

# 用于关联规则挖掘的一种基于小生境技术的 GEP 算法

陈云亮<sup>1,2</sup> 李欣<sup>1</sup> 杨捷<sup>3</sup> 谢长生<sup>1</sup>

(华中科技大学计算机学院 武汉 430074)<sup>1</sup> (中国地质大学计算机学院 武汉 430074)<sup>2</sup>

(华中科技大学图像识别与人工智能研究所 武汉 430074)<sup>3</sup>

**摘要** 为了提高关联规则挖掘算法处理大数据集的能力,在基因表达式编程进化算法(Gene Expression Programming)的基础上,提出了一个新的挖掘强关联规则的算法框架。主要贡献在于提出并实现了基于小生境技术的基因表达式编程进化算法 NGE,用于挖掘关联规则。NGE 算法首先进行小生境演化,融合小生境并剔除同构的优秀个体,然后对小生境解进行笛卡儿交叉,以产生更好的结果。实验结果表明,与同类优秀的算法对比,NGE 算法的种群多样性与精确度都有很好的结果,并且在提取有效规则的效率上也有较大的提高。

**关键词** 数据挖掘,关联规则,基因表达式编程,小生境

## Novel Algorithm GEP Based on Niche Applied to Association Rules Mining

CHEN Yun-liang<sup>1,2</sup> LI Xin<sup>1</sup> YANG Jie<sup>3</sup> XIE Chang-sheng<sup>1</sup>

(School of Computer Science, Huazhong University of Science and Technology, Wuhan 430074, China)<sup>1</sup>

(School of Computer Science, China University of Geosciences, Wuhan 430074, China)<sup>2</sup>

(Institute for Pattern Recognition and Artificial Intelligence, Huazhong University of Science and Technology, Wuhan 430074, China)<sup>3</sup>

**Abstract** In order to advance the performance of association rules mining algorithm when disposing big dataset, a novel GEP based on Niche (NGE) was presented to solve the problem. The procedure of NGE began with the niche evolution and then fused some of the sub-niches according to the similarity of best individuals. Then Cartesian product was nested in the kernel set of niches to generate better outcomes. The experimental results show that our algorithm performs better than the other similar evolutionary algorithm in terms of diversity of population and precision; besides, it can discover more association rules.

**Keywords** Data mining, Association rule, Gene expression programming, Niche

## 1 前言

关联规则挖掘是数据挖掘的一个重要方面,其目的是从数据中挖掘出满足一定条件的依赖关系。它通过挖掘数据库中的数据项集之间的某种潜在关系,从而在大量数据中发现一些潜在和有趣的关联关系,以此为依据来帮助决策者做出合理、适当的决定。近年来,与此相关的研究工作有了长足的进步,大致可分为以下两类:一类是在经典 Apriori 算法的基础上做出改进<sup>[1,2]</sup>,但由于需要多次搜索数据库,形成频繁集(frequent itemsets)而导致计算量巨大,因此面对海量数据库时算法效率锐减;另一类结合人工智能的方法(如仿生学或神经网络等)<sup>[3-7]</sup>直接从实验样本中抽取关联规则,由于人工智能的方法能有效处理大数据集的问题,因此此类方法得到了广泛的认可。比较典型的算法有基于遗传算法、神经网络、蚂蚁算法的关联规则挖掘等。但是应用遗传算法进行挖掘时可能存在收敛速度慢或漏报规则较多等问题。还有一些学者<sup>[5]</sup>

提出了一种基于遗传算法和蚂蚁算法相结合的多维算法,在运行时间上取得了一定的效果,但挖掘的有效规则数量不算非常理想。

基因表达式编程(Gene Expression Programming, GEP)是函数关系挖掘的新方法,它继承和发展了遗传算法(Genetic Algorithm, GA)和遗传编程。GEP 能够挖掘出传统方法不易表达、不易发现的关联知识。本文在文献<sup>[11,12]</sup>的基础上结合小生境技术,提出并实现了基于小生境的基因表达式编程算法(niche-Gene Expression Programming, 简称 NGE)。NGE 充分利用小生境技术便于多目标、多峰值搜索的特点,结合基因表达式编程算法(GEP)操作简单、鲁棒性强的优势对解空间进行搜索,既保证了解集分布的均匀性和多样性,又加快了收敛的速度<sup>[8,9]</sup>。本文的目的在于研究 NGE 是否能有效地进行大规模数据集的关联规则挖掘,并研究比较该算法的性能。

本文第 2 节回顾了关联规则的相关概念;第 3 节给出了

到稿日期:2008-12-24 返修日期:2009-03-09 本文受国家自然科学基金资助项目(No. 60603074 和 No. 60603075),国家 973 重大项目(2004CB318203),中国地质大学(武汉)优秀青年教师资助计划项目(CUGQNL44)资助。

陈云亮(1979-),男,博士研究生,讲师,CCF 会员,主要研究方向为智能计算、智能网络存储, E-mail: cyl\_king2004@163.com; 李欣(1986-),女,硕士研究生,主要研究方向为智能网络存储; 杨捷(1983-),男,博士研究生,主要研究方向为人工智能、图像处理和模式识别; 谢长生(1957-),男,教授,博士生导师,CCF 高级会员,主要研究方向为计算机系统结构。

基于小生境技术的基因表达式编程算法实现;第4节中引入2个规则挖掘的例子,并将结果与同类优秀算法进行了比较;最后是对工作的总结与展望。

## 2 预备知识

### 2.1 关联规则

记含  $m$  个事务的数据库  $D_m = \{d_1, d_2, \dots, d_m\}$ , 每个事务都含有  $n$  个项目属性集合  $A_n = \{a_1, a_2, \dots, a_n\}$ 。关联规则是如下形式的逻辑蕴涵:  $X \Rightarrow Y, X \in A_n, Y \in A_n$ , 且  $X \cap Y = \phi$ 。关联规则具有如下两个重要的属性: 支持度 ( $spt(X \Rightarrow Y, D_m)$ ) 和置信度 ( $cnf(X \Rightarrow Y, D_m)$ ), 同时满足最小支持度 ( $\min\_spt$ ) 和置信度 ( $\min\_cnf$ ) 的规则被称为强规则。

### 2.2 GEP 算法

基因表达式程序设计算法是一种新的基于基因表达规律的进化算法, 它的主要特点是利用定长的线性串(基因型组)表达不定长度和形态的非线性实体(表达式树)。采用 GEP 算法进行关联规则的挖掘, 优势在于:

- (1) 线性编码、遗传操作简单; 解码后对应的规则树具有灵活的表示能力, 降低了误报率和漏报率;
- (2) 收敛速度快, 实验证明至少比 GA 和 GP 快 2 个数量级<sup>[8,9]</sup>;
- (3) 利用表达式树描述关联规则, 在形式上更加直观, 使规则易于理解。

初始群体由众多初始个体组成。初始个体为所要解决问题的各种可能的符号表达式(算法树), 它通过随机方法产生。个体是用功能符号集和终止符号集对问题的自然描述。个体的功能符号集由 3 个逻辑符号 OR, AND 和 NOT 组成; 终止符号集是数据各种属性名及其对应模糊值的集合, 表示方法为属性名=属性值。

评价一条关联规则优劣的标准是以其实际的支持度和置信度来决定的, 多数文献<sup>[4,5,7]</sup>采取如下的计算方法

$$fitness(X \Rightarrow Y) = \partial_s \times (spt(X \Rightarrow Y, D_m) - \min\_spt) + \partial_c \times (cnf(X \Rightarrow Y, D_m) - \min\_cfd)$$

其中,  $\partial_s, \partial_c$  为支持度和置信度的惩罚系数。然而在实际应用中, 发现按照如上函数评价染色体易产生“伪规则”, 即存在

$$spt(X \Rightarrow Y, D_m) - \min\_spt < 0$$

$$cnf(X \Rightarrow Y, D_m) - \min\_cfd \geq spt(X \Rightarrow Y, D_m) - \min\_spt$$

或

$$cnf(X \Rightarrow Y, D_m) - \min\_cfd < 0$$

$$spt(X \Rightarrow Y, D_m) - \min\_spt \geq cnf(X \Rightarrow Y, D_m) - \min\_cfd$$

的情况, 依然有  $fitness(X \Rightarrow Y) \geq 0$ , 而这样的个体显然不符合强关联规则的定义。为此, 对评价函数进行了修改, 首先定义一个“或乘 $\otimes$ ”运算。

**定义 1** 假设两个实数  $x, y \in (-1, 1)$ , 用科学计数法表示为  $x = a \times 10^m, y = b \times 10^n$ 。其中  $a, b \in (-1, 1), m, n \in [0, +\infty)$ , 则有

$$x \otimes y = \begin{cases} a + b > 0, & \text{if } a > 0, b > 0 \\ 0, & \text{else} \end{cases}$$

由此定义新的适应度函数为

$$fitness^*(X \Rightarrow Y) = 100 \times [\partial_s \times (spt(X \Rightarrow Y, D_m) - \min\_spt) \otimes \partial_c \times (cnf(X \Rightarrow Y, D_m) - \min\_cfd)]$$

## 3 基于小生境的 GEP 算法

### 3.1 小生境技术

小生境技术的具体思想就是把整个种群分解成若干小生境, 在初始条件下它们具有相同的群体规模。然而随着进化的深入, 规模的大小随着平均适应度自适应地发生动态调整。当然, 每个小生境通过设置其最大规模 MAX 和最小规模 MIN, 控制子种群上下限。染色体的各种遗传操作仅限于小生境内部发生。这样极大地丰富了种群的多样性, 也使搜索的范围更加宽泛。详细的技术流程可参看文献<sup>[11,12]</sup>。本节中主要叙述小生境技术应用于搜索最优规则集时的演化操作, 包括小生境融合和分裂等子算法。

#### 3.1.1 小生境融合

考虑到每代进行后可能出现某些小生境的优秀个体同构的现象, 为维持种群的多样性, 防止冗余规则的出现, 有必要对该小生境进行融合操作。不失一般性, 以 2 个小生境为例。

##### 算法 1 小生境融合

- ① 两个将要融合的小生境(假设为  $nich_1, nich_2$ , 融合前两者规模分别为  $s_1$  和  $s_2$ ) 个体全部归入  $nich_1$  中;
- ② 对  $nich_1$  进行相似度检查(执行同构互斥操作), 剔除同构染色体, 以进行融合, 得到修改后的  $nich_1$  规模  $s_1'$ ;
- ③ 如果  $s_1'$  大于 MAX, 则用轮盘赌法选出多余的个体, 调整  $s_1'$ , 执行小生境分裂算法;
- ④ 如果  $s_1'$  小于 MIN, 则引入新个体直到满足最小规模, 调整  $s_1'$ ;
- ⑤ 随机构造子小生境  $nich_2$ , 使规模数满足  $s_2' = s_1 + s_2 - s_1'$ 。

#### 3.1.2 小生境演化

记当前父代  $P(t)$  演化得到的子代为  $S$ , 则  $P(t+1) = P(t) \cup S$ , 只要  $\cup P(t+1)$  保持在  $R$  的均匀分布, 则从统计的角度上看  $\cup P(t+1)$  的非劣集合将极大地逼近非劣集合  $OP(R, <)$ , 即将  $\cup(OP(P(t+1), <))$  作为  $OP(R, <)$  的极限逼近。

##### 算法 2 小生境演化

- ① 所有小生境内部执行传统的交叉与变异等遗传操作, 以得到子代  $S$ , 将后代  $S$  并入父代小生境群体, 得到  $P(t+1) = P(t) \cup S$ ;
- ② 在集合  $P(t+1)$  上执行算法 1, 对所有得到的所有小生境解进行笛卡儿交叉, 以此获得非劣集合  $OP(P(t+1), <)$ , 将它作为本代小生境的输出结果;
- ③ 采用精英保留策略, 取子代最优的 5% 个体替换父代等数差的个体。

### 3.2 NGEP 算法流程

下面给出了基于小生境融合的 NGEP 算法流程:

- Step1 针对具体问题, 确定染色体的头部长度、各种算子概率、函数集和终结符集;
- Step2 随机创建初始种群, 设定小生境个数以及 MAX 和 MIN, 将染色体平均分配到这些小生境中;
- Step3 在每个小生境内部独立执行算法 2, 计算个体适应度并将得到的最优规则集  $OP(P(t+1), <)$  放入小生境核集作为本代计算结果(即  $\cup(OP(P(t+1), <))$ );
- Step4 在各小生境之间对最优个体进行相似度判断; 若出现相似个体, 则根据算法 1 进行融合;
- Step5 在各小生境内执行同构互斥操作; 若出现相似个体, 则将适应度小的那个取出, 并补充等数的随机个体。
- Step6 收敛性判定, 如果满足收敛性条件(本文中设定最大进化代数), 则结束进化过程, 输出小生境核集; 否则返回 Step 3。

## 4 实验与评价

引用一个人工模拟数据库(Artificial Simulation Database, ASD)和一个衡量生态环境压力的问题,来检验 NGEP 算法的有效性。实验平台配置如下:CPU:AMD AM2 Athlon 64 X2 5000 (2.6GHz),内存 1GB,OS: Windows XP, Professional, Service Pack2。文献[5]提出了一种基于遗传算法(GA)和蚂蚁算法(AA)相结合的规则挖掘算法 GA-AA,由于综合遗传和蚂蚁算法共有的全局搜索能力,显示出了优于单个遗传算法或蚂蚁算法的求解性能,并且该算法与其他智能挖掘算法在性能上都有比较强的优势,因此本文特将 NGEP 与 GA-AA 进行比较。两者定义相同的适应度函数,NGEP 的参数设置如表 1 所列。

表 1 NGEP 的参数设置

实验参数	详细描述	实验参数	详细描述
函数集	$F = \{\wedge, \vee, \neg\}$ 分别代表与、或、非三种基本操作	项目属性集合 $A_n$	
终结符集		常数据集	随机模拟
种群规模	100	交叉率	0.33
头部长度	8	变异率	0.01
基因数量	5	选择算子	轮盘赌算子
连接函数	+	进化代数	1000
IS 交换率	0.1	小生境个数	5
RIS 交换率	0.1	个体重组概率	0.1
倒置概率	0.1	小生境最大规模	60
小生境最大规模	60	小生境最小规模	10

GA-AA 的参数引自文献[5],所有实验均独立进行 100 次。此外,引入在线性能、离线性能评价算法的有效性,并给出了一个评价规则集多样性的定量分析函数。

定义 2 群体(规则集)多样性性能指标为

$$DIV_{pop} = \frac{\sum_{m=1}^N \sum_{n=1}^N d_{m,n}}{N^2}$$

其中,  $N$  为群体规模,  $d_{m,n}$  表示个体  $m$  与个体  $n$  之间的海明距离。

### 4.1 人工模拟数据库(ASD)

数据库  $D$  由 128 个数据事务组成,而项目属性集合  $A_n$  含有 7 个二值属性  $\{a_1, a_2, \dots, a_7\}$ , 其中  $\{a_7\}$  作为干扰属性存在。数据库按照 3 条既定规则构成:  $(a_1 \wedge a_2) \vee (\neg a_3) \rightarrow a_4$ ,  $(a_1 \vee a_2) \wedge a_3 \rightarrow a_5$  和  $(\neg a_1) \wedge (a_2 \vee a_3) \rightarrow a_6$ 。取  $\min\_spt = 0.1$ ,  $\min\_cnf = 0.8$ , 所得结果如表 2 所列。

表 2 ASD 运行结果

统计平均值	运行	关联	平均
算法	时间/s	规则(条)	准确率
NGEP	6.1	39	74.2
GA-AA	8.9	28	53.8

结果显示,无论是在运行时间有效关联规则的挖掘个数,还是在平均精度上,NGEP 与 GA-AA 的性能都有较大的提高。

图 1 是两种算法在规则种群多样性以及进化曲线的比较。

图 1(a)是求解过程中的多样性曲线对比。可以看到,相比较 GA-AA 而言,NGEP 算法到 600 代时,多样性依然很好;随着求解的进行,总体上多样性降低(这是合理的),但依然保持较高的多样性,因此具有不断获得新解的能力,而不易陷入局部最优。

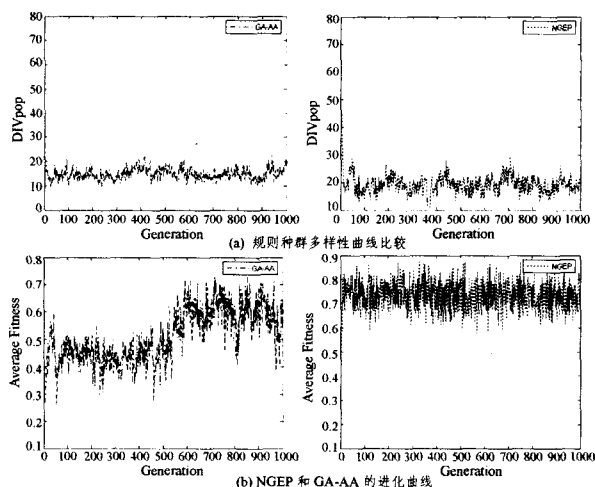


图 1

图 1(b)是 NGEP 与 GA-AA 算法的进化曲线的比较。从图中可以看出 GA-AA 随着演化代数的增加,它的拟合精度有一个逐渐增加的过程,该算法只有在演化到较大代数之后,算法的拟合精度才能达到一个较为理想的均值 53.8。而 NGEP 算法具有一定的稳定性,随着演化代数的增加,平均拟合精度一直在 74.2 附近小幅振荡。

### 4.2 Iris 数据集的处理

本文实验中采用的 Iris 数据集来源于 UCI 机器学习数据集,可以从 <http://www.sgi.com/Technology/mlc/db/> 下载。Iris 数据集拥有 150 个包含 4 个属性值的记录,这 4 个属性值分别是“萼片长度”、“萼片宽度”、“花瓣长度”、“花瓣宽度”。本文选择其中的 100 个记录作为挖掘关联规则的实验数据集,剩下的 50 个记录作为检测挖掘结果的检测数据集。首先将数据集进行标准化预处理,然后使用 k-means 聚类算法将元组的每一个属性划分为两个值变量(例如用“长”、“短”形容“萼片长度”和“花瓣长度”,用“宽”、“窄”形容“萼片宽度”和“花瓣宽度”)。之后,使用 NGEP,GA-AA 分别对实验数据集进行挖掘,实验参数是  $\min\_spt = 0.1$ ,  $\min\_cnf = 0.5$ 。实验结果如表 3 所列。

表 3 NGEP 和 GA-AA 的实验结果

统计后的平均值	时间	规则	准确率	测试数据集的
算法	/s	个数	(%)	预测精度(%)
NGEP	11.9	32	53.4	41.2
GA-AA	16.8	25	48.7	32.3

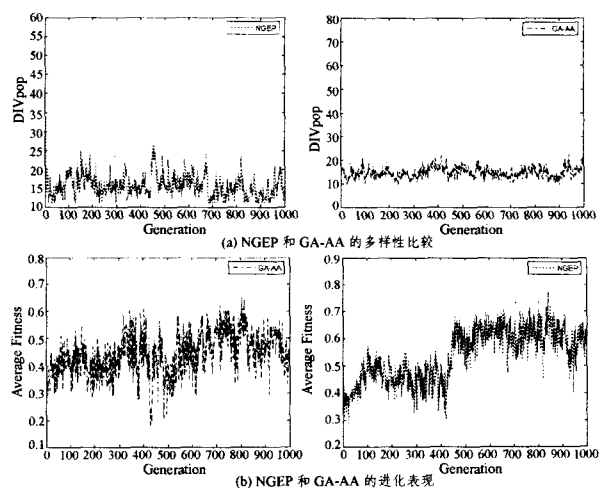


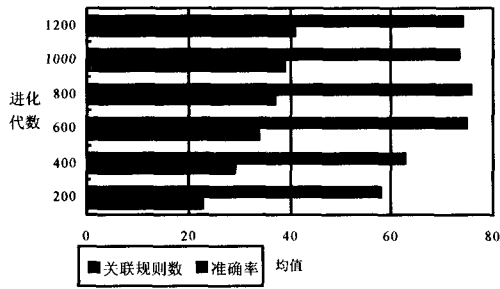
图 2

图 2(a)和图 2(b)显示了 NGEP 算法和 GA-AA 算法在进化表现和多样性两个方面的比较结果。可以看到,NGEP 算法在处理大规模数据上有优势。尽管 NGEP 算法在算法前期(450 代之前)陷入局部最优,但是它充分利用了小生境技术逃逸了局部极值;在后期,NEGP 算法的效率有明显的提高,很大程度上摆脱了局部最优,并且成功得到了最优规则集。

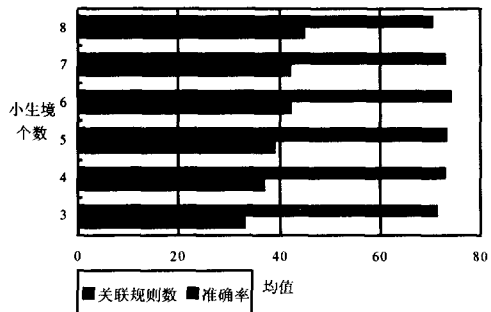
表 4 是从实验结果中抽取出来的一些规则。

	染色体	关联规则
规则 1:	$\Lambda \vee \Lambda \rightarrow \text{heehfea} \rightarrow \vee \vee \vee$ $\text{faabbc} \rightarrow \text{fV} \wedge \wedge \text{bddfce}$	如果花瓣长度短,则 Iris 是 Setosa
规则 2:	$\vee \vee \text{cd} \rightarrow \text{dadfbec} \wedge$ $\rightarrow \text{ggbefdgc} \wedge \vee \text{hb} \rightarrow \text{bcaefhh}$	如果花瓣宽度宽,则 Iris 是 Virginica;
规则 3:	$\Lambda \wedge \wedge \text{g} \rightarrow \text{gghahe} \rightarrow \vee$ $\rightarrow \text{eebfcdg} \wedge \text{aegeaacgbh}$	如果萼片长度短,花瓣长度短,花瓣宽度窄,则 Iris 是 Setosa
规则 4:	$\Lambda \wedge \wedge \text{e} \rightarrow \text{gafcdh} \wedge \wedge \text{c}$ $\rightarrow \text{abedgha} \rightarrow \vee \rightarrow \text{ffbfngc}$	如果萼片宽度窄,花瓣长度长,花瓣宽度窄,则 Iris 是 Virginica
规则 5:	$\text{g} \wedge \vee \rightarrow \text{bdcaecae} \wedge \text{c} \vee \text{d}$ $\rightarrow \text{dfebae} \wedge \rightarrow \wedge \text{efebahcb}$	如果萼片长度短,萼片宽度窄,花瓣长度短,花瓣宽度窄,则 Iris 是 Setosa
规则 6:	$\rightarrow \wedge \wedge \rightarrow \text{bhagcdd} \wedge \wedge \wedge \text{bd}$ $\text{ddecab} \wedge \rightarrow \wedge \rightarrow \text{cfhehc}$	如果萼片宽度长,萼片宽度宽,花瓣长度长,花瓣宽度宽,则 Iris 是 Virginica

**结束语** 本小结的目的是通过实验确定 NGEP 算法的最佳参数组合,以得到有效解,减少计算时间。相对于传统的 GEP 算法,NGEP 具有独特的小生境进化机制。所以下面的实验是基于 4.1 节(ASD 数据集)小生境的进化代数 and 个数来设计的。实验步骤简化如下:首先,仅有一个参数(进化代数或者小生境的个数)发生改变,接着表 1 中其他的参数曲线发生变化,使得结果收敛。得到的实验结果如图 3(a)、图 3(b)所示。



(a) 进化代数对 NGEP 算法的影响



(b) 小生境个数对 NGEP 算法的影响

图 3

从图 3 中得到的小生境的进化代数和数量都明显地影响到了算法的收敛速度。

实验显示进化代数的最佳取值是 800, 不仅收敛速度快,而且可以获得最佳的准确率。在进化代数的参数取值为其他

值时,例如取值 200,400,600,准确率都不及取值 800 时的准确率。也就是说,取其他值时,NGEP 算法产生了一些拥有低准确率的规则,虽然规则的数量有所上升,但是规则的质量却在下降。在进化代数取值 1000,1200 时,虽然精确度比较接近取值 800 时的准确率,但是导致了计算时间的增加。

同样从图 3(b)中可以看到小生境数量的上升能够增加挖掘所得到的规则数目,但同时带来了准确率的下降。以上结论可以从小生境数量取值 7,8 时明显看出来。所以小生境数量的取值不能盲目地选择最大值,它需要同时考虑规则数量和规则质量。基于上述标准,从图 3(b)中可以得到 6 是小生境数量的最佳参数取值。

**结束语** 关联规则的挖掘是一个多目标问题,所以找到一个最优解是很困难的。本文采用了结合 GEP 的小生境技术,致力于搜寻规模较大的解空间。NEGP 算法首先进行小生境演化,融合小生境并剔除同构的优秀个体,然后对小生境解进行笛卡儿交叉以产生更好的结果。对于 NGEP 的挖掘能力和效率,本文进行了两个实验。实验证明,NGEP 算法能够得到质量较高的规则集,并且有很快的收敛速度。

但是,与其他的智能算法类似,对于如何提高 NGEP 算法挖掘的效率,还有很多的工作要做:

- 1) 构建一个好的初始种群,可以在一定程度上加快最优解的收敛。
- 2) 测试新的小生境演化策略或者集成遗传操作的聚集技术,检测这些新的技术能否提高算法的收敛速度和准确性。
- 3) 在本文中挖掘关联规则时,仅考虑了两个重要的参数:支持度和置信度。如果能够结合其他的参数,例如综合度和兴趣度<sup>[4,6]</sup>,可以得到更加全面的挖掘结果。
- 4) 此外,本文仅关注了数值属性和普通关联规则,分类属性和模糊关联规则也是值得研究的方向。

## 参考文献

- [1] Nevcihan D. An application of apriori algorithm on a diabetic database[C]//Conference(KES 2005), Proceedings. 2005;398-404
- [2] Cristian A, Mitica C. Grid implementation of the apriori algorithm[J]. Advances in Engineering Software, 2007, 8(5):295-300
- [3] Freitas A A. A survey of evolutionary algorithms for data mining and knowledge discovery[M]// Ghosh A, Tsutsui S, eds. Advances in Evolutionary Computing. New York; Springer-Verlag, 2003; 819-845
- [4] Romão W, Freitas A A, Gimenes I M de S. Discovering interesting knowledge from a science and technology database with a genetic algorithm[J]. Applied Soft Computing, 2004, 4(2):121-137
- [5] 沈国强,覃征. 一种新的多维关联规则挖掘算法[J]. 小型微型计算机系统, 2006, 27(2):291-294
- [6] Nodaet E A. Discovering interesting prediction rules with a genetic algorithm[C]// Proceedings of the Congress on Evolutionary Computation, 1999; 1322-1329
- [7] Freitas A A. Data Mining and Knowledge Discovery with Evolutionary Algorithms[M]. New York; Springer-Verlag, 2002
- [8] Mutation F C. Transposition, and recombination; An analysis of the evolutionary dynamics[A]// Proceedings of the 6th Joint Conference on Information Sciences, 4th International Workshop on Frontiers in Evolutionary Algorithms [C]. 2002; 614-617
- [9] Ferreira C. Genetic representation and genetic neutrality in gene expression programming [J]. Advances in Complex Systems, 2002, 5(4):389-408
- [10] 邹晓峰,陆建江,宋自林. 基于模糊分类关联规则的分类系统[J]. 计算机研究与发展, 2003, 40(5):651-656
- [11] Zhang Jun, Huang De-shuang, Lok Tat-ming, et al. A novel adaptive sequential niche technique for multimodal function optimization[J]. Neurocomputing, 2006, 69:2396-2401
- [12] Wei Wei, Wang Qi, Wang Hua, et al. The feature extraction of nonparametric curves based on niche genetic algorithms and multi-population competition[J]. Pattern Recognition Letters, 2005, 26:1483-1497