要性表示为:

$$\mathrm{SGF}(a,B,D)=\frac{|\mathrm{POS}_U(B,D)|-|\mathrm{POS}_U(B-\{a\},D)|}{|U|},$$

求属性约简最常用并且高效的方法是基于辨识矩阵的方法. 优势关系下辨识矩阵的定义如下.

定义 9(辨识矩阵) 设为一个目标信息系统,$A=C\cup D$,$C\cap D=\varnothing$,其辨识矩阵是一个 $n\times n$ 的矩阵,记为 $\boldsymbol{Q}$,矩阵元素定义为 $f(x_i,a)$,表示对象 x_i 在属性 a 上的属性值.

辨识矩阵中,单个元素的并为核,在核的基础上按属性重要性依次添加属性,直至与辨识矩阵中的每个元素的交都不空为止,最终得到的属性集合即为约简.

2 基于样例对的矩阵约简算法

基于等价关系的属性约简是粗糙集研究的一个重要问题,传统的辨识矩阵求所有约简时,其辨识函数是由矩阵中所有元素进行合取/析取运算后得到,根据化简后的辨识函数,便可得到原决策系统的所有约简,此过程中矩阵中所有元素都要参与运算,而矩阵中的元素是由特定样例对决定的,即原决策表中的所有样例都会参与到运算中,且每个样例都被认为是同等重要的. 文献[23]指出辨识矩阵中只有最小的元素对约简有用,从而说明只有对应最小元的样例对在计算约简中是有效的. 因此,[23]基于等价关系提出了粗糙集框架下的样例对选择方法来进行属性约简,有效地缩减了求取约简所用的时间与空间. 将其中的概念和方法扩展到基于优势-等价关系的信息系统上,并建立相应的约简方法. 下面首先给出基于优势-等价关系信息系统的定义,再将样例对选择方法推广,给出相应的算法.

定义 10(优势-等价关系信息系统) 设 $S=(U,A,V,f)$ 为一个目标信息系统,$A=C\cup D$,$P\subseteq C$,$x,y\in U$,对于每一个有序属性 $q\in P$,称 $R_P^{\leqslant}=\{(x,y)\in U\times U:f(x,q)\leqslant f(y,q)\}$,$R_D=\{(x,y)\in U\times U:f(x,D)=f(y,D)\}$ 分别为定义在 S 上的优势关系和等价关系,满足此关系的信息系统 S 称作基于优势-等价关系的信息系统.

定义 11(样例对选择定义) 设 $S^*=(U,C\cup D)$ 是一个决策系统,其中 $U=\{x_1,x_2,\cdots,x_n\}$,$\boldsymbol{Q}^*(C\cup D)$ 是 S^* 的优势矩阵,优势函数为 $f_U(C\cup D)$,$S\subset U\times U$,令 $f(S,C\cup D)=\wedge(\vee C_{ij})$,$C_{ij}\neq\phi$,$C_{ij}\in\boldsymbol{Q}^*(C\cup D)$,$(x_i,x_j)\in S$,那么 $f(S,C\cup D)\geqslant f_U(C\cup D)$ 成立. 假设 $(x_i,x_j)\in U\times U$,如果 $f(U\times U-\{(x_i,x_j)\},C\cup D)=f_U(C\cup D)$,那么 (x_i,x_j) 在 $U\times U$ 中可去;否则 (x_i,x_j) 在 $U\times U$ 中不可去. $U\times U$ 中所有不可去的样例对的集合称作 S^* 的样例对的核,记做 $\mathrm{Core}_{C\cup D}(U\times U)$. 如果 S 是 $U\times U$ 中的一个最小子集,使得 $f(S,C\cup D)=f_U(C\cup D)$ 成立,那么 $S\subset U\times U$ 称作 S^* 的一个样例对选择. 样例对选择不唯一,S^* 中所有样例对选择的集合记为 $\mathrm{Sel}_{C\cup D}(U\times U)$,且 $\mathrm{Core}_{C\cup D}(U\times U)=\cap\mathrm{Sel}_{C\cup D}(U\times U)$.

定义 11 给出的样例对选择方法需先求出优势矩阵,并且选择样例对的方法是一个盲目搜索的过程,计算负载较大,因此给出了在不求优势矩阵的前提下选择样例对的方法,以下给出了相应的定义.

定义 12(二元关系 $\mathrm{DIS}(\{c\})$) 优势-等价关系下,称二元关系 $\mathrm{DIS}(\{c\})$ 为条件属性 $c\in C$ 关于决策属性 D 的不可区分关系:

$$\mathrm{DIS}(\{c\})=\{(x_i,x_j):c(x_i)>c(x_j),i<j\}.$$

式中,x_i,x_j 满足以下条件之一:(1) $x_i\notin\mathrm{POS}_U(C,D)$ 且 $x_j\in\mathrm{POS}_U(C,D)$;(2) $x_i\in\mathrm{POS}_U(C,D)$ 且 $x_j\notin\mathrm{POS}_U(C,D)$;(3) $x_i,x_j\in\mathrm{POS}_U(C,D)$ 且 $D(x_i)\neq D(x_j)$.

$\mathrm{DIS}(\{c\})$ 中所有 (x_i,x_j) 的集合定义为 $\mathrm{DIS}(C)=\cup_{c\in C}\mathrm{DIS}(\{c\})$.

条件属性 c 可以划分到 3 个不相交的子集中:(1)属于核;(2)属于某些约简;(3)不属于任何约简. 由此给出以下定义来描述不同子集的条件属性所对应的区分关系.

定义 13(不同子集的条件属性对应的区分关系) 假设 $S^*=(U,C\cup D)$ 是一个决策系统,对每一个条件属性 $c\in C$,以下关系成立:(1) $c\in\mathrm{Core}_U(C\cup D)\Leftrightarrow\exists C_{ij}\neq\phi$,使得 $C_{ij}=\{c\}$;(2) $c\in\cup\mathrm{Red}_U(C\cup D)-\mathrm{Core}_U(C\cup D)\Leftrightarrow$存在一个最小的 $C_{ij}\neq\phi$,使得 $\{c\}\subset C_{ij}$;(3) $c\notin\cup\mathrm{Red}_U(C\cup D)\Leftrightarrow$对于 $\forall C_{ij}$ 如果 $c\in C_{ij}$,那么存在 $C_{i0j0}\neq\phi$,使得 $C_{i0j0}\subset C_{ij}$ 并且 $c\notin C_{i0j0}$.

在计算约简时,满足定义 13 中条件 1 的核属性将被分到每一个约简中,满足条件 2 的属性将被分到部分约简中,而满足条件 3 的属性将不会分到任何约简中去.

求约简只需找到满足定义 13 中的前两个条件的属性 c 即可,由此,得到推论 1 如下:

推论 1 $c\in\cup\mathrm{Red}_U(C\cup D)\Leftrightarrow\exists(x_{i0},x_{j0})\in\mathrm{DIS}(\{c\})$ 使 $\mathrm{DIS}(C)-\cap\{\mathrm{DIS}(\{b\}):(x_{i0},x_{j0})\in\mathrm{DIS}(\{b\})\}\subseteq\cup\{\mathrm{DIS}(\{c'\}):(x_{i0},x_{j0})\notin\mathrm{DIS}(\{c'\})\}$ 成立.

将满足以上条件的样例对 (x_{i0},x_{j0}) 称作属性 c 的关键样例对,记做 C_k. 如果 $C_k\neq\phi$,那么存在 $(x_{i0},x_{j0})\in$

C_k,使得 $c \in C_{i0j0}$,并且 C_{i0j0}是一个最小元. 反之,对于每一个最小元 C_{i0j0},存在 $c \in \cup \mathrm{Red}_U(C \cup D)$使得 $c \in C_{i0j0}$,这意味着$(x_{i0},x_{j0}) \in C_k$ 成立. 因此,$\{C_{ij}:(x_i,x_j) \in C_k, C_k \neq \phi\}$是优势矩阵 $\boldsymbol{Q}^*(C \cup D)$的所有最小元的集合,从而可以找到优势矩阵中所有的最小元,而不用计算优势矩阵中的每个元素.

因此,可以通过定理 1,2 描述 $\mathrm{Core}_{C \cup D}(U \times U)$和样例对选择.

定理 1　对于条件属性 c,$\mathrm{Core}_{C \cup D}(U \times U) = \{(x_i,x_j):C_k = \{(x_i,x_j)\}\}$.

定理 2　如果 $S \cap C_k \neq \phi$, $\forall C_k \neq \phi$,那么 $S \subset U \times U$ 包含 S^* 的一个样例对选择.

因此,在不计算优势矩阵的前提下,样例对选择的方法定义如下:

定义 14(样例对选择方法)　假设 $S^* = (U, C \cup D)$是一个决策系统,$f_D(U,C) = \wedge_{C_k \neq \phi}(\vee C_k)$称作 S^* 的优势关系函数,其中,C_k 中的每个元素被认为是布尔变量.

令 $g_D(U,C)$是 $f_D(U,C)$经过合取、析取后得到的最简形式,存在 t 和 $S_k \subseteq \mathrm{DIS}(\{C\})$,$k = 1 \cdots t$,使得 $g_D(U,C) = (\wedge S_1) \vee \cdots \vee (\wedge S_t)$,并且有 $\mathrm{Sel}_{C \cup D}(U \times U) = \{S_1 \cdots S_t\}$. S_k 中的样例对和辨识矩阵中的最小元素 C_{ij}存在一一对应的关系,根据其中一个 S_k,便可以得到辨识矩阵中的所有最小元,进而得到所有的约简.

根据定义 14,便可得到原决策系统中所有的样例对选择方案,根据其中一个方案便可得到所有的约简. 但现实生活中,没有必要求取决策系统的所有约简,只需得到一个次优的约简即可. [23]中基于样例对选择的约简加速算法是基于等价关系的,在处理连续属性值时需先将其离散化,不能充分保留连续值或有序值的信息,因此,在上述概念的基础上,直接对[23]中的加速算法进行修改,设计了基于样例对的矩阵算法来求取原决策系统的一个约简,该算法在不计算优势矩阵的前提下,通过选择样例对,找出优势矩阵中对约简有用的属性,因此可以有效节省约简算法的时间耗费. 具体算法如下所示:

算法 1　基于优势-等价关系的样例对矩阵约简算法

输入:U,C

输出:REDUCT

初始化:REDUCT=ϕ

Step 1:计算每一个 $DIS(\{c\})$和 $DIS(C)$.

Step 2:将 $DIS(C)$中的样例对按其出现次数进行升序排序.

Step 3:Do while($DIS(C) \neq \phi$)

3.1 选择第一个样例对$(x_{i0},x_{j0}) \in DIS(C)$;

3.2 计算 $Attr = \cup\{b \in C:(x_{i0},x_{j0}) \in DIS(\{b\})\}$;

3.3 $\forall c \in Attr$ 计算 $DIS(\{c\})$的基数,将基数最大的一个 $c^* \in Attr$ 加入到 REDUCT 中;

3.4 更新 $DIS(C) = DIS(C) - DIS(\{c^*\})$.

Step 4:如果 REDUCT 独立,输出 REDUCT;否则,删除 REDUCT 中的冗余属性后输出 REDUCT.

通过对上述算法进一步分析发现,在处理较大规模的数据集时,该算法中 Step 4 耗时较大,原因是在判断约简独立性时,频繁调用 Step 4 之前的函数. 实际上,在不强调得到真正的约简时,求得一个近似的约简即可,由此,设计了一个近似的约简算法,即算法 2,该算法不再对 REDUCT 的独立性进行判断. 根据依赖度求得决策表中每个条件属性的重要性,然后,在核属性的基础上,按属性重要性大小加入属性到 REDUCT 中,直至保持了原信息系统的分辨能力,具体算法如下:

算法 2

输入:U,C

输出:REDUCT

初始化:REDUCT=ϕ

Step 1:计算每一个 $DIS(\{c\})$和 $DIS(C)$.

Step 2:将 $DIS(C)$中的样例对按其出现次数进行升序排序.

Step 3:根据定义 8 计算每个 $c \in C$ 的 SGF 值.

Step 4:Do while($DIS(C) \neq \phi$)

4.1 选择第一个样例对$(x_{i0},x_{j0}) \in DIS(C)$;

4.2 计算 $Attr = \cup\{b \in C:(x_{i0},x_{j0}) \in DIS(\{b\})\}$;

4.3 $\forall c \in Attr$ 将 SGF 值最大的一个 $c^* \in Attr$ 加入到 REDUCT 中;

4.4 更新 $DIS(C) = DIS(C) - DIS(\{c^*\})$.

Step 5:输出 REDUCT.

算法 2 既抽取了重要的条件属性,又保持了原信息系统的分辨能力,在算法 1 的基础上进一步节省了约简的时间耗费.

3 实验结果与分析

在本节中,为了验证所提算法的有效性,从 UCI 数据集中选取 10 组数据进行实验,实验环境为 Windows10,Intel(R)Core(TM)i5-6600K CPU @ 3.50 GHz 3.50 GHz 16.00 GB 64 位操作系统,基于 X64 的处理器,编程语言为 Matlab(R2015b). 实验结果为 10 次结果的平均值.

传统方法、算法 1 和算法 2 在 10 组数据集上约简后的属性个数平均值分别为 8.5,7.2,8.5,三者相差不明显,在这种前提下,比较 3 种方法的运行时间.

分三部分对传统方法(基于优势关系下辨识矩阵求约简)、优势-等价关系下基于样例对的矩阵约简算法(简称算法 1)、近似约简算法(简称算法 2)进行了 3 个阶段上运行时间的对比,这 3 个阶段分别为 Stage 1:求正域;Stage 2:已知正域求约简;Stage 3:算法总时间.

表 1-表 2 给出了 3 种方法求约简的具体时间耗费(单位:s)以及所提方法与传统方法相比约简时间的缩减率. 由于算法 1 与算法 2 求正域时所用方法相同,因而时间耗费相同. 由表 1 可以看出,所提方法与传统方法相比,时间优势明显,3 种方法在 Stage 1 中,时间耗费区别不大,可见,主要的时间耗费在 Stage 2 上,这直接导致了求约简的总时间的差距,由此可知,所提的两种方法在求约简的过程中,由于采用的矩阵中只对有用的样例对进行运算,从而明显降低了时间耗费. 表 2 给出了在 Stage 2 中具体的时间缩减率,即在 Stage 2 中的(传统算法时间-所提算法时间)/传统算法时间.

表 1　3 种方法求约简的分阶段时间比较

Table 1　Time duration comparison of three methods during different attribute reduction stages

Datasets	Stage 1			Stage 2			Stage 3		
	传统方法	算法 1	算法 2	传统方法	算法 1	算法 2	传统方法	算法 1	算法 2
Tae(151 * 6)	0.036 5	0.034 3	0.034 3	0.417 3	0.533 2	0.250 5	1.002 0	0.695 2	0.424 0
Back(212 * 28)	0.082 3	0.069 7	0.069 7	7.594 4	8.596 1	2.358 7	13.011 0	9.382 3	2.839 6
ILPD(583 * 11)	0.479 9	0.482 3	0.482 3	77.128 7	5.636 9	5.830 0	89.974 0	6.518 6	7.117 6
Bal(625 * 5)	0.591 2	0.583 9	0.583 9	3.419 4	2.555 3	3.173 2	7.421 0	3.495 8	4.413 6
Pima(768 * 9)	0.933 6	0.842 7	0.842 7	176.845 1	7.550 9	8.839 1	196.620 0	8.959 3	10.473 7
Cmc(1 473 * 10)	3.126 3	3.119 8	3.119 8	194.770 1	30.770 0	33.780 2	228.991 0	35.816 0	41.370 6
Yeast(1 484 * 9)	3.145 5	3.181 3	3.181 3	3 002.600 0	30.927 2	33.182 7	3 042.254 0	36.336 0	40.220 0
Car(1 728 * 7)	4.585 1	4.340 8	4.340 8	544.967 0	119.390 0	35.896 9	632.179 0	132.814 0	43.210 5
Wilt(4 339 * 6)	31.150 0	26.965 5	26.965 5	2 056.800 0	426.361 9	196.877 2	2 059.164 0	484.232 0	239.331 0
Aba(4 177 * 9)	29.344 6	30.351 5	30.351 5	571.865 1	730.659 3	306.682 6	805.120 0	816.419 0	363.853 5
Average	7.347 5	6.997 2	6.997 2	663.640 7	136.298 1	62.687 1	707.573 6	153.466 8	75.325 4

由表 2 可以看出,根据已知的正域求约简时,算法 1 在 ILPD 、Pima、Yeast 3 个数据集与传统方法相比,时间缩减率达到了 90%以上;算法 2 在 ILPD 、Pima 、Yeast 、Car 、Wilt 5 个数据集上时间缩减率也在 90%以上,表明所提算法的有效性. 但在少数几个数据集,如 Tae 、Back 和 Aba 上算法 1 时间比传统算法有所增加,可能的原因是当属性取值和样例较多时,导致能够被区分的样例对数量较大,使得算法 1 中的独立性判别运算耗费较大.

表 3 给出了所提方法与传统方法相比,在求约简时总的时间缩减率. 可以看出,算法 1 在 ILPD、Pima、Yeast 3 个数据集中缩减率达到了 90%以上,算法 2 在 ILPD 、Pima、Yeast、Car 4 个数据集上时间缩减率也在 90%以上,并且,除 Aba 数据集外,所提两种算法均优于传统方法. 为了更直观地表示,给出了 3 种方法在 3 个阶段的折线图,其中,横坐标为实验数据集,纵坐标为 cpu 运行时间,method 1、method 2、method 3 分别代表传统方法、算法 1、算法 2,Cpu1、Cpu2、Cpu3 分别代表 Stage 1、Stage 2、Stage 3 的时间,具体如图 1 所示.

表 2　算法 1 和算法 2 与传统方法在 Stage 2 上的时间缩减率

Table 2　The time reduction rates of three methods during stage 2

Datasets	算法 1 相对传统算法/%	算法 2 相对传统算法/%
Tae(151 * 6)	-27.77	39.97
Back(212 * 28)	-13.19	68.94
ILPD(583 * 11)	92.69	92.44
Bal(625 * 5)	25.27	7.20
Pima(768 * 9)	95.73	95.00
Cmc(1473 * 10)	84.20	82.66
Yeast(1484 * 9)	98.97	98.89
Car(1728 * 7)	78.09	93.41
Wilt(4339 * 6)	79.27	90.43
Aba(4177 * 9)	-27.77	46.37
Average	65.80625	75.8

表 3　算法 1 及算法 2 与传统方法在 Stage 3 上的时间缩减率

Table 3　The time reduction rates of three methods during stage 3

Datasets	算法 1 相对传统算法/%	算法 2 相对传统算法/%
Tae(151 * 6)	30.62	57.68
Back(212 * 28)	27.89	78.18
ILPD(583 * 11)	92.76	92.09
Bal(625 * 5)	52.89	40.53
Pima(768 * 9)	95.44	94.67
Cmc(1473 * 10)	84.36	81.93
Yeast(1484 * 9)	98.81	98.68
Car(1728 * 7)	78.99	93.16
Wilt(4339 * 6)	76.48	88.38
Aba(4177 * 9)	-1.40	54.81
Average	63.684	78.011

由于算法 1 与算法 2 在求正域时时间相同,因此图 1 中描述 method 2 和 method 3 的两条折线重合,且在数据集规模较小时,3 种方法所用时间几乎相同,但随着数据集的增大,所提方法与传统方法相比,稍显优势.

由图 2 可知,随着数据集的规模的增大,所提出两种算法均优于传统方法,而对于算法 1 来说,除 Aba 外,时间也都明显优于传统方法,由此可知,它们在求约简的总时间上效果也很显著,具体如图 3 所示.

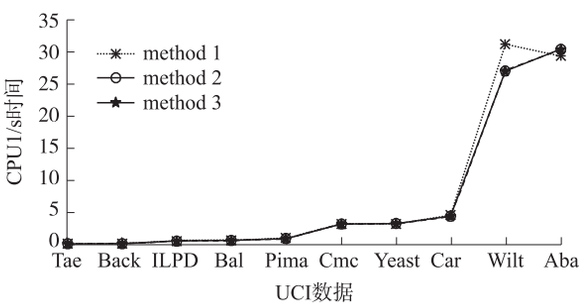

图 1　Stage 1 中 3 种方法时间图

Fig. 1　The time plot of three methods during stage 1

由图 3 及表 1 可以看出,算法 1 在 Car、Wilt、Aba 这 3 个数据集上时间耗费明显增加,尤其在 Aba 上,所用时间甚至多于传统方法,原因是随着数据规模的增加,矩阵算法在判断约简是否独立时耗时较大,为克服这一问题,提出了算法 2,该算法在 Car、Wilt、Aba 耗时明显小于算法 1,且图 3 中算法 2 所对应的曲线随着数据集规模的增加,上升的更为平缓,由此可知,算法 2 在数据集规模增大时,明显优于算法 1.

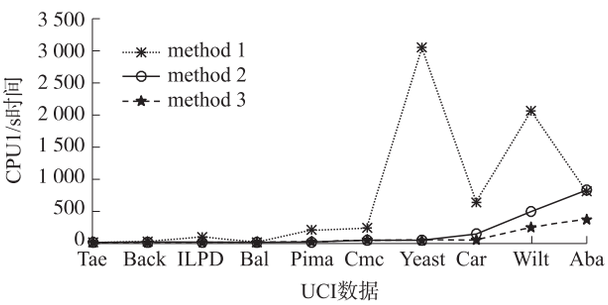

图 2　Stage 2 中 3 种方法时间图

Fig. 2　The time plot of three methods during stage 2

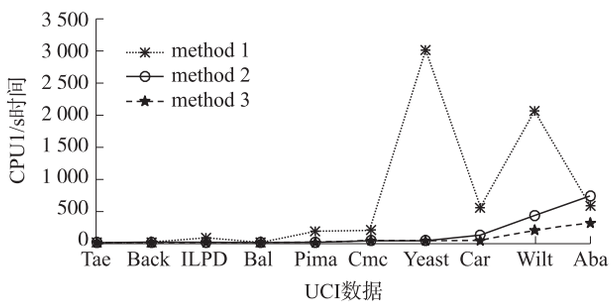

图 3　Stept 3 中 3 种方法时间图

Fig. 3　The time plot of three methods during stage 3

4　结束语

本文针对多准则分类问题中的目标信息系统,考虑了优势-等价关系下的属性约简问题,提出了基于样例对的矩阵约简算法,该算法在不求优势矩阵的前提下找出矩阵中对约简有用的元素所对应的样例对,进而得到约简,克服了传统方法中辨识矩阵存储空间大、计算复杂、时间耗费大的缺点,大大减少了存储空间及处理时间. 然而,在处理较大规模的数据集时,这种算法的耗时仍然较大,为克服这一问题,提出了一种近似约简算法,该方法随着数据规模的增大,时间耗费的增长较为平缓,且实验结果也表明该算法的有效性与可行性. 未来工作将进一步考虑在此基础上对动态信息系统的研究.

[参考文献]

[1] PAWLAK Z. Rough sets[J]. International journal of information and computer sciences,1982,11(3):341-356.

[2] PAWLAK Z. Rough sets:theoretical aspects of reasoning about data[M]. Boston:Kluwer Academic Publishers,1991.

[3] 苗夺谦,李道国. 粗糙集理论、算法与应用[M]. 北京:清华大学出版社,2008.

[4] ZHANG Q,XIE Q,WANG G. A survey on rough set theory and its applications[J]. CAAI transactions on intelligence technology,2016,1(4):323-333.

[5] YAO J T,LINGRAS P,WU W Z,et al. Rough sets and knowledge technology[C]//Proceeding of the 2nd Canadian. Toronto:Rough Sets Technology,2007.

[6] CHENG Q,QI Z,ZHANG G,et al. Robust modeling and prediction of thermally induced positional error based on grey rough set theory and neural networks[J]. The international journal of advanced manufacturing technology,2016,83(5):1-12.

[7] PéREZ D N,RUANO O D,FDEZ R F,et al. Boosting accuracy of classical machine learning antispam classifiers in real scenarios by applying rough set theory[J]. Scientific programming,2016,2016(3/4):1-10.

[8] PENG L,NIU R,HUANG B,et al. Landslide susceptibility mapping based on rough set theory and support vector machines:a case of the three gorges area,China[J]. Geomorphology,2014,204(1):287-301.

[9] 王珏,苗夺谦,周育健. 关于 Rough Set 理论与应用的综述[J]. 模式识别与人工智能,1996,9(4):337-344.

[10] 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.

[11] 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):12-18.

[12] KRYSZKIEWICZ M. Comparative studies of alternative of knowledge reduction in inconsistent systems[J]. Intelligent systems,2001,16(1):105-120.

[13] GRECOS,MATARAZZO B,SLOWINSKI R. Rough approximation of preference relation by dominance relations[J]. European journal of operational research,1999,117(1):63-83.

[14] 徐伟华,张文修. 基于优势关系下不协调目标信息系统的知识约简[J]. 计算机科学,2006,33(2):182-184.

[15] 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.

[16] STEFANOWSKI J. Handling continuous attributes in discovery of strong decision rules[M]. Berlin:Springer Berlin Heidelberg,1998.

[17] SAI Y,YAO Y Y,ZHANG N. Data analysis and mining in order information table[C]//IEEE International Conference on Data Mining. USA:IEEE Computer Society Press,2001:497-504.

[18] SHAO M W,ZHANG W X. Dominance relation and rules in an incomplete ordered information system[J]. International journal of intelligent systems,2005,20:13-27.

[19] 张文修,姚一豫,梁怡. 粗糙集与概念格[M]. 西安:西安交通大学出版社,2006.

[20] 苟光磊,王国胤. 基于不协调置信优势原理关系的知识约简[J]. 计算机科学,2016,43(6):204-207.

[21] 黄琴,魏玲. 基于布尔矩阵的序信息系统属性约简方法[J]. 小型微型计算机系统,2016,37(8):1 717-1 720.

[22] 贺明利,魏玲. 基于优势关系的序形式背景约简[J]. 计算机科学,2015,42(6):46-49.

[23] CHEN D G,ZHAO S Y,ZHANG L,et al. Sample pair selection for attribute reduction with rough set[J]. IEEE Trans Know Data Eng,2012,24(1):2 080-2 093.

[责任编辑:黄　敏]