文章编号: 1001-0920(2007)04-0453-04

基于差别矩阵的属性核快速更新算法

杨 明, 杨 萍
(南京师范大学 数学与计算机科学学院, 南京 210097)

摘 要: 核求解是粗糙集理论的重要内容之一,尽管在核求解问题上已有大量的研究成果,但有关核更新算法的报道却不多. 有人提出一种在对象增加情况下核的增量式更新算法,但未讨论对象动态删除的情况. 对此,提出一种基于差别矩阵的属性核快速更新算法 ——FUAC. 该算法在更新差别矩阵时仅需删除某一行及某一列,或插入某一行,因而可有效提高核的更新效率. 理论分析表明,该算法是有效可行的.

关键词: 粗糙集; 差别矩阵; 核; 更新; 属性约简
中图分类号: TP311 **文献标识码**: A

Fast updating algorithm of computation of a core based on discernibility matrix

YANG Ming, YANG Ping
(School of Mathematics and Computer Science, Nanjing Normal University, Nanjing 210097, China. Correspondent: YANG Ming, E-mail: m.yang@njnu.edu.cn)

Abstract: Computing a core is one of important parts researched in rough set theory. Many algorithms have been proposed for the computation of a core. However, little work has been done in updating of a core. So an incremental updating algorithm of the computation of a core based on discernibility matrix is introduced, which only focuses on the case of inserting and don't concern with that of deleting. Therefore, a fast updating algorithm for computing an attributes core based on discernibility matrix is proposed, which only deletes an old row and the corresponding column, or inserts a new row when updating the decernibility matrix. The updating efficiency of a core is improved. Theoretical analysis shows the feasiblity and the effectiveness of the proposed algolithm.

Key words: Rough set; Discernibility matrix; Core; Updating; Attributes reduction

1 引 言

Pawlak[1] 于 20 世纪 80 年代初提出的 Rough Set(RS,粗糙集)是一种新的处理不精确、不完全与不相容知识的数学理论,近年来该理论在机器学习、数据挖掘及模式识别等多个领域得到了广泛的应用[2,3]. 核和属性约简是粗糙集理论中的两个重要研究内容[4-8].

在现有的求解核方法中, HU[4] 提出的基于差别矩阵的求解核方法是经典的核求解方法之一. 该方法可有效提高求解核的效率,但在某些情况下不能得到正确的核. 于是,叶东毅教授[8] 提出了改进的差别矩阵,但计算量大. 为此,王国胤教授[9,10] 对属性核求解问题进行了深入的探讨,分析了代数观和信息观求解属性核的一致性和差异性,有效补充和完善了文献[8],但算法的效率仍需改进. 在文献[8]的基础上,文献[11]对差别矩阵作了进一步的改进. 然而,信息系统中的对象是动态变化的,已得到的核将可能不再有效,这就需要对核进行动态修改. 文献[12]提出一种基于改进差别矩阵的核增量式更新算法,主要考虑对象动态增加情况下核的更新,但未讨论对象动态删除的情况. 因此,针对该情况,本文提出一种基于差别矩阵的属性核快速更新算法 ——FUAC. 该算法在更新差别矩阵时仅需删除某一行及某一列,或插入某一行,因而可提高核的更新效率. 理论分析表明,本文算法是有效可行的.

2 粗糙集概念

收稿日期: 2005-12-21; **修回日期**: 2006-02-28.
基金项目: 国家自然科学基金项目(70371015); 江苏省自然科学基金项目(BK2005135); 江苏省高校自然科学基金项目(05KJB520066).
作者简介: 杨明(1964—),男,安徽宁国人,教授,博士,从事数据挖掘、机器学习等研究; 杨萍(1967—),女,安徽宁国人,副教授,从事管理决策、粗糙集理论与应用等研究.

为节省篇幅,仅介绍和属性约简及核有关的一些概念,粗糙集的其他一些概念可见文献[2,3].信息系统 IS 是一个 4 元组$\langle U, Q, V, f\rangle$,其中:$U$ 是一组对象的非空有限集合,含 n 个对象,可表示为 $U = \{x_1, x_2, \cdots, x_n\}$;$Q$ 是属性集合;$V = \bigcup_{a\in Q} V_a$,V_a 为属性 a 的值域集;f 是 $U \times Q \to V$ 的映射.属性集合 Q 通常分为条件属性集 C 与决策属性集 D.为便于叙述,设条件属性集合 C 中有 m 个属性 $C_1, C_2, \cdots, C_m$,其值域为有限离散集合,并用 $|\cdot|$ 表示集合的基,同时假设仅有一个决策属性 D,其取值范围是 $1, 2, \cdots, k$.由 D 导出的等价类构成 U 的一个划分 $\{\Psi_1, \Psi_2, \cdots, \Psi_k\}$,其中 $\Psi_i = \{x \in U: f(x, D) = i\}$,$i = 1, 2, \cdots, k$.

若两个不同的对象 x 和 y 具有相同的条件属性和不同的分类,则称 x 和 y 为不一致的;否则称 x 和 y 为一致的.

定义 1 设 $X \subseteq U$ 为论域的一个子集,$P \subseteq C$,X 关于 P 的下近似为 $\underline{P}X = \{x \in U: [x]_P \subseteq X\}$,其中 $[x]_P = \{y \mid f(x, a) = f(y, a), \forall a \in P\}$.

定义 2 设 $P \subseteq C$,对划分 $\{\Psi_1, \Psi_2, \cdots, \Psi_k\}$ 的 P-近似精度为 $\gamma_P = \sum_{i=1}^{k} |\underline{P}\Psi_i| / |U|$.

定义 3 设 $P \subseteq C$,若 $\gamma_P = \gamma_C$,且不存在 $R \subset P$ 使得 $\gamma_R = \gamma_C$,则称 P 为 C 的一个(相对于决策属性 D 的)属性约简.所有 C 的属性约简的交称为 C 的核(简称核),记为 Core(C).

3 已有的差别矩阵定义及其求核方法

为有效求核,HU 等学者提出一种简洁的利用改进差别矩阵来确定核的方法,但得出的结论在某些情况下是错误的,如该方法不能得到例 1 的核,却可得到例 2 的核,详见文献[8].

例 1 表 1 为二值数据表,其中共有 5 个元素和 4 个属性,$C = \{C_1, C_2, C_3\}$ 为条件属性集,D 为决策属性.

例 2 在例 1 中删除第 1 个对象(即第 1 条记录)即得到表 2.

表 1 数据表(1)

元素	属性			
	C_1	C_2	C_3	D
x_1	1	0	1	1
x_2	1	0	1	0
x_3	0	0	1	1
x_4	0	0	1	0
x_5	1	1	1	1

针对 HU 方法的缺陷,文献[8]提出了新的差别矩阵定义并给出求核方法,但计算量大.为改进文献[8]的不足,文献[11]提出了改进的差别矩阵定义以及求解核方法,详细内容可参见文献[11].

表 2 数据表(2)

元素	属性			
	C_1	C_2	C_3	D
x_1	1	0	1	0
x_2	0	0	1	1
x_3	0	0	1	0
x_4	1	1	1	1

然而上述方法均为静态求核方法,于是,文献[12]提出了基于差别矩阵的核增量式更新算法,但该方法未考虑对象删除情况.为此,本文以文献[12]为基础,引入下面的定义 4 和定理 1,定义 4 增加了对不一致对象的计数,其目的是在某个不一致对象删除后可快速判断其相应的不一致对象是否变为一致.

定义 4 对给定的信息系统 IS,定义差别矩阵 $M_1 = \{m_{ij}\}$ 为

$$m_{ij} = \begin{cases} \{a \in C: f(x_i, a) \neq f(x_j, a)\}, \\ \quad f(x_i, D) \neq f(x_j, D), x_i \in U_1, x_j \in U_1; \\ \{a \in C: f(x_i, a) \neq f(x_j, a)\}, \\ \quad x_i \in U_1, x_j \in U_2; \\ \varnothing, \text{其他}. \end{cases}$$

其中:$U_1 = \bigcup_{i=1}^{k} \underline{C}\Psi_i$,$U_2 = U - U_1$,$U_2' = \text{delrep}(U_2)$.函数 delrep($U_2$) 描述如下:

```
Begin
U2′= ∅;
for 任意 x ∈U2 do
  if 不存在 y ∈U2 使得 ∀a ∈C, f(x,a) =
     f(y,a) 且 f(x,D) ≠f(y,D) then {
     U2′= U2′∪{x};
     x.count = 1; // x.count 表示与对象 x
                  // 不一致的对象个数
  }
  else{
       查找 U2 中与 x 不一致的对象 y;
       y.count = y.count + 1;
       }
return U2′;
end.
```

性质 1 对 $U_1 = \bigcup_{i=1}^{k} \underline{C}\Psi_i$,$U_2 = U - U_1$,若 $U_2 \neq \varnothing$,则 $|U_2'| < |U_2|$.

定理 1 对于信息系统 IS,若记 IDM(C, M_1)

$= \{m_{ij} \mid m_{ij} \in M_1$ 且 m_{ij} 为单个属性$\}$,则有 $IDM(C, M_1) = Core(C)$,即当且仅当某个 m_{ij} 为单个属性时,该属性属于核 $Core(C)$.

证明　类似于文献[12]的定理 3. □

对例 2,$U_1 = \{x_1, x_4\}, U_2 = \{x_2, x_3\}, U_2' = \{x_2\}$. 依据定义 4 建立的差别矩阵为

$$\begin{array}{c} \\ x_1 \\ x_4 \end{array}\begin{array}{c} \begin{array}{ccc} x_1 & x_4 & x_2 \end{array} \\ \begin{bmatrix} \varnothing & \{C_2\} & \{C_1\} \\ \{C_2\} & \varnothing & \{C_1, C_2\} \end{bmatrix} \end{array},$$

其中 $x_2.count = 2$.

4　核的快速更新算法

对 $U_1 = \bigcup_{i=1}^{k} \mathcal{L}\Psi_i, U_2 = U - U_1, U_2' = delrep(U_2)$,由定义 4 得到差别矩阵为 M_1,若删除对象为 x,则只要得到 $(U - \{x\})$ 的差别矩阵(简记为 $M_1(x)$),便可由定理 1 求得核. 因此,在对象删除情况下核的快速更新本质上就是差别矩阵的更新问题,M_1 的更新分以下两种情况进行:

1) 若 $x \in U_1$,则删除 M_1 中对象 x 对应的行和列,$U_1 = U_1 - \{x\}$.

2) 若 $x \in U_2$,且 y 为 U_2'中与 x 不一致的对象,则分以下两种情况讨论:

①当 $x \neq y$ 时,如果 $y.count > 2$,则 $y.count = y.count - 1, U_2 = U_2 - \{x\}$,且 M_1 保持不变;否则 $U_2' = U_2' - \{y\}, U_1 = U_1 \cup \{y\}$,且在 M_1 中增加 y 对应的行;

②当 $x = y$ 时,在 $(U_2 - U_2')$ 中查找任意一个与 x 不一致的对象 $z(z \neq x)$;如果 $y.count > 2$,则 $y.count = y.count - 1, z.count = y.count, U_2 = U_2 - \{x\}$,用 z 替代 y 在 M_1 和 U_2'中的位置;否则 $U_2' = U_2' - \{y\}, U_1 = U_1 \cup \{z\}, U_2 = U_2 - \{z\}$,且在 M_1 中增加 z 对应的行并替换相应列中 y 的位置.

依据上述分析,核的快速更新算法(FUAC)描述如下:

输入:1) $U_1 = \bigcup_{i=1}^{k} \mathcal{L}\Psi_i, U_2 = U - U_1, U_2' = delrep(U_2)$,差别矩阵为 M_1;

2) 删除对象为 x;

输出:$M_1(x)$ 及相应的核 $Core(C)$.

```
Begin
1) if x ∈U1 then
   {
     删除 M1 中对象 x 对应的行和列;U1 = U1 - {x};
   }
   else
     if x ∈U2 then
     {在 U2′中找到与 x 不一致的对象 y;
     if x ≠y then {
     if y.count > 2 then {
     y.count = y.count - 1;U2′= U2′- {x};
     }
   else{
     y.count = y.count - 1;U2′= U2′- {y};
     U1 = U1 ∪{y};U2 = U2 - {y,x};
      在 M1 中增加 y 对应的行;
     }
   else // x = y
   {
      在U2中查找与 x不一致的某个对象z且 z ≠x;
     if y.count > 2 then {
     y.count = y.count - 1;z.count = y.count;
     U2 = U2 - {x};
      用 z 替代 y 在 M1 和 U2′中的位置;
     }
   else{
     y.count = y.count - 1;
     U2′= U2′- {y},U1 = U1 ∪{z};U2 = U2 - {z,
   y};
     在 M1 中增加 z 对应的行并替换相应列中 y 的
   位置;
        }
     }
   }
   2) 由定理 1 得到核 Core(C);
   end.
```

定理 2　设 $U_1 = \bigcup_{i=1}^{k} \mathcal{L}\Psi_i, U_2 = U - U_1, U_2' = delrep(U_2)$,由定义 4 得到的差别矩阵为 M_1,若删除对象为 x,则由 FUAC 算法可得正确的核.

证明　对 $(U_1 - \{x\})$,若设由定义 4 建立的差别矩阵为 U_1',则由上述算法可知,$m_1 \in U_1'$ 当且仅当 $m \in M_1(x)$,而由 U_1'及定理 1 可得正确的核,故由 FUAC 算法可得到正确的核. □

为说明 FUAC 算法,参照例 2,通过删除以下不同对象来说明 3 种不同情况下的差别矩阵修改结果:

1) 若删除 x_3,则由 FUAC 算法知 x_2 为一致对象,$U_1 = \{x_1, x_4, x_2\}, U_2 = U_2' = \varnothing$,故增加 x_2 对应行即可,

$$M_1(x) = \begin{array}{c} \\ x_1 \\ x_4 \\ x_2 \end{array} \begin{array}{c} \begin{array}{ccc} x_1 & x_4 & x_2 \end{array} \\ \begin{bmatrix} \varnothing & \{C_2\} & \{C_1\} \\ \{C_2\} & \varnothing & \varnothing \\ \{C_1\} & \varnothing & \varnothing \end{bmatrix} \end{array};$$

2) 若删除 x_2,则由 FUAC 算法知 x_3 为一致对象,$U_1 = \{x_1, x_4, x_3\}$,$U_2 = U_2' = \varnothing$,故增加 x_3 对应行即可,

$$M_1(x) = \begin{array}{c} \\ x_1 \\ x_4 \\ x_3 \end{array} \begin{array}{c} \begin{array}{ccc} x_1 & x_4 & x_2 \end{array} \\ \begin{bmatrix} \varnothing & \{C_2\} & \varnothing \\ \{C_2\} & \varnothing & \{C_1, C_2\} \\ \varnothing & \{C_1, C_2\} & \varnothing \end{bmatrix} \end{array};$$

3) 若删除 x_4,则由 FUAC 算法可知 $U_1 = \{x_1\}$,$U_2' = \{x_2\}$,故仅需删除 x_4 对应的行和列即可,

$$M_1(x) = \begin{array}{c} \\ x_1 \end{array} \begin{array}{c} \begin{array}{cc} x_1 & x_2 \end{array} \\ \begin{bmatrix} \varnothing & \{C_1\} \end{bmatrix} \end{array};$$

从上述实例说明可知,应用 FUAC 算法仅需变动差别矩阵的少量行和列即可,因而提高了差别矩阵的更新效率.

对 $U_1 = \bigcup_{i=1}^{k} \mathcal{L}\Psi_i$,$U_2 = U - U_1$,$U_2' = \mathrm{delrep}(U_2)$,若动态删除的对象为 x,则 FUAC 算法的空间复杂度为 $O(|U_1| * (|U_1| + |U_2'|))$. 由 $|U_1| * (|U_1| + |U_2'|) \leqslant |U_1| * |U| \leqslant |U| * |U|$ 知,当 U 中不一致对象较多时,可有效降低空间复杂度.

对 FUAC 算法,当动态删除对象 x 时,在 $U_1 \cup U_2'$ 中查找 x 的时间最多为 $(|U_1| + |U_2'|)$,修改差别矩阵的时间最多为 $(|U_1| + |U_2'|)$,扫描差别矩阵求核时间最多为 $(|U_1| + 1) * (|U_1| + |U_2'|)$. 为避免扫描差别矩阵,类似于文献[12],可通过对差别矩阵中单个属性出现次数的更新来优化 FUAC 算法,使其时间复杂度降为 $O(|U_1| + |U_2'|)$.

可见,对 FUAC 算法的优化可进一步提高动态求解核的效率. 此外,对于动态修改对象情况下的核更新,可采用先删除再增加的策略来实现. 因此,本文提供的核更新算法是文献[12]的有效补充,与文献[12]相配合可有效解决信息系统增加、删除和修改情况下核的更新问题.

5 结 语

本文提出一种基于差别矩阵的属性核快速更新算法,主要考虑对象动态删除情况下核的更新问题. 该算法通过删除或插入某行或某列的方式对差别矩阵进行更新,保证了核的高效更新,为对象动态删除情况下核的动态更新提供了一条新途径.

参考文献(References)

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

[2] Pawlak Z. Rough set approach to mult-attribute decision analysis[J]. European J of Operational Research, 1994, 72(2): 443-459.

[3] 刘清. Rough 集及 Rough 推理[M]. 北京: 科学出版社, 2001.
(Liu Q. Rough sets and rough reasoning[M]. Beijing: Science Press, 2001.)

[4] Hu X H, Cercone N. Learning in relational databases: A rough set approach[J]. Computational Intelligence: An Int J, 1995, 11(2): 323-338.

[5] Jelonek J, Krawiec K, Slowinski R. Rough set reduction of attributes and their domains for neural networks[J]. Computational Intelligence, 1995, 11(2): 339-347.

[6] Wang J, Wang J. Reduction algorithm based on discernibility matrix the ordered attributes method[J]. J of Computer Science and Technology, 2001, 16(6): 489-504.

[7] Guan J W, Bell D A. Rough computational methods for information systems[J]. Artificial Intelligences, 1998, 105(1-2):77-103.

[8] 叶东毅,陈昭炯. 一个新的差别矩阵及其求核方法[J]. 电子学报, 2002, 30(7): 1086-1088.
(Ye D Y, Chen Z J. A new discernibility matrix and the computation of a core[J]. Acta Electronica Sinica, 2002, 30(7): 1086-1088..)

[9] Wang G Y, Zhao J, An J J, et al. Theoretical study on attribute reduction of rough set theory: Comparison of algebra and information views[C]. Proc of the 3rd IEEE Int Conf on Cognitive Informatics. Washington D C, 2004: 148-155.

[10] Zheng Z, Wang G Y, Wu Y. Objects' combination based simple computation of attribute core[C]. Proc of the 2002 IEEE Int Symposium on Intelligent Control. Vancouver, 2002: 514-519.

[11] 杨明, 孙志挥. 改进的差别矩阵及其求核方法[J]. 复旦大学学报, 2004, 43(5): 865-868.
(Yang M, Sun Z H. Improvement of discernibility matrix and the computation of a core[J]. J of Fudan University, 2004, 43(5): 865-868.)

[12] 杨明. 一种基于改进差别矩阵的核增量式更新算法[J]. 计算机学报, 2006, 29(3): 407-413.
(Yang M. An incremental updating algorithm of the computation of a core based on the improved discernibility matrix[J]. Chinese J of Computers, 2006, 29(3): 407-413.)