

基于窗口的二次关联规则挖掘

阴仁杰

(中国海洋大学, 山东 青岛 266100)

摘要:传统的关联规则和时态关联规则挖掘只考虑到事务数据库中的各项之间的关系,没有考虑到挖掘出来的关联规则也可能存在一定的关系。为此,本文在基于窗口的时态关联规则之上,提出了二次关联规则挖掘。把一个窗口中的时态关联规则当成一条发生的事务,挖掘出时态关联规则之间存在的隐藏关系。最后,给出了一个实例分析,发现了其中的时态关联规则之间隐藏的关系。

关键词:窗口;二次关联规则;数据挖掘

中图分类号:TP274+.2 **文献标志码:**A **文章编号:**1671-1807(2011)07-0110-04

关联规则是数据挖掘^[1]的一个重要研究领域,由 Agrawal 等人提出。人们很快发现随着时间的变化,关联规则过一段时间也许不适用了。此后, Cheung 等人提出了增量式关联规则挖掘,使得关联规则随着数据的更新规则不断的变化^[2]。但是,这样做也是把关联规则的时间跨越到整个数据库空间,也只是相对的动态,仍然没有考虑关联规则的时间适用性。

由于时态关联规则更能体现规则的有效性,近年来,大量的研究者们对时态关联规则的理论及实证进行了研究,也产生了大量的时态关联规则论文。孟志青^[3]提出了时态关联规则的有关概念和相关性质,给出了时态挖掘模型,但是没有提出相关的算法进行挖掘。崔晓军,薛永生^[4]提出了基于日历模式的时序关联规则,可以发现任意日历模式的关联规则,但是分析和理解起来比较困难,算法也比较复杂,有些细小的日历模式一般情况下并没有太大意义。董祥军等人^[5]在之前的研究上,提出了一种容易理解的时态关联规则,算法也比较简洁。但是,它只对某些特定的时段进行挖掘,有些时候必须要知道商品的生命周期,这样实践起来会比较麻烦,也没有很好的从整个历史时间上考察时态关联规则随时间的变化性质。荣冈等人^[6]提出了动态关联规则挖掘,可以考察关联规则随时间变化而变化的性质,但是它首先要在挖掘整个数据库时间段上有效的关联规则,也就是说,它是把静态的关联规则进行切分,使它变成了“动态”,

它仍然不会发现上面所说的生命周期很短的商品的关联规则。

但是,不管是传统关联规则还是时态关联规则,都是发掘事务数据库中的项之间的关系,没有考虑到发掘出来的关联规则之间隐藏的关系。本文在首先介绍基于窗口的时态关联规则挖掘,然后在挖掘出来的时态关联规则基础之上进行二次挖掘,来发掘时态关联规则之间隐藏的关系。

1 相关定义

定义 1 一个时态关联规则可以看作是一个二元式 $\langle AR, TF \rangle$, 式中 AR 是一条形如 $X \Rightarrow Y$ 的关联规则, TF 是一个时间特征, 如商品的生命周期、基于日历的时间特征、循环的时间特征等。二元式 $\langle AR, TF \rangle$ 表示了在时间特征 TF 中关联规则 AR 都成立。

定义 2 设 $I = \{i_1, i_2, \dots, i_m\}$ 是项的集合, 任务相关的数据 D 为数据库事务的集合, 事务数记作 $|D|$, 每个事务 T 是项的集合, 使得 $T \in I$ 并且都有一个时间戳 t ; 所有 $t \in TF$ 的事务集合记作 d , d 中的事务个数记作 $|d|$ 。

1) 设 $X \subset I$, 项集 X 的时态支持度是事务集 d 中包含 X 的事务数与 d 中事务数之比, 记

为 $\text{supp}(X, TF)$, 即 $\text{supp}(X, TF) = |\{T: X \subseteq T, T \in d\}| / |d|$ 。

2) 设 $X \subset I, Y \subset I$ 并且 $X \cap Y = \Phi$, 规则 $X \Rightarrow Y$ 在

收稿日期:2011-04-07

作者简介:阴仁杰(1984—),男,山西平遥人,中国海洋大学管理学院硕士研究生,研究方向:智能决策支持系统与工业集成工程。

事务数据库 D 中的时态支持度是事务集 d 中包含 X 和 Y 的事务数与 d 中事务数之比,记为 $\text{supp}(X \Rightarrow Y, \text{TF})$ 即 $\text{supp}(X \Rightarrow Y, \text{TF}) = |\{T: X \cup Y \subseteq T, T \in d\}| / |d|$ 。

3) 规则 $X \Rightarrow Y$ 在事务集中的时态置信度是指事务集 d 中包含 X 和 Y 的事务数与包含 X 的事务数之比,记为 $\text{conf}(X \Rightarrow Y, \text{TF})$,即

$$\text{conf}(X \Rightarrow Y, \text{TF}) = |\{T: X \cup Y \subseteq T, T \in d\}| / |\{T: X \subseteq T, T \in d\}|$$

选择时间特征 TF 的目的是为了减少支持度计算式中分母的值用部分事务数 $|d|$,而不是全部事务数 $|D|$ 从而更有效地挖掘关联规则。这样就可以避免了传统静态关联规则把时间段跨越到整个数据库上的缺陷,从而更符合关联规则的实际意义。

定义 3 设 DS 是数据流序列 $\langle s_1, s_2, \dots, s_k, \dots \rangle$, $s_k = \{i_1, i_2, \dots, i_m, \dots\}$, $s_k (k = 1, 2, \dots)$ 为事务 $i_m (m = 1, 2, \dots, p, \dots)$ 的属性。将数据流 DS 按时间分段,每一个分段对应一个数据流子序列且每个子序列对应一个相等的时间间隔,这样的一个数据分段称为一个基本窗口,记做 BW 。

定义 4 窗口:一个窗口 W 对应一个连续的基本窗口序列 $\langle BW_1, BW_2, \dots, BW_k, \dots \rangle$,它所容纳的基本窗口的数目是一个不定的值 k ,如图 1 所示。

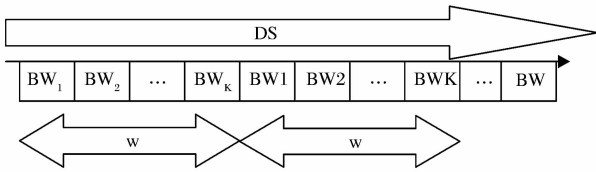


图 1 窗口图

定义 5 从数据库中挖掘的基于窗口的时态关联规则所组成的集合记为 L , L 中的元素是一条形如 $\text{conf}(X \Rightarrow Y, W) = |\{T: X \cup Y \subseteq T, T \in d\}| / |\{T: X \subseteq T, T \in d\}|$ 的关联规则, $X \Rightarrow Y$ 为一条关联规则, W 为窗口。

定义 6 二次关联规则挖掘的数据集记为 $D, D = \{t_1, t_2, \dots, t_k, \dots, t_n\}$, $t_n = \{i_1, i_2, \dots, i_m, \dots, i_p\}$, $t_k (k = 1, 2, \dots, n)$ 称为窗口, $i_m (m = 1, 2, \dots, p)$ 称为关联规则项目。

定义 7 二次关联规则是描述数据库中关联规则之间存在的潜在关系的规则,形式为 $X' \Rightarrow Y'$, 其中 $X' \subseteq L, Y' \subseteq L$, 且 $X' \cap Y' = \emptyset$, X' 称为规则头 (antecedent), Y' 称为规则尾 (consequent)。项集之间的关联表示:如果 X' 出现在一个窗口中,那么 Y' 在这

个窗口中同时出现的可能性比较高。

2 基于窗口的挖掘时态关联规则算法

2.1 挖掘基于窗口的时态关联规则

挖掘基于窗口的时态关联规则首先要将整个数据库中的事务分成一个个基本窗口,然后由基本窗口组成需要的窗口。在给定的窗口中产生时态支持度和时态置信度分别大于用户给定的最小支持度 (min_supp) 和最小置信度 (min_conf) 的关联规则。挖掘过程可以分两步 (1) 对每个窗口,求频繁项集 X 使 $\text{supp}(X, BW) \geq \text{min_supp}$, 用频繁项集 X 发现规则对于每一个 $A \cup B = X \cap B = \emptyset$ 验证 $\text{conf}(A \Rightarrow B, \text{TF}) \geq \text{min_conf}$ 。(2) 合并所有窗口的大项集和关联规则。

算法也是根据上面的过程来进行,最小支持度是根据每个窗口来定的,所以对每个窗口来说最小支持度是不一样的。那么首先就要计算每个窗口中的事务数,然后观察每个窗口中的项集 X 的是否满足最小支持度的要求。关联规则也是根据每个窗口中的频繁项集算得,最后合并每个窗口中的频繁项集和关联规则。

算法中规定 W 为窗口, $|W|$ 为窗口中的事务数, L^i 为每个窗口中的频繁项集, A^i 为每个窗口中的关联规则,具体算法如下:

输入: D : 划分好窗口的数据库, min_sup : 最小支持度, min_conf : 最小置信度

输出: 以窗口为单位的数据库中的所有频繁项集和所有关联规则

方法:

foreach $W \in D$ do

$\text{min_sup} = \text{min_sup} * |W|$

$L_1 = \{1 - \text{frequent itemsets}\}$

for ($k = 2; L_{k-1} \neq \emptyset; k++$) {

$C_k = \text{Apriori_gen}(L_{k-1})$

foreach transaction $t \in W$ do

$C_t = \text{subsets}(C_k, t)$

foreach $c \in C_t$ do $c.\text{count}++$ end for

end for

$L_k = \{c \in C_k \mid c.\text{count} \geq \text{min_sup}\}$

end for

$L = \bigcup_k L_k$

$A = \text{Association_rule_gen}(L)$

$L^i = L$

$A^i = A$

end for

$\text{Largesets} = \bigcup_i L^i$

$Association_rules = \bigcup_i A^i$

该算法是在传统 Apriori 算法上的改进,所以比较容易理解的。Apriori_gen()函数就是通常的候选项集产生函数,Association_rule_gen()函数是根据频繁项集产生关联规则,最后面是得到整个数据库的频繁项集和关联规则。

2.2 二次规则挖掘

把挖掘到的初始时态关联规则按照窗口号进行分组,相同窗口挖掘出来的时态关联规则放在一起,表示这些时态关联规则是同时发生的,就好比事务数据库中的一条事务。然后就可以利用传统的关联规则挖掘算法对基于窗口的时态关联规则进行二次挖掘。算法如下:

```
L'为大项集,C'为候选项集,minsup'表示最小支持度,minconf'表示最小置信度,w 表示某个窗口
L1' = {large 1 - itemsets}
for(k = 2;Lk-1' ≠ φ;k++){
Ck' = Apriori_gen(Lk-1') //构造候选项集
foreach transaction t ∈ d do
Ci' = subsut(Ck',t) //搜索事务 t 中包含的候选项集
foreach c' ∈ Ci' do c'.count ++ end for //计算支持数
end for
Lk = {c' ∈ Ck' | c'.count ≥ min_sup} //得到 k 阶大项集
end for
L' = ∪k Lk'
```

得到大项集后,就可以根据传统的求关联规则的算法求出二次关联规则。

3 实验分析

运用前面的算法来进行分析,算法采用 PHP 语言进行实现的,实验平台:CPU 是 Pentium(R) Dual-Core E5300 2.60GHz,内存 2G,操作系统是 Windows XP.

采用的是 SQL SERVER2005 自带的 AdventureWorks 数据库,对其中的销售数据进行关联规则挖掘。经过预处理,订单号相同的放在一个购物篮里,取其中的 2003 年 8 月 1 号到 2004 年 7 月 31 号一整年的数据进行分析,总共有 23 434 条记录。

我们设置一天为一个基本窗口,30 天作为一个窗口,最小支持度为 0.04,最小置信度为 50%,可以得到每个窗口的时态关联规则,如表 1 所示:

表 1 基于窗口的时态关联规则

ID	confidence
1	871—>870, 872—>870, 880—>870, 930—>921
2	871—>870, 872—>870, 930—>921
3	871—>870, 872—>870
4	871—>870, 872—>870, 930—>921
5	871—>870, 872—>870, 930—>921
6	871—>870, 872—>870, 930—>921
7	871—>870, 872—>870
8	871—>870, 872—>870
9	871—>870, 872—>870, 923—>934, 934—>923
10	871—>870, 872—>870, 930—>921

得到初次的时态关联规则数据库以后,我们就可以对该数据库进行二次关联规则挖掘,在这里,设定支持度为 3,置信度为 60%,于是得到大项集,如表 2 所示:

表 2 基于时态关联规则的大项集

大项集	支持度
871—>870	10
872—>870	10
930—>921	6
871—>870, 872—>870	10
871—>870, 930—>921	6
872—>870, 930—>921	6
871—>870, 872—>870, 930—>921	6

根据大项集得到二次关联规则如表 3 所示:

表 3 二次关联规则

关联规则	置信度
(871—>870)→(872—>870)	100%
(872—>870)→(871—>870)	100%
(871—>870)→(930—>921)	60%
(930—>921)→(871—>870)	100%
(872—>870)→(930—>921)	60%
(930—>921)→(872—>870)	100%
(871—>870, 872—>870)→(930—>921)	60%
(930—>921)→(871—>870, 872—>870)	100%
(871—>870, 930—>921)→(872—>870)	100%
(872—>870, 930—>921)→(871—>870)	100%

从表 3 中可以看出,哪些基于窗口的时态关联规则同时发生的概率会比较高。

4 小结

本文首先介绍了时态关联规则,然后提出时态关联规则的概念,传统的时态关联规则只是针对事务数据库中的项来进行挖掘,没有考虑到时态关联规则之间隐藏着的联系。为此,本文提出了二次关联规则挖

掘,来发掘时态关联规则之间隐藏的关系。最后给出了实例分析,并发现了其中时态关联规则之间隐藏着的关系。

参考文献

[1]RAKESH AGRAWAL, TOMASZ IMIELINSKI, ARUN SWAMI. Mining association rules Between Sets of Items in Large Database. ACM SIGMOD [M]. Washington, D. C. : 1993;207—216.
[2]DAVID WAI—LOK CHEUNG, JIAWEI HAN, VINCENT NG, C Y WONG. Maintenance of Discovered Association

Rules in Large Databases; An Incremental Updating Technique[M]. Proceeding of ICDE,1996;106—114.
[3]孟志青. 时态关联规则挖掘的若干性质[J]. 计算机工程与应用,2001,37(10):86—87.
[4]崔晓军,薛永生. 基于日历的时序关联规则挖掘算法[J]. 计算机应用,2006,26(8):1898—1903.
[5]董祥军,宋瀚涛,姜合,陆玉昌. 时态关联规则的研究[J]. 计算机工程,2005,31(15):24—26.
[6]荣冈,刘进锋,顾海杰. 数据库中动态关联规则的挖掘[J]. 控制理论与应用,2007,24(1):129—133

The Second Association Rules Mining Based on Windows

YIN Ren-jie

(School of Management,Ocean University of China, Shandong,Qingdao Shandong 266100,China)

Abstract: Traditional association rules and temporal association rule mining only consider the relationship between the items in the transaction database, don't take into account that the association rules mined out may exist a certain relationship. Therefore, this paper propose the second association rule mining on top of the window-based temporal association rules. Put the temporal association rules in a window as a transaction occurs, dig hidden relationships between the temporal association rules. Finally, we provide a case study, and find the relationship hidden in the temporal association rules.
Key words: windows;second association rules;date mining

(上接第 74 页)

[6]AIT-SAHALIA Y, DUARTE J. Nonparametric option pricing under shape restrictions[J]. Journal of Econometrics, 2003,116(1—2):9—47.
[7]CHEN G, CHOI Y K, ZHOU Y. Nonparametric estimation of structural change points in volatility models for time series [J]. Journal of Econometrics,2005,126(1):79—114.
[8]俞雪飞,王金玉,潘德惠. 非参数区间估计在证券投资分析中的应用[J]. 东北大学学报,2003,24(9):877—880.
[9]杨二鹏,张德生,李文静. 基于非参数模型的沪深 300 股指波动性研究[J]. 纺织高校基础科学学报,2010,23(1):42—45.
[10]叶阿忠. 我国通货膨胀的非参数回归模型[J]. 数理统计与管

理,2002,21(1):47—51.
[11]吕寒玉,肖庆宪. 非参数方法在我国证券市场收益波动研究中的应用[J]. 数学的实践与认识,2005,35(4):89—94.
[12]王琳. 非参数局部线性估计方法及对中国股市杠杆效应的实证分析[J]. 大众科技,2008,105(5):185—187.
[13]许冰,任军峰. 基于非参数 GARCH 模型的一种波动率估计方法[J]. 统计与信息论坛,2006,223(10):63—72.
[14]鲁万波. 基于非参数 GARCH 模型的中国股市波动性预测 [J]. 数理统计与管理,2006,25(4):455—464.
[15]叶阿忠,李子奈. 非参数计量经济联立模型的局部线性工具变量估计[J]. 清华大学学报,2002,42(6):714—717.

Review of the Research on Stock Market Volatility: Nonparametric Model

CHEN Xiao-xuan

(School of Economics and Management,Beihang University,Beijing 100191,China)

Abstract: With the development of the financial industry and the improvement of financial markets, the security market becomes more and more important in people's life. As the basis for existence and development of the stock market, volatility has increasingly become the focus of financial economics. However, the perspective, methods and conclusions are quite different in these studies. This paper summarizes the literatures about the volatility of stock market in the recent twenty years both home and abroad, and furthermore brings up the underlying problems in the past studies.
Key words: stock market volatility;ARCH models;nonparametric model