基于互信息下粒子群优化的属性约简算法

续欣莹[1],张 扩[1],谢 珺[1],谢 刚[2]
(1. 太原理工大学,山西晋中 030060;2. 太原理工大学国际教育交流学院,山西太原 030024)

摘 要: 最小属性约简是粗糙集理论中属性约简的优化问题. 在寻找最小属性约简的问题上,基于粒子群优化的属性约简算法(ARPSO 算法)优于传统的属性约简算法. 在现有的 ARPSO 算法中,正域部分通常被作为启发式信息,但是它并不能够很好地衡量不确定性,而互信息是粗糙集理论中一种更有效的度量不确定信息的重要工具. 为此,提出基于互信息下的粒子群优化的属性约简算法(MIPSO 算法),该算法把互信息作为适应度函数,通过增强粒子能迅速靠近吸引子的这一特性,改进了内嵌区域震荡搜索的粒子群优化算法(简记为 RSPSO 算法),防止算法较早的陷入局部最优,使得粒子群中的粒子更快的找到最优值,因此使得算法尽可能实现全局收敛. 实验结果表明,该算法不仅提高了寻优的能力,加快了算法的速度,提升了算法的精度,而且也能够使得约简后剩余属性的互信息值与约简前所有属性的互信息值近似相等.

关键词: 互信息;粒子群优化;最小属性约简;粗糙集;局部搜索模式
中图分类号: TP181 **文献标识码:** A **文章编号:** 0372-2112 (2017)11-2695-10
电子学报 URL: http://www.ejournal.org.cn **DOI:** 10.3969/j.issn.0372-2112.2017.11.017

An Attribute Reduction Algorithm Based on Mutual Information of Particle Swarm Optimization

XU Xin-ying[1], ZHANG Kuo[1], XIE Jun[1], XIE Gang[2]
(1. *College of Information Engineering, Taiyuan University of Technology, Jinzhong, Shanxi* 030060, *China*;
2. *College of International Education and Exchange, Taiyuan University of Technology, Taiyuan, Shanxi* 030024, *China*)

Abstract: Minimum attribute reduction is the optimum problem in the attribute reduction of the rough sets theory. To seek the minimum attribute reduction, the attribute reduction algorithm based on the particle swarm optimization (ARPSO algorithm) beats the traditional attribute reduction algorithm. In existed ARPSO algorithms, the positive region is usually taken as the heuristic information, however, it is not precision enough to measure the uncertainty. The mutual information is a more efficient tool to measure the uncertainty in the rough sets theory. To handle this problem, an attribute reduction algorithm based on the particle swarm optimization takes the mutual information (MIPSO algorithm) as a term in the fitness function, The proposed MIPSO algorithm improves the regional shock search embedded particle swarm optimization algorithm (RSPSO) by enhancing the speed which the particle is close to the attractor, preventing from being local optimum early and finding the optimum as soon as possible. Consequently, the global convergence of the MIPSO algorithm is guaranteed as soon as possible. The experimental results show that the proposed MIPSO algorithm not only improves the optimization ability, accelerates the speed and improves the accuracy, but also can keep the mutual information value of all attributes before reducing approximately equal to the value of remaining attributes after reducing.

Key words: mutual information; particle swarm optimization; minimum attribute reduction; rough set; local search schemes

收稿日期:2016-01-29 修回日期:2016-06-07;责任编辑:蓝红杰
基金项目:人社部留学回国人员科技活动择优资助项目(No. 2013-68);山西省自然科学基金(No. 2014011018-2);山西省回国留学人员科研资助项目 No. 2013-033, No. 2015-045)

1 引言

波兰数学家 Pawlak 在 20 世纪 80 年代初提出了粗糙集(RoughSet,RS),它是一种处理不完备信息的数学理论[1].当前全世界的数据迅速地增长,且数据处于动态更新的状态,数据处理技术成为了数据挖掘领域中重要的一部分[2].在粗糙集理论中,属性约简是粗糙集应用不可或缺的分支,也是挖掘和简化数据的重要步骤[3,4].为了获得最精简的特征集,需要对决策表进行最小属性约简.利用粗糙集理论进行属性约简与特征选择已经在许多领域得到广泛应用,例如海量数据分析、聚类分析、神经网络、模式识别、过程控制、天气情况、诊断疾病、机器学习、大数据处理等[5~9].

粒子群优化算法(Particle Swarm Optimization,PSO)是由 Kennedy 和 Eberhart 于 1995 年提出的一种寻优算法[10].它可以解决大量非线性、不可微和多峰值的复杂问题,并能应用于函数优化、神经网络训练、属性约简等多个领域[11].但是该算法容易陷入局部最优.为了提升算法的寻优能力,使得算法能够全局收敛,相关学者做了大量的研究,对算法进行了改进[12].大多数改进的粒子群算法都是将变异操作引入,当粒子趋于局部收敛时或算法后期使粒子产生变异,使得粒子能够跳出局部收敛位置,从而实现全局收敛.

目前,采用粗糙集理论进行属性约简的方法有许多种,这些属性约简方法基本上采用的是启发式算法,能够得到一个约简.文献[13]提出了基于可行域的遗传约简算法(简记为 GA 算法),文献[14]提出了经典的启发式最小属性约简算法(简记为 Hu 算法),文献[15]提出了基于二进制的粒子群优化的属性约简算法(简记为 PSO 算法),文献[16]提出了基于免疫粒子群优化的最小属性约简算法(简记为 IPSO 算法),文献[17]提出了基于二进制粒子群优化的一个最小属性约简算法(简记为 BPSO 算法),文献[18]提出了最小属性约简问题的一个有效的组合人工蜂群算法(简记为 CABC 算法).前两种算法都能够对数据进行属性约简,但不一定能够得到最小的属性约简;后四种算法都是基于代数观下的正域为属性重要度,但是利用正域来衡量数据的不确定性是不够准确的,因为基于正域的属性重要度具有较弱的启发信息,对不确定性度量能力较差.

互信息是粗糙集属性约简中基于信息观下的一种度量工具,能够度量数据信息系统的不确定性,也能够用来去除冗余属性,样本中属性的增减直接影响互信息大小的变化.文献[19]已经证明在一致决策表中,代数观和信息观是等价的,而在不一致决策表中,信息观的约简包含代数观,所以运用互信息来进行属性约简要比使用正域来进行约简的效果要好一些.因此,为了能够找到数据属性的最小属性约简,同时也为了能够保证属性约简后的准确度,本文基于互信息度量,在标准粒子群算法的理论基础上深入的研究了内嵌区域震荡搜索的粒子群优化算法,并对算法中吸引子的求解过程做了进一步的改进,利用互信息度量属性重要度,提出了 MIPSO 算法,并采用 UCI 数据集进行实验,实验结果表明该算法是有效的.

2 互信息与粗糙集属性约简

定义 1 定义 $IS=(U,E,V,f)$ 为一个数据信息系统,其中 U 为非空数据样本集,称为论域空间,E 是有限属性集,$V=\cup_{a\in E}V_a$,V_a 表示属性 a 的值域,$f:U\times E\to V$ 是一个信息度量函数,即对 $x\in U,a\in E$ 有 $f(x,a)\in V_a$.任一属性子集 BE 决定了一个二元等价关系 $IND(B)$:$IND(B)=\{(x,y)\in U\times U\mid a\in B,f(x,a)=f(y,a)\}$.$U/IND(B)$ 或者 U/B 称为 U 的一个等价划分,是论域 U 上的一个知识.该等价划分包括多个等价类,每个等价类称为一个知识粒.

定义 2[19] (互信息)设 U 是论域,$K_1=(U,P)$ 和 $K_2=(U,Q)$ 是关于 U 的两个知识库,其中 $P=\{P_1,P_2,\cdots,P_m\}$,$Q=\{Q_1,Q_2,\cdots,Q_n\}$,P 与 Q 的互信息定义为 $E(Q;P)=\sum_{i=1}^{n}\sum_{j=1}^{m}\frac{|Q_i\cap P_j||Q_i^c\cap P_j^c|}{|U||U|}$.其中 Q_i^c 为 Q_i 在 U 上的补集,P_j^c 为 P_j 在 U 上的补集.

定理 1[19] 设 U 是论域,$K_1=(U,P)$ 和 $K_2=(U,Q)$ 是关于 U 的两个知识库,则有:

$$E(Q;P)=E(Q)-E(Q\mid P).$$

定义 3 定义 $DS=(U,C\cup D,V,f)$ 为一个数据决策系统,其中 C 为条件属性,其等价划分为 $U/C=\{C_1,C_2,\cdots,C_m\}$,$D$ 为决策属性,其等价划分为 $U/D=\{D_1,D_2,\cdots,D_n\}$.

定义 4 (决策表的互信息)设数据决策系统 $DS=(U,C\cup D,V,f)$,其中 C 为条件属性,D 为决策属性,BC,则定义 B 对 D 的互信息为 $E(D;B)=\sum_{i=1}^{n}\sum_{j=1}^{m}\frac{|D_i\cap B_j||D_i^c\cap B_j^c|}{|U||U|}$,其中 B 表示的是条件属性下关于 U 的类元素,即 $B=\{B_1,B_2,\cdots,B_m\}$,D 表示的是决策属性下关于 U 的类元素,即 $D=\{D_1,D_2,\cdots,D_n\}$.其中 D_i^c 为 D_i 在 U 上的补集,B_j^c 为 B_j 在 U 上的补集.

定理 2[19] 设数据决策系统 $DS=(U,C\cup D,V,f)$,B 和 P 分别为 U 上的两个属性集合.若 $IND(B)=IND(P)$,则有 $E(D;B)=E(D;P)$.

证明 因为 $IND(B)=IND(P)$,所以 B 和 C 在 U 上得到的划分是相同的,故有 $E(D;B)=E(D;P)$.

定义 5　设数据决策系统 $DS=(U,C\cup D,V,f)$,B,AC,$U/B=\{X_1,X_2,\cdots,X_m\}$,$U/A=\{Y_1,Y_2,\cdots,Y_n\}$,若 $X_i\in U/B(Y_j\in U/A(X_iY_j))$,则称 B 比 A 更细,记为 $B\leqslant A$;若 $X_i\in U/B(Y_j\in U/A(X_iY_j))$,则称 B 比 A 严格细,记为 $B<A$.

定理 3[19]　设 U 为论域,$K_1=(U,P)$ 和 $K_2=(U,Q)$ 是关于 U 的两个知识库,D 是 U 的决策属性. 如果 $P<Q$,则有 $E(D:P)\geqslant E(D:Q)$.

定理 4　设数据决策系统 $DS=(U,C\cup D,V,f)$,其中 C 为条件属性,D 为决策属性,且论域 U 是在 C 相对于 D 是一致的,则 C 中的一个属性 c 相对于 D 是不必要的,其等价条件为 $E(D:C)=E(D:C-\{c\})$.

证明　由定理 2,定义 7 可证,略.

定理 5　设数据决策系统 $DS=(U,C\cup D,V,f)$,其中 C 为条件属性,D 为决策属性,BC,若 $E(D:B)=E(D:C)$,且不存在 $R\subset B$,使得 $E(D:R)=E(D:C)$,则称 B 为 C 的一个相对于决策属性 D 的属性约简.

证明　由定理 4、定理 5 可证,略.

3　互信息下的粒子群属性约简

PSO 算法是一种基于群体的具有全局寻优能力的优化工具. 该算法具有很多优点,如算法收敛速度较比较,涉及参数较少,实现起来较为简单,具有比较强的全局优化能力. 但是算法也存在着不足. 比如在粒子后期,粒子在行进的过程中慢慢的趋于相同的状态,群体逐渐变为单一性,使得算法的计算速度以及收敛速度逐渐由快变慢,导致最终不能找到最优解. 基于以上不足,本文对 PSO 算法进行了改进.

3.1　内嵌区域震荡搜索的粒子群优化算法(RSPSO)

RSPSO 算法[20]是汤继涛等在 2013 年提出的一种改进的粒子群算法. 该算法提出吸引子的概念,显著地提高了性能.

RSPSO 算法的基本步骤是:

Step1　初始阶段. 首先在初始的状态下,系统初始一组解作为最优解,粒子在寻找最优解的过程中,随着粒子速度减小,粒子慢慢的靠近吸引子点,说明粒子在搜索的过程中,吸引子点在吸引着粒子. 计算出粒子此时所在的位置与吸引子点的距离,也就是区域震荡因子 $\Delta_{i,j}(t)$,距离的大小反映粒子的聚集程度.

其中每个粒子吸引子点的公式为:

$$p_{i,j}(t)=\varphi_{i,j}(t)\times pb_{i,j}(t)+[1-\varphi_{i,j}(t)]\times gb_j(t)$$
$$\varphi_{i,j}(t)=\frac{c_1r_{i,j}(t)}{[c_1r_{1,j}(t)+c_2r_{2,j}(t)]} \tag{1}$$

区域震荡因子的公式为:

$$\Delta_{i,j}(t)=|x_{i,j}(t)-p_{i,j}(t)| \tag{2}$$

公式说明:公式(1)中 i 表示的是当前进行寻优的粒子,j 表示该粒子进行寻优的维度,t 表示当前进行迭代的次数. $pb_{i,j}(t)$ 为在当前迭代次数下粒子在第 j 维迄今搜索到的个体最优解;gb 代表整个群体迄今搜索到的全局最优解,$gb_j(t)$ 代表在当前迭代次数下 gb 在第 j 维的解;公式(2)中 $x_{i,j}(t)$ 为搜索前粒子所在位置. 每次迭代后各参数的值都是基于上次迭代的计算结果. 其中:

$$pb_i(t+1)=\begin{cases}x_i(t),F(x_i(t))>F(pb_i(t))\\pb_i(t),F(x_i(t))\leqslant F(pb_i(t))\end{cases} \tag{3}$$

$$gb(t+1)=\begin{cases}x_i(t),F(x_i(t))>F(gb(t))\\gb(t),F(x_i(t))\leqslant F(gb(t))\end{cases} \tag{4}$$

其中 $pb_{i,j}(t+1)$ 为 $pb_i(t+1)$ 在第 j 维的解,$F(x_i(t))$ 为粒子 i 在第 t 次迭代下进行震荡搜索后的适应度值,$F(pb_i(t))$ 为粒子 i 搜索到的个体最优解的适应度值;$gb_j(t+1)$ 为 $gb(t+1)$ 在第 j 维的解,$F(gb(t))$ 为当前迭代次数下搜索到的全局最优解的适应度值.

算法在进行迭代之前,首先通过适应度函数求出当前状态下的个体最优值以及全局最优值;在开始进行迭代之后,当 $t=1$ 时,先求出所有粒子吸引子点的位置,并通过适应度函数求出当前状态下粒子 i 的个体最优解以及全局最优解,并与之前得到的这两个解进行比较(公式 3、公式 4),对其进行更新. 当所有粒子都对个体最优解、全局最优解进行更新之后,完成了一次迭代,随即进行下一次迭代,也就是 $t=2$,以此循环,直到达到设定的迭代次数.

Step2　子震荡搜索过程简介. 该过程是以吸引子为中心,$\Delta_{i,j}(t)$ 为半径的区域来进行寻优,粒子进行震荡搜索的方向就是吸引子吸引粒子的方向,粒子震荡搜索一次,粒子的各维的所在的位置就会改变一次. 因此由于区域震荡因子线性减小,使得粒子能够在吸引子周围找到更佳的值.

其中当前震荡次数时的震荡幅度的公式为:

$$\Delta_{i,j}(k)=\Delta_{i,j}(S)-(\Delta_{i,j}(S)-\Delta_{i,j}(E))\times k/RS \tag{5}$$

公式(5)中 $\Delta_{i,j}(S)$ 是粒子与吸引子点之间的距离(震荡搜索的震荡幅度),它的值一般为 $2\times\Delta_{i,j}(t)$;$\Delta_{i,j}(E)$ 是限制粒子搜索最远距离的常数,它的值一般为 0;RS 是粒子在内嵌区域中震荡搜索的所有次数.

当粒子的每一个维度进行震荡搜索之后,它的位置都会发生改变,位置更新之后的公式为:

$$\begin{cases}x_{i,j}(k)=x_{i,j}(k)+flag\times\Delta_{i,j}(k)\\flag=1,(x_{i,j}(t)-p_{i,j}(t))\leqslant 0\\flag=-1,(x_{i,j}(t)-p_{i,j}(t))>0\end{cases} \tag{6}$$

其中 $x_{i,j}(k)$ 为第 k 次在内嵌的区域范围内吸引子的附

近震荡搜索的位置,$x_{i,j}(t)$为搜索前粒子所在位置;*flag*为控制粒子的寻优的方向,保证粒子不断的往吸引子的方向运动.

粒子震荡搜索的具体过程描述:粒子从第一维开始进行搜索,每次搜索都需要结合该粒子其他的还没有进行震荡过的维度进行适应度的计算,在震荡搜索完成之后选择适应度最大的位置,其他维度也用同样的方法来进行实现. 当粒子所有维度的最好的位置都找到之后再对粒子的适应度进行计算,如果该适应度的值要比历史最优极值要好,则该值取代震荡搜索前粒子的历史最优极值;如果该适应度的值要比种群的全局最优极值要好,则该值取代震荡搜索前种群的全局最优极值.

粒子群中的每一个粒子都通过以上步骤进行震荡搜索寻优,直至达到预定的最大迭代次数(记为 *maxgen*)为止,算法输出粒子群中适应值最高的粒子位置(可行解)作为寻优的解.

3.2 改进的内嵌区域震荡搜索粒子群优化算法(NRSPSO 算法)

在 PSO 算法中,如何在保证全局最优值不陷入局部极值的情况下更快的使种群聚集,是本算法优化的重点. 本文提出 NRSPSO 算法,改进了 RSPSO 算法中吸引子的公式.

根据 PSO 公式,速度矢量公式为:

$$v_i(t+1)=wv_i(t)+c_1r_1(pb_i(t)-x_i(t))+c_2r_2(gb(t)-x_i(t)) \tag{7}$$

其中 $x_i=(x_{i1},x_{i2},\cdots,x_{in})$ 为粒子目前位置;$v_i=(v_{i1},v_{i2},\cdots,v_{in})$ 是离子的速度;pb_i 为迄今搜索到的个体最优解;gb 代表整个群体迄今搜索到的全局最优解;w 称为惯性权重;t 表示当前的迭代次数;c_1、c_2 为常数;$r_1\in(0,1)$,$r_2\in(0,1)$. 当 $pb_i=$ 的时候,粒子朝相同的位置进行移动,令该点为吸引子点. 设 $t_1\in(0,1)$、$t_2\in(0,1)$,吸引子点的公式表示为:

$$p(t)=(1-t_1)pb_i(t)+(1-t_2)gb(t) \tag{8}$$

其中,t 表示当前迭代的次数.

该公式只保留当次迭代下经过优化的结果,通过加入两个随机变量的作提高公式(6)的随机性,防止吸引子点陷入局部最优,使吸引子点能够快速的靠近最优解.

根据 RSPSO 算法中吸引子的公式(1),可知粒子的全局最优值和局部最优值相等,且知道是在(0,1)上均匀分布的无关随机数,基于此,粒子的吸引子公式可以改为:

$$p_i(t)=(1-t_1)pb_i(t)+(1-t_2)gb(t) \tag{9}$$

其中,$t_1\in(0,1)$、$t_2\in(0,1)$.

公式说明:公式(9)中 t_1、t_2 表示的是两个 0 到 1 之间的随机数,$pb_i(t)$、$gb(t)$ 分别表示的是第 i 个粒子当前迭代次数下的个体最优解、当前迭代次数下全局最优解,与公式 3、公式 4 中表达的含义相同. 由公式(9)可以得到公式(10):

$$p_{i,j}(t)=(1-t_1)pb_{i,j}(t)+(1-t_2)gb_j(t) \tag{10}$$

公式(10)与公式(1)表示的含义相同,都为求粒子 i 在第 j 维上吸引子点的位置. 同公式(1)相比,由于在算法迭代开始时,公式(10)引入两个随机变量,因此在每次迭代过程中,随机变量的加入增加了吸引子解的随机性,而且没有对算法的其他步骤造成干扰和影响,最终通过调整个体最优解和全局最优解的权重扰动吸引子位置,有利于找到更优解,增强了吸引子的寻优能力,使得算法优化性能提升.

算法的收敛性说明:在每个种群粒子的单一维度上与吸引子之间的 2 倍距离区间内(公式 5)进行内嵌区域的震荡搜索,持续寻找每个粒子的更优适应度值,通过修改种群中粒子位置和吸引子位置来优化种群,使种群向最优解收敛聚集,防止算法陷入局部最优,进而指导种群的优化.

3.3 本文提出的 MIPSO 算法

3.3.1 算法介绍

该算法是将基于二进制的粒子群优化算法、3.2 算法和粗糙集属性约简算法相结合的属性约简的方法.

Step1 算法通过运用在二进制粒子群算法中的原理,对每个粒子每一维度的解都对应“0”或者“1”,用来表示为使用或者不使用决策表中的相对应维数的属性. 通过这一思想的变换,“0”和“1”的数值被赋予了实际意义,实现了将粒子群算法与属性约简算法的结合.

Step2 定义 $NDT=(U,C,D)$ 为数据决策信息表,其中 U 为论域,C 为条件属性,D 为决策属性. 以互信息作为适应度函数,并将循环条件(也就是迭代次数)设定.

Step3 通过迭代寻优后得到种群在搜索空间中的全局最优解,并且转换为基于互信息的属性约简理论中的属性条件选择结果,最终得到了满足所给条件的决策信息表的最小属性约简.

整个算法将基于互信息属性约简理论和经过改进之后的粒子群算法结合在了一起,而且从已有的粒子群算法和粗糙集属性约简方法相结合的文章[17,18]来看,利用粒子群算法实现属性约简算法是可行的.

在种群中,可以通过使用与条件属性个数相等的二进制数字串表示粒子. 通过对决策信息表分析和理解,可以用优化函数的解来求解最小属性约简问题:

设数据决策系统 $DS=(U,C\cup D,V,f)$,m 为 C 的个数,φ 为给定的阈值. P 表示粒子群内一个粒子的解,$P=\{P_1,P_2,\cdots,P_m\}$,其中每一个元素 P_i 都对应着 C 中每一个元素 C_i,$P_i\in\{0,1\}$,表示粒子每一维的分量,则:

$$\begin{cases}\min\ C(P) = \sum_{i=1}^{m} P_i \\ E(D:C) - E(D:C) \leqslant \varphi\end{cases} \tag{11}$$

当 $P_i=1$,说明该维对应属性 C_i 被选中;$P_i=0$,说明该维所对应属性 C_i 没有被选中,c 为 P 中 P_i 为 1 所对应条件属性集合,也就是 C_i 被选中集合;当公式(11)成立,说明找到最小属性约简.

3.3.2 粒子适应度函数

为了能够反映实际问题的真实程度,需要对粒子群中的粒子进行评价,而粒子适应度函数作为唯一确定性标准,它的选取直接确定全局最优解的具体位置. 虽然无法知晓这个最优解的具体位置,但是却等待着种群粒子对这个最优解位置进行搜索. 因此,粒子适应度函数的制定直接影响到粒子群的寻优效果. 通过分析一般信息系统决策表属性约简的要求,适应度函数可以定义如下:

$$F(p)=\begin{cases}m/k(p)+E(D:C),E(D:C)-E(D:C)>\varphi \\ 3m/2k(p)+E(D:C),E(D:C)-E(D:C)\leqslant\varphi\end{cases} \tag{12}$$

其中,$F(P)$为适应度函数,m 为总的条件属性的个数,$k(p)$为条件属性使用的个数,c 为 P 中 P_i 为 1 所对应的条件属性集合,也就是 C_i 被选中的集合. $E(D:C)$是关于被选中的条件属性的互信息,$E(D:C)$为包含全部属性集合下的互信息. φ 为给定的阈值.

该函数的设计目的是在使用条件属性最少的情况下得到决策信息系统最大的互信息值. 通过上式可以知道,粒子的解中最后存在的属性个数 $k(p)$越少,函数值就会越大;粒子中对应属性的互信息的值 $E(D:C)$越大,函数的适应度值也越大;当互信息的值近似等于所有条件属性的互信息的值,公式 12 下半部分成立. 由于 $E(D:C)$与 $E(D:C)$都是大于等于 0 小于等于 1 的数,而且 $E(D:C)$只是略大于 $E(D:C)$,所以为了使公式(12)下半部分成立时得到的值相比公式(11)上半部分成立时得到的值更大一些,加入因子 3/2. 当公式(12)下半部分成立,同时粒子的解中最后存在的属性个数 $k(p)$最少时,得到的适应度值最大,该粒子的解也就是最小属性约简.

3.4 算法描述

在算法 1 中,算法一次迭代的计算量主要由适应度函数中的互信息的计算次数来决定. 设 p 为实际的迭代次数,q 为粒子的个数,n 为样本的个数,m 为条件属性的个数,在最坏的情况下,计算互信息的复杂度为 $O(mn\log n)$. 所以,该算法在最坏情况下的迭代的计算量为 $O(p\times q\times mn\log n)$,在相同的迭代次数情况下,该算法与相同群体规模的其他群体智能算法的计算量大致相同.

算法 1　MIPSO 算法

输入:初始化种群及相关参数. 设 $DS=(U,C\cup D,V,f)$为一个数据决策系统,种群规模 *sizepop*,每个粒子的维度 D,种群的迭代次数 *maxgen*,粒子区域震荡搜索次数 *RS*,适应度值为 $F(P)$,个体最优值 pb_i,种群的全局最优解 *gb*.

输出:最小约简 $R_{\min}$.

Step 1 根据 *sizepop*、D 随机生成种群 M,并对 M 中所有粒子进行二进制离散化,得到二进制粒子群 P;

Step 2 For $i=1$:*sizepop*

利用公式(12)计算 $F(P_i)$;

end For

选出最大 $F(P_i)$,得出 pb_i、*gb*、$F(pb_i(t))$、$F(gb(t))$.

Step 3 For $i=1$:*maxgen*

For $j=1$:*sizepop*

For $t=1$:$D-1$

根据公式(9)、(10)计算每个粒子吸引子 *pi* 位置;

end For

For $t=1$:$D-1$

根据公式(2)计算出粒子 i 在 t 维的区域震荡因子;

if $M_{i,j}(t)-pi_{i,j}(t)>0$

flag = −1;%*flag* 在公式(6)中需要用到

else

flag = 1;

end if

For $k=1$:*RS*

根据公式(5)计算出粒子 i 当前震荡次数时的震荡度 $\Delta_{i,j}(k)$;

粒子 i 的第 k 维按照公式(6)进行震荡搜索得到 $P_{i,t}(k)$,并将 $P_{i,t}(k)$进行处理得到 A;

if *rand*(1) < A

$A=1$;

else

$A=0$;

end if

求此时粒子 i 的适应度 $F_k(P_i)$;

end For

将 $F_k(P_i)$中最大的所对应的 A 赋给 $P_{i,t}$;

求震荡搜索后粒子 i 的适应度 $F(P_i)$;

end For

if $F(P_i)>F(pb_i)$

$pb_i=P_i$;

$F(pb_i)=F(P_i)$;

end if

if $F(P_i)>F(gb)$

$gb=P_i$;

$F(gb)=F(P_i)$;

end if

end For

end For

Step 4 输出最优解粒子 P_i,并且通过转换输出该粒子所对应的属性的集合 Y,Y 就是对应决策信息表满足 MIPSO 算法搜索条件的最小属性约简.

4 实验分析

为验证算法的有效性,实验分别采用文献[13]GA算法,文献[14]Hu算法,文献[15]PSO算法,文献[16]IPSO算法,文献[17]BPSO,文献[18]CABC算法与本文算法(MIPSO)对算法性能进行了评估,主要采用以下三种指标:

(1)约简程度的比较,评价算法的约简效果;

(2)收敛速度分析,收敛速度越快说明得到最优值的时间越短;

(3)有效性与高效性分析,比较各算法得到的最优值的相关指标情况.

实验采用普通PC机,操作系统为Win7旗舰版,CPU为Intel(R)Core(TM)i7,4G内存,开发工具为MATLAB R2014a版本,数据集的基本信息如表1、表2所示,其中表1为UCI中小类型的数据集,表2为大数据规模下的数据集,ID为数据集的编号,n和m分别是数据集对象的个数和条件属性的个数,Ncls是决策类的个数.其中Mushroom数据集去掉了非完备的对象.

表1 UCI数据集的基本信息

ID	UCI数据集	n	m	Ncls
1	Lymphography	148	18	4
2	Audiology	200	69	24
3	Car	1728	6	4
4	Vote	435	16	2
5	SPECT Heart(test)	187	22	2
6	Post	90	8	3
7	Tic-Tac-Toe	958	10	2
8	Monks	432	7	2
9	Soybean(small)	47	35	4
10	Breast Cancer	699	9	2

表2 大规模数据集的基本信息

ID	大规模数据集	n	m	Ncls
11	Car Certificate	217485	15	2
12	People	83968	105	2
13	Hotel	149740	21	2
14	Weather	101745	12	5

其中表2中的数据集11为某一车型合格证的数据集,在该数据中详细记录了某一车型合格证的相关参数,包括排放标准、排放量、燃油种类、功率、驱动形式、整备质量、车长、产量总计等数据项;数据集12为用于行人检测的图像数据集,包括MIT图像集、INRIA图像集、NICTA图像集、随机拍摄的图像集.其中图像集包括两类图像:有行人目标的正样本集以及没有行人目标的负样本集.将图像集转换成数字数据集,进而对其进行特征提取;数据集13为国内一些著名旅游网站上对某著名酒店的评价,其中数据集是从各网站经过筛选、累积而获得的.评价方面包括对酒店的价格、服务质量、卫生、地理位置等,通过分析这些评价得出住客最看重酒店的哪些方面;数据集14为某地区某时间段空气质量的数据,它是从数据堂网站中获得.通过分析判断影响该地区空气质量的因素.为了保证各数据集中属性量纲不对结果造成过大的影响,需要将各数据集的属性值进行相应的预处理.

4.1 约简效果分析

为了对约简效果进行分析,同时也为了保证实验客观真实性,本文算法与不同文献的算法分别使用相同的数据集(表1中小型数据集、表2中大型数据集)来进行比较,最终采用约简率来表示约简的效果.结果如图1、图2所示,其中图1、图2中各个图中横坐标1代表GA算法、2代表Hu算法、3代表PSO算法、4代表IPSO算法、5代表BPSO算法、6代表CABC算法、7代表MIPSO算法.

约简率的公式如下:

$$rate = (|C| - |R|)/|C| * 100\% \qquad (13)$$

这里$|C|$表示的是数据集的总条件属性个数,$|R|$表示约简后条件属性的个数.从公式中可以知道,分母$|C|$是不变的,如果分子越大,也就是$|R|$越小,$rate$就会越大,约简的效果会越好.

4.1.1 约简效果比较

在这里将大型数据集和小型数据集放在一起进行分析:从图1、图2可以看出,对于数据集Hotel、Monks、Breast Cancer,MIPSO算法的约简率最高,其次的是PSO、BPSO、IPSO、CABC这四种算法,而GA、Hu算法的约简率最低;对于数据集Car、Audiology、Vote,MIPSO算法的约简率最高,其次的是CABC算法的约简率,其他几种算法的约简率相同且最低;对于数据集Lymphography,CABC、MIPSO算法的约简率最高,其次的是BPSO、IPSO算法,然后是Hu、PSO算法,GA算法的约简率最低;对于数据集SPECT、Tic-Tac-Toe、People、Weather,MIPSO算法的约简率最高,其次的是BPSO、IPSO、CABC算法,GA、Hu、PSO算法的约简率最低;对于数据集Post、Car Certificate,MIPSO算法的约简率最高,而其他几种算法的约简率相同;对于数据集Soybean(small),MIPSO算法的约简率最高,其次的是Hu、PSO、IPSO、BPSO、CABC算法,GA算法的约简率最低.

综上所述,MIPSO算法约简率比较高,并且基本能够得到更小的属性约简,表明MIPSO算法具有良好的全局寻优能力.

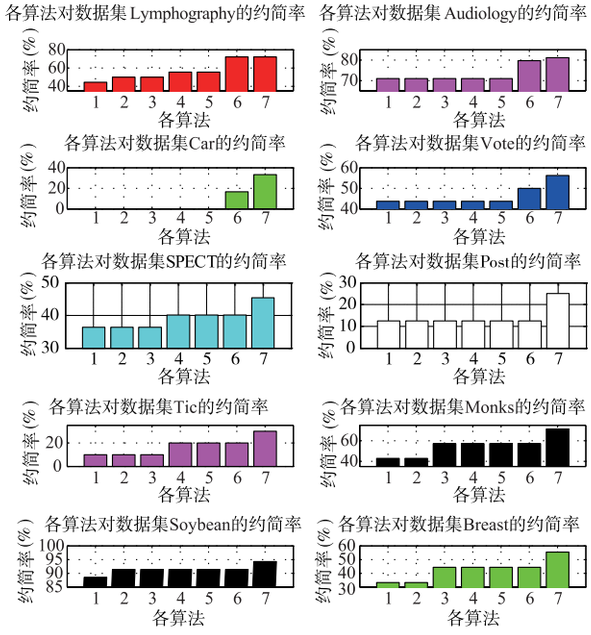

图1　各算法对小型数据集进行属性约简后的约简率

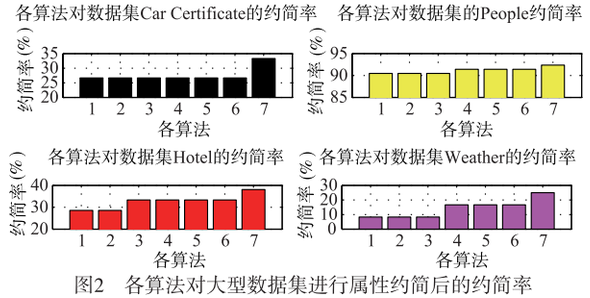

图2　各算法对大型数据集进行属性约简后的约简率

4.2　收敛速度分析

为了分析本文的收敛速度,实验对算法性能进行了评估及比较,采用对选取的六种算法同一数据集执行二十次,然后求平均值,得出平均迭代次数,结果如图3、图4 所示. 其中图3、图4 中各个图中的横坐标1 代表GA 算法、2 代表PSO 算法、3 代表IPSO 算法、4 代表BPSO 算法、5 代表CABC 算法、6 代表MIPSO 算法. 并用图5 表示 Breast Cancer、Lymphography 数据集在 IPSO 算法、BPSO 算法、CABC 算法以及 MIPSO 算法的属性约简迭代过程中属性个数的变化曲线(其中迭代次数大于500 的按照500 次来取值).

从图3、图4 可以看出,无论是在小型数据集上还是在大型数据集上,与 GA 算法、PSO 算法、IPSO 算法、BPSO 算法、CABC 算法相比,MIPSO 算法在各个数据集上不仅约简后属性的个数更少,而且迭代次数少,从图4 中还可以看出当面临大数据集时,MIPSO 算法和其他几种算法相比,都表现出了较优的收敛速度,而其他几种算法,CABC 算法的收敛速度要好于剩余几种算法;

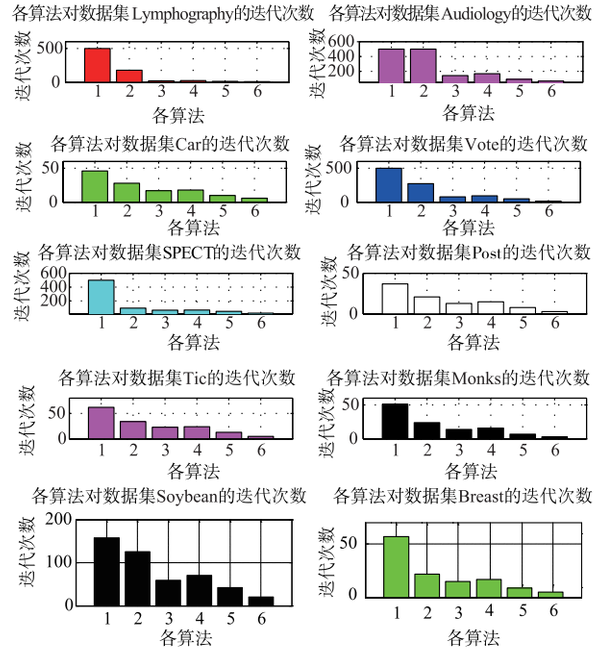

图3　迭代次数比较(小型数据集)

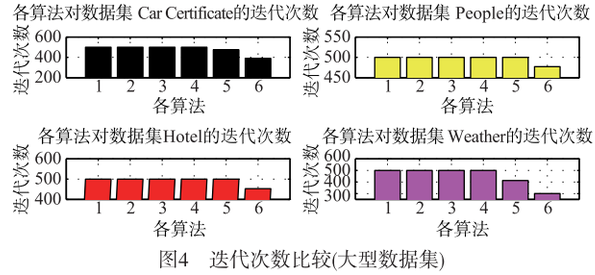

图4　迭代次数比较(大型数据集)

在图5 中可以看出,同其他几种算法相比,MIPSO 算法不仅具有更快收敛速度,而且能在较少迭代次数内完成约简,说明 MIPSO 算法具有较好粒子寻优能力.

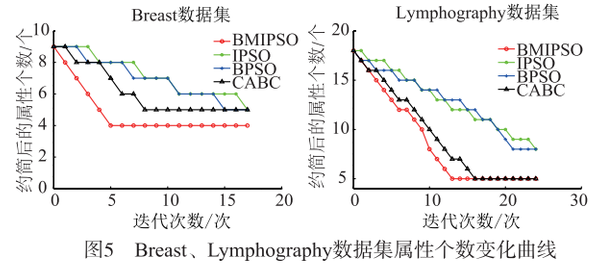

图5　Breast、Lymphography数据集属性个数变化曲线

4.3　有效性与高效性分析

当数据集进行属性约简之后,如何既能保证数据集约简率尽量高,又能保证约简后的数据互信息的值尽可能地靠近约简前数据的互信息值是判断算法有效性的关键;如何保证数据集在约简后剩余属性越少,又能保证约简后的数据互信息的值尽可能地靠近约简前数据的互信息值,还能保证迭代次数较少是判断算法

高效性的关键.

本文通过运用约简率、迭代次数、约简后的属性个数、互信息值相结合的加权计算公式(14)、公式(15)来判断各算法的有效性与高效性. 公式(14)、公式(15)的具体表达式如下所示:

$$Effectiveness=\begin{cases}\lambda(|C|-|R|)/|C|+(1-\lambda)E(D:c)\\ \lambda(|C|-|R|)/|C|+(1-\lambda)E(D:C)\end{cases} \tag{14}$$

$$He=\begin{cases}\alpha(|C|-|R|)/T+(1-\alpha)E(D:c)\\ \alpha(|C|-|R|)/T+(1-\alpha)E(D:C)\end{cases} \tag{15}$$

其中 λ、α 为加权系数,同公式(12)一样,$|R|$为表示约简后条件属性的个数,$(|C|-|R|)/|C|$为约简率,T 为约简所需的迭代次数,c 为各算法约简后剩余的条件属性;$E(D:c)$是关于被选中的条件属性的互信息,$E(D:C)$为包含全部属性集合下的互信息. 当 $E(D:C)-E(D:c)>\varphi$ 时(φ 为给定的阈值),公式(14)、公式(15)上半部分式子成立,当 $E(D:C)-E(D:c)\leqslant\varphi$ 时,公式(14)、公式(15)下半部分式子成立,通过比较 *Effectiveness*、*He* 值的大小来比较各算法的有效性和高效性. *Effectiveness*、*He* 值越大,说明算法约简结果具有较强的有效性以及高效性.

为了对比文献[15]PSO 算法、文献[16]IPSO 算法、文献[17]BPSO 算法、文献[18]CABC 以及本文提出的 MIPSO 算法的有效性和高效性,采用 UCI 数据集对这几种算法的 *Effectiveness* 值、He 值进行了测试,结果如图6~图9所示(其中迭代次数大于500的按照500次算).

从图6、图7可以看出,对于算法有效性的度量,

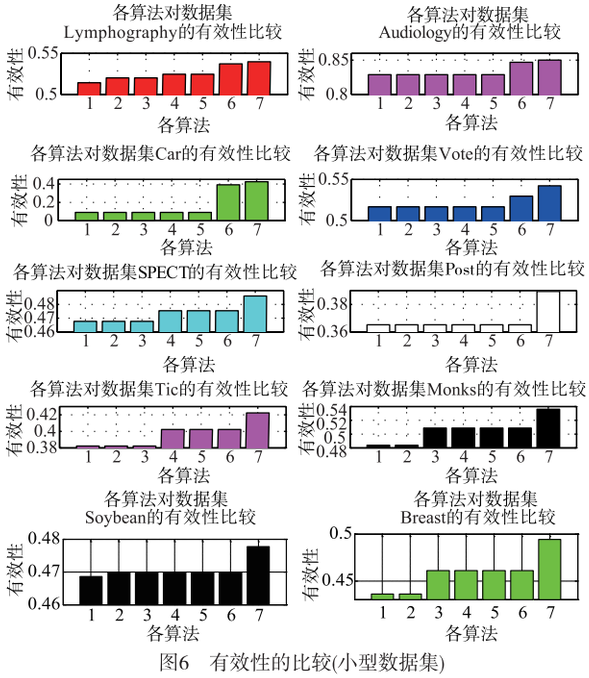

图6 有效性的比较(小型数据集)

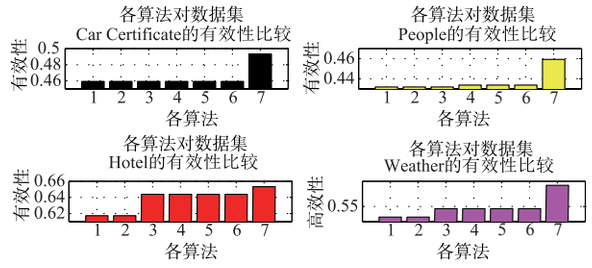

图7 有效性的比较(大型数据集)

MIPSO 算法无论是在小型数据集下还是大型数据集下的 *Effectiveness* 值都是最大的,说明 MIPSO 算法具有较强的有效性;其次是 CABC 算法,而且在 Lymphography、Audiology、Soybean(small)数据集下,CABC 算法得到的 Effectiveness 值只是略小于 MIPSO 算法得到的 *Effectiveness* 值,说明 CABC 算法也具有较强的有效性;在其他几种算法中,几种算法的 *Effectiveness* 值相近,GA 算法的 *Effectiveness* 值略小于其他几种算法,但是也存在着几种算法 *Effectiveness* 值相等的情况,说明这几种算法的有效性相近,GA 算法在一些数据集上略差于其他几种算法.

从图8、图9可以看出,对于算法高效性的度量,MIPSO 算法无论是在小型数据集下还是大型数据集下的 *He* 值都是最大的,说明 MIPSO 算法具有高效性;其次是 CABC 算法,虽然 *He* 的值比 MIPSO 算法的要小一些,但是该算法也具有较快的收敛速度;在其他几种算法中,BPSO 算法与 IPSO 算法的 *He* 值相近,GA 算法与 PSO 算法的 *He* 值相近,而且前者 *He* 值要大于后者的

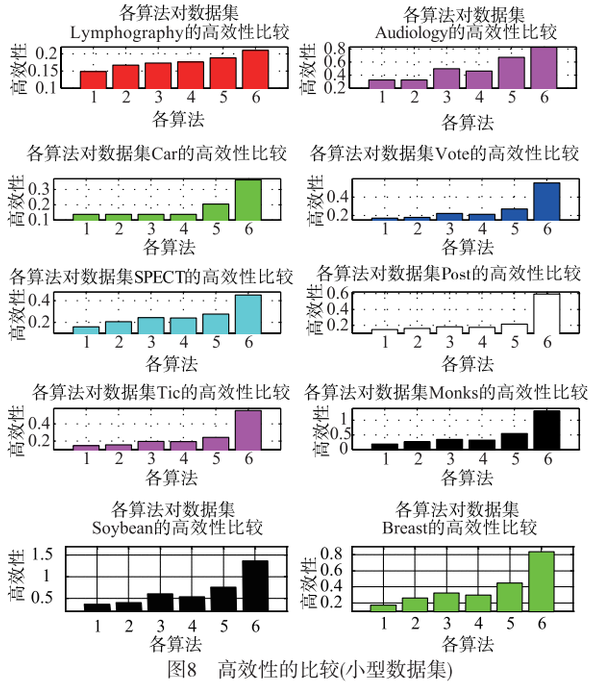

图8 高效性的比较(小型数据集)

He 值,说明 BPSO 算法与 IPSO 算法的收敛速度相近,GA 算法与 PSO 算法的收敛速度相近,BPSO 算法与 IPSO 算法的有效性要稍好于 GA 算法与 Hu 算法.

综上所述,MIPSO 算法不仅约简的效果较好,而且收敛速度快,具有较强的有效性和高效性.

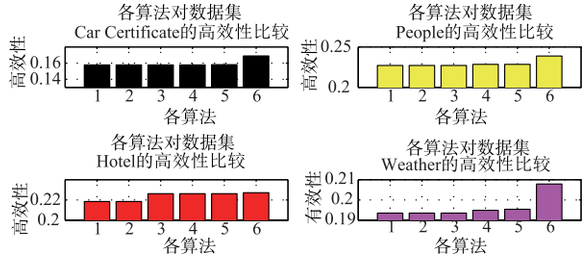

图9　高效性的比较(大型数据集)

5　结束语

本文将信息论中的互信息引入到基于粒子群优化的属性约简算法当中,并与改进的内嵌区域震荡搜索粒子群优化算法相结合,提出了 MIPSO 算法. 该算法利用增强粒子迅速的靠近吸引子这一特点,使得种群中的粒子能够较快地找到最优值,从而完成循环计算,提高了粒子寻优的能力. 实验结果表明该算法在约简率、收敛速度、有效性、高效性上都有较好的结果.

本文的研究进一步推广了粗糙集理论,为数据挖掘研究提供了一个新的方向.

参考文献

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

[2] 叶东毅. Jelonek 属性约简算法的一个改进[J]. 电子学报,2000,28(12):81 – 82.
Ye Dong-yi. An improvement to Jelonek's attribute reduction algorithm[J]. Acta Electronica Sinica,2000,28(12):81 – 82. (in Chinese)

[3] Xu Xinying, Liu Haifeng, Shen Xuefen, Xie Jun. The research of attribute reduction algorithm based on extension neighborhood relation[J]. Journal of Computational Information Systems,2013,9 (16):6613 – 6620.

[4] Jensen R, Shen Q. Fuzzyrough sets for descriptive dimensionality reduction [A]. Proc of 11th Internat Conf on Fuzzy Systems[C]. Hawaii,2002. 29 – 34.

[5] Zhang Xu, Song Ping. An attribute reduction algorithm based on clustering and attribute-activity sorting [A]. Computational and Information Sciences (ICCIS), 2010 International Conference on[C]. IEEE,2010. 709 – 712.

[6] Yao Yiyu, Deng Xiaofei. Quantitative rough sets based on subsethood measures[J]. Information Sciences, 2014, 267 (20):306 – 322

[7] He Yihai, et al. A fuzzy TOPSIS and Rough Set based approach for mechanism analysis of product infant failure [J]. Engineering Applications of Artificial Intelligence, 2016,47:25 – 37.

[8] Knorr E, Ng R, Tucakov V. Distance-based outliers: Algorithms and applications[J]. VLDB Journal: Very Large Databases,2000,8(3 – 4):237 – 253.

[9] Lang G, Li Q, Yang T. An incremental approach to attribute reduction of dynamic set-valued information systems[J]. International Journal of Machine Learning and Cybernetics, 2014,5(5):775 – 788.

[10] Kennedy J, Eberhart R C. Particle swarm optimization [A]. Proc of IEEE International Conference on Neural Networks[C]. IEEE,1995. 1942 – 1948.

[11] Vasant P, Ganesan T, Elamvazuthi I. An improved PSO approach for solving non-convex optimization problems. Proc of 2011 9th International Conference on ICT and Knowledge Engineering[C]. IEEE,2012. 80 – 87.

[12] Lei Kaiyou, Qiu Yuhui. A study of constrained layout optimization using adaptive particle swarm optimizer [J]. Journal of Computer Research and Development,2006,43 (10):1724 – 1731.

[13] 李订芳,章文,李贵斌,牛艳庆. 基于可行域的遗传属性约简算法[J]. 小型微型计算机系统,2006,27(2):312 – 315.
Li Dingfang, Zhang Wen, Li Guibin, Niu Yanqing. Genetic reduction algorithm based on feasible region[J]. Journal of Chinese Computer Systems,2006,27(2):312 – 315. (in Chinese)

[14] Hu Xiaohua, Cercone N. Learning in relational databases: a rough set approach[J]. Computational Intelligence, 1995, 11(2):323 – 338.

[15] Kennedy J, Eberhart R C. A discrete binary version of the particle swarm algorithm[A]. Proc of IEEE International Conference on Systems, Man and Cybernetics[C]. Piscataway, USA,1997. 4104 – 4109.

[16] 廖建坤,叶东毅. 基于免疫粒子群优化的最小属性约简算法[J]. 计算机应用,2007,27(3):550 – 555.
Liao Jiankun, Ye Dongyi. Minimal attribute reduction algorithm based on particle swarm optimization with immunity[J]. Journal of Computer Applications,2007,27(3): 550 – 555. (in Chinese)

[17] 叶东毅,廖建坤. 基于二进制粒子群优化的一个最小属性约简算法[J]. 模式识别与人工智能,2007,20(3): 295 – 300. (in Chinese)
Ye Dongyi, Liao Jiankun. Minimal attribute reduction algorithm based on particle swarm optimization with immunity[J]. Pattern Recognition and Artificial Intelligence,

2007,20(3):295 – 300.
[18] 叶东毅,陈昭炯. 最小属性约简问题的一个有效的组合人工蜂群算法[J]. 电子学报,2015,43(5):1014 – 1020.
Ye Dongyi,Chen Zhaojiong. An effcient combinatorial artificial bee colony alogorithm for solving minimum attribute reduction problem[J]. Acta Electronica Sinica,2015,43(5):1014 – 1020. (in Chinese)
[19] 杨明. 决策表中基于条件信息熵的近似约简[J]. 电子学报,2007,35(11):2156 – 2160.
Yang Ming. Approximate reduction based on information reduction in decision tables[J]. Acta Electronica Sinica,2007,35(11):2156 – 2160. (in Chinese)
[20] 汤继涛,戴月明. 内嵌区域震荡搜索的粒子群优化算法[J]. 计算机工程与应用,2013,49(21):33 – 36.
Tang Jitao,Dai Yueming. The oscillation search of PSO in embedded area[J]. Computer Engineering and Applications,2013,49(21):33 – 36. (in Chinese)

作者简介

续欣莹 男,1979 出生. 副教授,博士,研究方向为粒计算、数据挖掘与计算机视觉.
E-mail:xuxinying@ tyut. edu. cn

张 扩 男,1991 出生. 硕士研究生,研究方向为机器学习、数据挖掘与智能信息处理.
E-mail:zkzk4451@126. com