

优势关系下基于浓缩布尔矩阵的属性约简方法

李 艳 郭娜娜 吴婷婷 湛 燕

(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)

摘 要 在优势关系粗糙集方法(DRSA)的框架下,针对不协调的目标信息系统求属性约简。基于优势矩阵的方法是最常用的一类约简方法,但矩阵中不是所有的元素都有效。浓缩优势矩阵只保留对求约简有用的最小属性集,因而可以明显降低约简过程中的计算量。进一步地,浓缩布尔矩阵通过布尔代数的形式有效地弥补了优势矩阵生成效率低的缺点。文中将等价关系上的浓缩布尔矩阵属性约简方法扩展到优势关系上,针对优势矩阵提出了浓缩布尔矩阵的概念,建立了相应的高效约简方法,使效率得到明显提高。最后采用 9 组 UCI 数据进行实验,结果验证了所提方法的有效性。

关键词 粗糙集,属性约简,优势关系,浓缩优势矩阵,浓缩布尔矩阵

中图法分类号 TP181 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2018.10.042

Attribute Reduction Based on Concentration Boolean Matrix under Dominance Relations

LI Yan GUO Na-na WU Ting-ting ZHAN Yan

(Key Lab of Machine Learning and Computational Intelligence, College of Mathematics and Information Science, Hebei University, Baoding, Hebei 071002, China)

Abstract Under the framework of dominance relation-based rough set approach (DRSA), attribute reduction was studied for inconsistent target information systems. The methods based on dominance matrix are the most commonly used ones, but not all elements in the matrix are valid. The concentration dominance matrix only preserves the smallest set of attributes which are useful for attribute reduction, and thus the computational complexity can be significantly reduced. On the other side, the concentration Boolean matrix further improves the generation efficiency of the dominance matrix by Boolean algebra. This paper extended the concentration Boolean matrix method under equivalence relations to that under dominance relations. The concept of concentration Boolean matrix was proposed for the dominance matrix, and the corresponding efficient reduction method was established to improve the efficiency of the reduction algorithm. Finally, nine UCI data sets were used in the experiments, and the results show the feasibility and effectiveness of the proposed method.

Keywords Rough set, Attribute reduction, Dominance relation, Concentration dominance matrix, Concentration boolean matrix

1 引言

Rough 理论^[1]是由 Pawlak 教授于 1982 年提出的一种用于处理不精确、不完全与不相容知识的数学理论,近年来被广泛应用于机器学习、数据挖掘及模式识别等多个领域。属性约简^[2-8]是粗糙集理论的核心问题之一,其中最常用且高效的方法是基于辨识矩阵的方法^[9-12]。所谓属性约简,就是在保持知识库分类能力不变的条件下,发现并去除其中不相关或不重要的属性,从而使知识表示简化,又不丢失基本信息。目前为止,许多学者通过不同的方法从不同的角度对知识约简进行了深入的研究,并取得了很多成果。但这些研究主要是在等价关系下的信息系统中进行的,而在实际问题中有一些属性值是有序的,比如综合评判问题中的“得分”属性,它的属性值分为“高、中、低”,这时往往不能忽视属性值的偏好性信息。为了解决这个问题,Greco 等^[13-15]提出了基于优势关系的粗糙集方法(DRSA),用优势关系代替等价关系来描述属性值的偏好顺序,之后在这个框架下取得了一系列的研究成果。文献[16-18]定义了优势关系下协调目标信息系统的约简,并给出了相应的优势矩阵方法。徐伟华等对优势关系下的不协调目标信息系统做了深入研究,引入了协调近似空间

到稿日期:2017-08-16 返修日期:2017-12-03 本文受国家自然科学基金(61170040,61473111),河北大学自然科学研究计划项目(799207217069)资助。

李 艳(1976—),女,博士,教授,CCF 会员,主要研究方向为机器学习、Rough 集理论、计算智能,E-mail:ly@hbu.cn(通信作者);**郭娜娜**(1992—),女,硕士生,主要研究方向为粒计算与知识发现,E-mail:1498976702@qq.com;**吴婷婷**(1989—),女,硕士生,主要研究方向为粒计算与知识发现;**湛 燕**(1978—),女,副教授,主要研究方向为机器学习。

的概念^[19-20],随后进一步引入了分布约简和最大分布约简的概念^[21-22],并得到了分布及最大分布约简的判定定理以及优势矩阵^[23-24]。不难得知,优势矩阵是优势关系下约简的最常用的方法。然而在优势矩阵中并不是所有的属性集都对约简有效,如果在计算约简时遍历整个优势矩阵将造成空间和时间的浪费。因此,本文把等价关系下的浓缩辨识矩阵方法^[25]扩展到优势关系下,基于优势矩阵的概念提出了相应的浓缩矩阵,其中只保留进行约简时必要的属性集。在进行约简时只对浓缩优势矩阵进行遍历,不仅节省了存储空间,而且提高了约简的效率。进一步,在此基础上引入浓缩布尔矩阵^[26-28],通过布尔代数的形式有效地弥补了现有优势矩阵生成效率低的缺点,进一步提高了属性约简的效率。在 9 组 UCI 数据上,将传统的优势关系下的优势矩阵方法和浓缩优势矩阵方法的约简效率作比较,实验结果表明了优势关系下浓缩布尔矩阵的属性约简的可行性和有效性。

2 基本概念

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; $A=C\cup D\neq\emptyset$ 是一个有限属性集,其中 C 为条件属性集, $D=\{d\}$ 为决策属性集; V 是属性值的集合; $f:U\times A\rightarrow V$,指在每个属性上对 U 中的每个对象给出一个 V 中的值,即 $\forall a\in A,x\in U,f(x,a)\in V$ 。

在实际应用中,一个信息系统通常用一个二维表来表示,其中行表示论域中的对象,列表示描述对象的属性。如果决策属性是符号值,那么这些属性基于等价关系构成论域的一个划分(记作 U/D);如果决策属性是具有优劣关系的真值,那么这些属性基于优势关系构成论域的一个覆盖。

定义 2(优势/劣势关系) 设 $S=(U,A,V,f)$ 为一个目标信息系统, $B\subseteq A$,记:

$$R_B^{\leq}=\{(x_i,x_j)\in U\times U:f(x_i,a)\leq f(x_j,a),\forall a\in B\}$$

$$R_B^{\geq}=\{(x_i,x_j)\in U\times U:f(x_i,a)\geq f(x_j,a),\forall a\in B\}$$

其中, $R_B^{\leq}$ 和 $R_B^{\geq}$ 分别为信息系统上的优势关系和劣势关系。

根据定义 2,优势类定义如下:

$$[x_i]_B^{\leq}=\{x_j\in U:(x_i,x_j)\in R_B^{\leq}\}=\{x_j\in U:f(x_i,a)\leq f(x_j,a),\forall a\in B\}$$

其表示 x_i 关于属性集 B 的优势类,它是优于 x_i 的所有对象的集合。相应地,劣势类的定义是劣于对象 x_i 的所有对象的集合。

定义 3(基于优势关系的上/下近似和正域) 设 $S=(U,A,V,f)$ 为一个目标信息系统, $B\subseteq A$,对于任意 $X\subseteq U$, X 关于优势关系 $R_B^{\leq}$ 的下近似与上近似分别为 $\underline{R}_B^{\leq}(X)=\{x_i|[x_i]_B^{\leq}\subseteq X\}$ 和 $\overline{R}_B^{\leq}(X)=\{x_i|[x_i]_B^{\leq}\cap X\neq\emptyset\}$ 。

类似地,我们也可以定义 X 关于劣势关系的近似集。进一步,基于优势关系的条件属性集分类相对于决策属性集分类的正域被定义为 $POS_B^{\leq}(D)=\bigcup_{X\in U/D} \underline{R}_B^{\leq}(X)$,正域仍旧是相应关系下可以完全确定划分到某一决策类的所有元素的集合。

定义 4(正域约简) 设 $S=(U,A,V,f)$ 为一个目标信息

系统,对于 $B\subseteq A$,记 $POS_A^{\leq}(D)=\bigcup_{X\in U/D} \underline{R}_A^{\leq}(X)$ 。若 $POS_A^{\leq}(D)=POS_B^{\leq}(D)$,则 B 为正域协调集。若 B 是正域协调集,但 B 的任何真子集都不是正域协调集,则称 B 是正域约简。

定义 5(属性的重要性度量) 对于优势关系下的属性 $a(a\in B\subseteq C)$,其重要性表示为 $SGF(a,B,D)=\frac{|POS_B^{\leq}(D)|-|POS_{B/a}^{\leq}(D)|}{|U|}$,当属性 a 的重要性 $SGF(a,B,D)=0$ 时,该属性对于属性集 B 而言是不重要的,可以约去;否则,属性 a 相对于 B 而言是重要的,不可以约去。

求属性约简最常用且高效的方法是基于优势矩阵的方法,其定义如下。

定义 6(优势矩阵) 设 $S=(U,A,V,f)$ 为一个目标信息系统,其优势矩阵是一个 $n\times n$ 的矩阵,记为 Q 。矩阵元素定义为:

$$D^{\leq}(x_i,x_j)=\{a\in C|f(x_i,a)<f(x_j,a)\wedge f(x_i,d)\neq f(x_j,d)\}$$

其中, $f(x_i,a)$ 表示对象 x_i 在属性 a 上的属性值。

优势关系下基于优势矩阵的属性约简类似于等价关系上基于辨识矩阵的属性约简,首先通过矩阵中单个元素的并找到核属性集,再根据属性的重要性依次在核中加入重要的属性,直到与优势矩阵 Q 中的每个元素的交集都不为空为止,此时得到的即为原始属性的约简。但这只是针对于协调的目标信息系统,对于不协调的目标信息系统,优势矩阵的定义如下。

定义 7(优势关系下的正域优势矩阵) 设 $S=(U,A,V,f)$ 为不协调目标信息系统,记:

$$D^{\leq}(x_i,x_j)=\begin{cases} \{a_k\in A,f(x_i,a_k)>f(x_j,a_k)\}, & x_i\in POS_B^{\leq}(D),x_j\in U \\ \emptyset, & \text{else} \end{cases}$$

称 $D^{\leq}(x_i,x_j)$ 为 x_i 与 x_j 的正域优势属性集;称矩阵 $Q=(D^{\leq}(x_i,x_j),(x_i,x_j)\in U)$ 为该目标信息系统的正域优势矩阵。

3 优势关系下基于浓缩布尔矩阵的属性约简

与传统的优势矩阵的不足相同,生成优势矩阵和正域优势矩阵的计算代价较高,会产生不必要的重复的矩阵元素,且并不是所有的元素都对求约简有效,从而造成了时间和存储的浪费。为了避免生成不必要的元素,本文定义了优势关系下的浓缩优势矩阵。

定义 8(优势关系下的浓缩优势矩阵) 浓缩优势矩阵是正域优势矩阵 Q 中一些属性集的集合,记为 $Q_{\min}$ 。 $Q_{\min}$ 中的属性集不存在两两包含的关系,即:

$$Q_{\min}=\{D|D(D\neq\emptyset)\in Q,\forall D'(D'\in Q)\not\subseteq D\}$$

其中, D' 为 D 的补集。

浓缩布尔矩阵是在浓缩差别矩阵的基础上提出的,它是用布尔代数的形式代替符号值,从而进一步减小矩阵的存储空间,有效改善了算法的时间和空间复杂度,提高了属性约简的效率。本文把基于浓缩布尔矩阵的属性约简的思想用于优势关系下的不协调目标信息系统,将文献[25]中基于等价关系的浓缩布尔矩阵扩展为优势关系下的布尔矩阵、浓缩布尔矩阵。

定义 9(优势关系下的布尔矩阵) 布尔矩阵 BQ 中每个元素的定义为:

$$D^{\leq}((x_i, x_j), C_k) = \begin{cases} 1, & \text{当 } f(x_i, D) \neq f(x_j, D) \text{ 且 } f(x_i, C_k) > f(x_j, C_k), \\ & x_i \in POS_C^{\leq}(D), x_j \in U \\ 0, & \text{其他} \end{cases}$$

其中, $i=1,2,3,\cdots,n, j=1,2,3,\cdots,n, k=1,2,3,\cdots,m$ 。

该布尔矩阵的含义为: $D^{\leq}((x_i, x_j), C_k) = 1$ 表示属性 C_k 能区分对象 x_i 与 x_j , $D^{\leq}((x_i, x_j), C_k) = 0$ 表示属性 C_k 不能区分对象 x_i 与 x_j 。即在布尔矩阵中, 仅用相应位置上的 m 位 01 串(m 为属性的个数)来表示区分相应对象的属性集。

当然, 布尔矩阵只是优势关系下的优势矩阵的 01 串表示, 并未去除约简时不必要的属性。下面为了引入浓缩布尔矩阵, 首先分析如何判别布尔矩阵中属性子集之间的包含关系。

设 d' 和 d 均为 C 的子集, 判断 d' 是否为 d 的子集的方法为让 d' 和 d 的各位做或运算, 其运算结果可能有 3 种情况:

- 1) 结果各位与 d' 的各位相同, 则说明 $d \subset d'$;
- 2) 结果各位与 d 的各位相同, 则说明 $d' \subset d$;
- 3) 结果与 d 和 d' 都不相同, 则说明 d 和 d' 不存在包含与被包含的关系。

定义 10(优势关系下的浓缩布尔矩阵) $IME(BQ) = \{d | d(d \neq \emptyset) \in BQ\}$ 且不存在 $d'(d' \neq \emptyset) \in BQ$ 使得 d 与 d' 各位进行或运算的结果与 d 的各位相同。

浓缩布尔矩阵只需要利用 0,1 之间的或运算就可以快速判断出矩阵元素之间的包含关系, 选出影响计算约简的矩阵元素并进行存储, 大大提高了优势矩阵的生成效率。

在以上定义的基础上, 为了实现快速约简, 本文提出了以下两个约简算法, 其均是在已知 D 的 C -正域的基础上, 简化优势矩阵的存储和计算。

算法 1 优势关系下的浓缩优势矩阵的属性约简算法

输入: $POS_C^{\leq}(D), U, data$

输出: 属性约简 $reduct$

Step 1 求浓缩优势矩阵与核

$Q_{min} = \emptyset, core = \emptyset$ //初始化浓缩优势矩阵与核

```
for  $x_i \in POS_C^{\leq}(D)$  do
for  $x_j \in U$  do{
if  $data(x_i, end) = data(x_j, end)$  then continue;
生成  $Q$  中的元素  $D^{\leq}(x_i, x_j)$ , 简记为  $m$ ;
if  $length(m) = 1$  then{
 $core = core \cup m$ ;
continue;} /* 如果  $m$  是点单集, 即是核属性, 则直接把它放入  $core$  中, 然后进行下一个  $m$  的求解 */
if  $\sim isempty(intersect(m, core))$  then continue; else /* 如果  $m$  不是单点集并且与核的交集不为空, 那么此  $m$  无用, 进行下一个  $m$  的求解 */
{
if  $Q_{min}$  中存在被  $m$  包含的元素 then 此  $m$  没用, 进行下一次  $m$  的求解;
else {
if  $Q_{min}$  中不存在包含  $m$  的元素 then
```

```
 $Q_{min} = Q_{min} \cup m$ ;
else {
删除  $Q_{min}$  中所有包含  $m$  的元素;
 $Q_{min} = Q_{min} \cup m$ 
}
}
```

去掉核中在 Q_{min} 中与 $core$ 中属性有交集的所有属性集; /* 因为此时的浓缩优势矩阵不是最简的, 所以需要进行这一步, 此时求解浓缩优势矩阵的过程结束 */

Step 2 在浓缩优势矩阵与核的基础上求约简

```
求核属性之外的每个属性的重要性  $sig(C')$ ;
 $(sig'(C'), index) = sort(sig(C))$ ; /* 按照属性的重要性排序 */
 $reduct = core$ ; /* 初始化  $reduct$  */
for  $Q_{min} \cap reduct \neq \emptyset$  do  $reduct = reduct \cup C'(index(i))$ ;
/* 按照属性重要性由高到低依次加入  $reduct$  直到  $reduct$  与  $Q_{min}$  中所有元素的交集为空 */
```

算法 2 优势关系下的浓缩布尔矩阵的属性约简算法

输入: $POS_C^{\leq}(D), U, data$

输出: 属性约简 $reduct$

Step 1 求浓缩布尔矩阵与核

$IME(BQ) = \emptyset, core = \emptyset$ //初始化浓缩布尔矩阵与核

```
for  $x_i \in POS_C^{\leq}(D)$  do
for  $x_j \in U$  do{
if  $data(x_i, end) = data(x_j, end)$  then continue;
生成  $BQ$  中的元素  $D^{\leq}((x_i, x_j), C_k)$ , 简记为  $d_{ij}$ ; /* 假设目标信息系统  $data$  是  $m$  维的, 则  $d_{ij}$  就是  $m$  位的 01 串 */
/* 如果  $d_{ij}$  只有一个 1, 那么这个 1 所对应的属性就是核, 同时把核属性所对应的下标存入  $core$ , 再进行下一个  $d_{ij}$  的求解 */
if  $sum(d_{ij}) = 1$  then
 $core = core \cup find(d_{ij} = 1)$ ;
continue;
if  $IME(BQ) = \emptyset$  then  $IME(BQ) = IME(BQ) \cup d_{ij}$ ;
else{
 $d_{ij}$  与  $IME(BQ)$  的每行(记为  $d$ )做“与运算”, 每次的结果记为  $d'$ ;
if  $d'$  的各位与  $d_{ij}$  的各位相同 then 得出的  $d_{ij}$  无效;
if  $d'$  的各位与  $d$  的各位相同 then  $IME(BQ) = IME(BQ) \cup d_{ij}$ ;
if  $d'$  的各位与  $d$  和  $d_{ij}$  都不相同 then  $IME(BQ) = IME(BQ) \cup d_{ij}$ ;
}
```

去掉核中属性在 $IME(BQ)$ 中值为 1 的所有行;

Step 2 在浓缩布尔矩阵与核的基础上求约简

```
求核属性之外的每个属性的重要性  $sig(C')$ ;
 $(sig'(C'), index) = sort(sig(C))$ ; /* 按照属性重要性排序 */
 $reduct = core$ ; /* 初始化  $reduct$  */
for  $sum(IME(BQ) \& reduct) \geq 1$  do
 $reduct = reduct \cup C'(index(i))$ ;
/* 按照属性重要性由高到低依次加入  $reduct$ , 直到与  $IME(BQ)$  中的每一行做与运算的结果中至少有一个 1 为止 */
```

可以看出, 在约简的过程中, 浓缩优势矩阵使用集合相交的方式判断是否是约简, 而浓缩布尔矩阵使用二进制串“与”运算的方式来进行判断。因此, 后者相比前者, 无论在时间和空间上都有着更低的耗费, 使其在较大数据集上更具优势。

4 实验结果与分析

本节的实验主要比较优势关系下的 3 种属性约简方法，分别为传统的优势矩阵方法、基于浓缩优势矩阵的方法、基于浓缩布尔矩阵的方法。数据来源于 UCI 数据集^[26]，共有 9 组数据，其详细信息如表 1 所列。

表 1 实验数据
Table 1 Experimental data

UCI 数据名称	属性值类型	样例个数	条件属性个数	决策类
kohkiloyeh	real	100	5	2
tae	symbol	151	5	3
backuplarge	symbol	212	27	12
ILPD	real	583	10	2
balancescal	symbol	625	4	3
cmc	symbol	1473	9	3
yeast	real	1484	8	10
wilt	real	4339	5	2
magic04	real	19020	10	2

本实验所比较的 3 种属性约简方法的说明如下：第 1 种方法是由定义 7 提出的属性约简方法，称为传统优势矩阵方法；第 2 种方法是由定义 8 提出的浓缩优势矩阵方法（即算法 1）；第 3 种方法是由定义 10 提出的本文重点研究的方法（即算法 2），即把基于浓缩布尔矩阵的属性约简方法应用在优势关系下，称为浓缩布尔矩阵方法，以期能够减少存储空间和计算量，提高约简的运行效率。实验的运行环境为 MATLABR 2015b，运行内存是 16GB。表 2 列出了传统方法和浓缩布尔矩阵方法求属性约简的运行时间，表 3 列出了浓缩优势矩阵和浓缩布尔矩阵方法求属性约简的运行时间。其中， α_1 表示浓缩布尔矩阵方法相比于传统优势矩阵方法所用时间的缩减率， β_1 表示浓缩布尔矩阵方法相比于浓缩优势矩阵方法所用时间的缩减率。

$$\alpha_1 = (\text{传统优势矩阵方法运行时间} - \text{浓缩布尔矩阵方法运行时间}) / \text{传统优势矩阵方法运行时间}$$

$$\beta_1 = (\text{浓缩优势矩阵方法运行时间} - \text{浓缩布尔矩阵方法运行时间}) / \text{浓缩优势矩阵方法运行时间}$$

从表 2 和表 3 中的结果可以看出，由于浓缩优势矩阵能够去掉无效的冗余元素，因此可以明显降低搜索的时间，大大提高了传统优势矩阵方法的效率；而浓缩布尔矩阵能够在此基础上利用布尔运算的优势进一步提高加速的效果。

表 2 传统方法与浓缩布尔矩阵方法求属性约简的时间

Table 2 Attribute reduction time of traditional method and concentration boolean matrix method

UCI 数据名称	传统方法/s	浓缩布尔矩阵/s	$\alpha_1/\%$
kohkiloyeh	0.460	0.209	54.57
tae	1.002	0.389	61.18
backuplarge	13.011	5.185	60.15
ILPD	89.974	8.640	90.39
balancescal	7.421	3.564	51.97
cmc	228.991	36.413	81.10
yeast	3042.254	41.228	98.64
wilt	2059.164	193.212	90.62
magic04	内存溢出	12225.149	1
平均值	—	1390.450	76.85

表 3 浓缩优势矩阵与浓缩布尔矩阵方法求约简的时间

Table 3 Attribute reduction time of concentration dominance matrix method and concentration boolean matrix method

UCI 数据名称	浓缩优势矩阵/s	浓缩布尔矩阵/s	$\beta_1/\%$
kohkiloyeh	0.316	0.209	33.86
tae	0.736	0.389	47.15
backuplarge	7.662	5.185	32.33
ILPD	11.605	8.640	25.55
balancescal	3.653	3.564	2.40
cmc	42.461	36.413	14.24
yeast	60.082	41.228	31.28
wilt	206.272	193.212	6.30
magic04	12410.1	12225.149	14.90
平均值	1415.876	1390.450	23.11

下面对算法进行更加精细的分析。浓缩优势矩阵相对于传统优势矩阵的具体改进之处是去掉了优势矩阵中的冗余元素，只保留必要的少量元素，因此明显加快了之后的约简计算。表 2、表 3 中的算法时间包含了每种算法均有的计算正域和属性的重要性的过程。为充分体现所提方法的效果，下面仅比较改进部分的运行时间，即在 3 种方法已分别计算得到优势矩阵、浓缩优势矩阵、浓缩布尔优势矩阵之后，到求得属性约简所需要运行的时间。表 4、表 5 列出了 3 种方法在改进后进行约简的运行时间的比较结果。其中， α_2 表示浓缩布尔矩阵方法相比于传统优势矩阵方法在改进后所用时间的缩减率， β_2 表示浓缩布尔矩阵方法相比于浓缩优势矩阵方法在此部分所用时间的缩减率。计算方法同表 2、表 3 中的说明。可以看出，相比表 2、表 3 中的结果，表 4、表 5 中的结果更加明显，这是由于大量去除了传统优势矩阵中冗余的元素，大大节省了存储空间，在之后约简的过程中只需要少量的计算就可以更加快速地得到结果。

表 4 改进后传统方法与浓缩布尔矩阵方法的约简时间

Table 4 Attribute reduction time of improved traditional method and concentration boolean matrix method

UCI 数据名称	传统方法/s	浓缩布尔矩阵/s	$\alpha_2/\%$
kohkiloyeh	0.106	0.018	83.02
tae	0.239	0.033	86.19
backuplarge	5.382	1.516	71.83
ILPD	71.003	2.243	96.84
balancescal	0.937	0.006	99.35
cmc	164.347	2.251	98.63
yeast	2993.66	6.703	99.78
wilt	1656.5	2.568	99.85
magic04	内存溢出	3973.6	1
平均值	内存溢出	423.135	81.83

表 5 改进后浓缩矩阵与浓缩布尔矩阵方法的约简时间

Table 5 Attribute reduction time of improved concentration matrix method and concentration boolean matrix method

UCI 数据名称	浓缩优势矩阵/s	浓缩布尔矩阵/s	$\beta_2/\%$
kohkiloyeh	0.074	0.018	75.68
tae	0.148	0.033	77.70
backuplarge	1.974	1.516	23.20
ILPD	3.298	2.243	31.99
balancescal	0.034	0.006	82.35
cmc	4.469	2.251	49.63
yeast	15.441	6.703	56.59
wilt	10.223	2.568	74.89
magic04	5198.2	3793.6	27.02
平均值	581.540	423.135	55.45

这 3 种方法在不同数据量的数据下得出了一个共同的结论:CPU 运行时间随着样例个数的增加呈上升趋势,但是在数据集 *balancescal* 和 *wilt* 上出现波动。原因是这两组数据的属性个数相对较少,属性约简的 CPU 运行时间不仅与样例数量有关,也与属性个数有关。

为了进一步观察 3 种方法的属性约简效率与样例的关系,图 1 给出了针对前 3 组数据在 3 种方法的属性约简下样例个数与约简所花费的时间的关系。通过复制原数据集生成几倍规模的数据集,样例呈整数倍增加。其中,横坐标表示成倍增加的 UCI 数据,纵坐标则是在相应数据下 3 种方法进行约简的 CPU 时间。

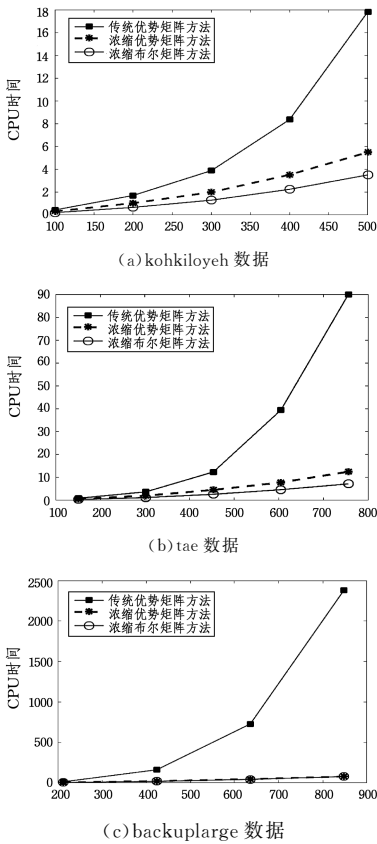


图 1 3 种属性约简方法的 CPU 时间和数据量的关系
Fig.1 Relationship between CPU time and data size for three attribute reduction methods

从图 1 中可以明显地看出,基于浓缩布尔矩阵的约简方法花费的时间最少,并且 CPU 时间随着数据量的增加而上升。

5 复杂度分析

为了更加充分地说明利用浓缩后的优势矩阵求约简的效率,表 6 列出了传统优势矩阵中元素数量和浓缩优势矩阵中元素数量的对比结果。可以看出,浓缩后元素数量大大下降,极大地节省了空间存储资源。

下面对 3 种方法约简的效率进行系统的分析。约简效率由两部分组成:1)求优势矩阵的时间复杂度;2)在优势矩阵的基础上求属性约简的时间复杂度。

对于传统优势矩阵方法,约简效率分两步求出。第一步:

首先求出优势矩阵并且存储,优势矩阵中的元素是可区分相应两个对象的条件属性的子集,而且保留了所有的可区分的情况。此步骤的时间复杂度为 $O(m * n^2)$ 。第二步:在后续求属性约简的过程中遍历整个完整的优势矩阵,时间复杂度仍为 $O(m * n^2)$ 。因此整体来说,传统方法求约简的时间复杂度为 $O(mn^2)$, m 代表属性的个数, n^2 代表优势矩阵中元素的个数。

对于浓缩优势矩阵方法,约简效率也分两步求出。第一步:求优势矩阵并进行浓缩的时间复杂度与传统方法相同,即为 $O(mn^2)$ 。第二步:利用浓缩优势矩阵求属性约简,不必扫描完整的优势矩阵,而只扫描浓缩矩阵中留下的有效元素。即在后续求属性约简的过程中遍历的是浓缩之后的优势矩阵,此步骤的时间复杂度为 $O(m * k)$, m 表示属性的个数, k 表示浓缩优势矩阵中元素的个数。从表 6 可以明显地看出, $k \ll n^2$,因此相比传统方法,该方法在第二步中利用了浓缩优势矩阵计算约简,明显加速了计算过程。因此,虽然整体的时间复杂度仍为 $O(m * n^2)$,但相比于传统方法大大地减少了计算量,提高了约简效率。

对于浓缩布尔矩阵方法,利用浓缩布尔矩阵求属性约简的时间复杂度与浓缩优势矩阵相同,为 $O(m * n^2)$ 。但它是在浓缩优势矩阵的基础上,用 01 串代替了原来的属性序号,0 表示两个对象的对应属性是可区分的,1 表示是不可区分的。在求解浓缩布尔矩阵以及约简的过程中,把对字符串的遍历、增加、删除等一系列操作替换为 01 串之间的与或运算,因为计算机可以直接处理 01 串,而对于非 01 串必须转化为 01 串才能识别,所以该方法的运算效率较高。如果条件属性的个数较多,则求解浓缩布尔矩阵的效率将远远高于求解浓缩优势矩阵。因此,该方法在空间复杂度上更具优势。

表 6 传统优势矩阵和浓缩优势矩阵中的元素数量
Table 6 Number of elements in traditional dominance matrix and concentration dominance matrix

UCI 数据名称	样例个数	传统优势矩阵中元素的个数 n^2	浓缩优势矩阵中元素个数 k
kohkiloyeh	100	100^2	5
tae	151	151^2	4
backuplarge	212	212^2	15
ILPD	583	583^2	10
balancescale	625	625^2	4
cmc	1473	1473^2	9
yeast	1484	1484^2	8
wilt	4339	4339^2	4
magic04	19020	19020^2	10
平均值	3109.67	42863062.78	7.7

结束语 本文主要针对优势关系下的属性约简的效率问题,把浓缩布尔矩阵方法扩展到优势关系上,提出了优势关系下的浓缩矩阵和布尔矩阵,并建立了相应的约简算法。由于去掉了无效的矩阵元素并用 0,1 二进制位的布尔运算代替了字符串的交运算,相比传统的优势矩阵方法和浓缩优势矩阵方法,所提方法不仅节约了存储空间而且大大减少了运行时间,提高了运行效率。

参 考 文 献

[1] PAWLAK Z. Rough set[J]. International Journal of Computer

- and Information Science, 1982, 11(5): 341-356.
- [2] QIAN Y H, LIANG J Y, PEDRYCZ W. An efficient accelerator for attribute reduction from incomplete data in rough set framework[J]. Pattern Recognition, 2011, 44(8): 1658-1670.
- [3] BLASZCZYNSKI J, GRECO S, SLOWINSKI R. Inductive discovery of laws using monotonic rules[J]. Engineering Applications of Artificial Intelligence, 2012, 25(2): 284-294.
- [4] ABBAS A, LIU J. Designing an intelligent recommender system using partial credit model and Bayesian rough set[J]. International Arab Journal of Information Technology, 2012, 9(2): 179-187.
- [5] CHANG B, HUNG H. A study of using RST to create the supplier selection model and decision making rules[J]. Expert Systems with Applications, 2010, 37(12): 8284-8295.
- [6] LIU D, YAO Y Y, LI T R. Three-way investment decisions with decision-theoretic rough sets[J]. International Journal of Computational Intelligence Systems, 2011, 4(1): 66-74.
- [7] WEI L, LI H R, ZHANG W X. Knowledge reduction based on the equivalence relations defined on attribute set and its power set[J]. Information Sciences, 2007, 177(15): 3178-3185.
- [8] CHEN X, ZIARKO W. Experiments with rough set approach to face recognition[J]. International Journal of Intelligent Systems, 2011, 26(6): 499-517.
- [9] KARAMI J, ALIMOHAMMADI A, SEIFOURI T. Water quality analysis using a variable consistency dominance-based rough set approach[J]. Computers, Environment and Urban Systems, 2014, 43: 25-33.
- [10] CHANG B, HUNG H. A study of using RST to create the supplier selection model and decision making rules[J]. Expert Systems with Applications, 2010, 37(12): 8284-8295.
- [11] LIU D, YAO Y Y, LI T R. Three-way investment decisions with decision-theoretic rough sets[J]. International Journal of Computational Intelligence Systems, 2011, 4(1): 66-74.
- [12] 刘清. Rough 集及 Rough 推理[M]. 北京: 科学出版社, 2001.
- [13] GRECO S, MATARAZZO B, SLOWINSKI R. Rough sets theory for multicriteria decision analysis[J]. European Journal of Operational Research, 2001, 129(1): 1-47.
- [14] GRECO S, MATARAZZO B, SLOWINSKI R. Rough approximation by dominance relations[J]. International Journal of Intelligent Systems, 2002, 17(2): 153-171.
- [15] GRECO S, SLOWINSKI R, MATARAZZO B, et al. Variable consistency model of dominance-based rough sets approach[C]// Proc. of International Conference on Rough Sets and Current Trends in Computing, LNAI, 2005. Springer, Berlin, 2005: 170-181.
- [16] 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, 2005.
- [17] LI S Y, LI T R, LIU D. Dynamic maintenance of approximations in dominance-based rough set approach under the variation of the object set[J]. International Journal of Intelligent Systems, 2013, 28(8): 729-751.
- [18] CHEN H M, LI T R. Parallel attribute reduction in dominance-based neighborhood rough set[J]. Information Sciences, 2016, 373(12): 351-368.
- [19] XU W H, ZHANG W X. Consistent approximation spaces based on dominance relations[J]. Computer Science, 2005, 32(9): 164-165. (in Chinese)
徐伟华, 张文修. 基于优势关系下协调近似空间[J]. 计算机科学, 2005, 32(9): 164-165.
- [20] ZHENG J Y, WANG H. Distribution Reduction of Inconsistent Interval-valued Target Information System Based on Dominance Relation[J]. The Practice and Understanding of Mathematics, 2016, 46(6): 129-135. (in Chinese)
郑俊艳, 王虹. 基于优势关系的不协调区间值目标信息系统的分布约简[J]. 数学的实践与认识, 2016, 46(6): 129-135.
- [21] BAO A N, JIANG H B. An Improved Distributed Reduction and Maximum Distribution Reduction Method[J]. Henan Science and Technology, 2014(13): 189-190. (in Chinese)
鲍爱娜, 姜洪冰. 一种改进的分布约简与最大分布约简求法[J]. 河南科技, 2014(13): 189-190.
- [22] FANG L H, LI K D. Distribution Reduction Based on Unequal Target Information System under Dominant-Equivalent Relation[J]. Fuzzy Systems and Mathematics, 2013, 27(3): 182-189. (in Chinese)
方连花, 李克典. 基于优势-等价关系下不协调目标信息系统的分布约简[J]. 模糊系统与数学, 2013, 27(3): 182-189.
- [23] XIE W Q, LIN G P. Distribution Reduction of Interval-valued Target Information System under Dominant-Equivalent Relation[J]. Journal of Qiqihar University, 2016, 32(6): 72-77. (in Chinese)
谢文琼, 林国平. 优势-等价关系下区间值目标信息系统的分布约简[J]. 齐齐哈尔大学学报, 2016, 32(6): 72-77.
- [24] CHEN D G, ZHAO S Y, ZHANG L, et al. Sample pair selection for attribute reduction with rough set[J]. IEEE Transactions on Knowledge and Data Engineering, 2012, 24(11): 2080-2093.
- [25] YANG M, YANG P. Discernibility matrix enriching and computation for attributes reduction[J]. Computer Science, 2006, 33(9): 181-183. (in Chinese)
杨明, 杨萍. 差别矩阵浓缩及其属性约简求解方法[J]. 计算机科学, 2006, 33(9): 181-183.
- [26] HUANG Q, WEI L. Attribute Reduction Method of Sequential Information System Based on Boolean Matrix[J]. Small Micro-computer System, 2016, 37(8): 1717-1720. (in Chinese)
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): 1717-1720.
- [27] YIN Z W, ZHANG J P. An attribute reduction algorithm Based on a concentration Boolean matrix[J]. Journal of Harbin Engineering University, 2009, 30(3): 307-311. (in Chinese)
殷志伟, 张健沛. 基于浓缩布尔矩阵的属性约简算法[J]. 哈尔滨工程大学学报, 2009, 30(3): 307-311.
- [28] ZHU J H, XU Z Y. An efficient attribute reduction algorithm for improved boolean conflict matrix[J]. Computer Engineering and Applications, 2017, 53(6): 145-149. (in Chinese)
朱金虎, 徐章艳. 改进的布尔冲突矩阵的高效属性约简算法[J]. 计算机工程与应用, 2017, 53(6): 145-149.