一种基于定位更新技术的人工蜂群聚类算法

汪佳玲,胡本木,孙越泓

(南京师范大学数学科学学院,江苏 南京 210023)

[摘要] 本文提出一种基于定位更新技术的人工蜂群算法,并将其应用于聚类分析问题.定位更新技术是在每一次待工蜂搜索结束后,充分利用当前最优解和最差解的信息,对最优解做进一步的更新.实验表明,基于定位更新技术的人工蜂群聚类算法,提高了算法利用先前的解来寻找更好解的开采能力.该算法与K-means算法、基于粒子群优化的聚类算法以及基于人工蜂群的聚类算法相比,具有更好的聚类性能.

[关键词] 定位更新技术,人工蜂群算法,聚类分析,开采能力

[中图分类号]TP391 [文献标志码]A [文章编号]1001-4616(2015)04-0095-08

An Artificial Bee Colony Clustering Algorithm Based on the Location Update Technology

Wang Jialing, Hu Benmu, Sun Yuehong

(School of Mathematical Sciences, Nanjing Normal University, Nanjing 210023, China)

Abstract: In this paper, an artificial bee colony (ABC) algorithm based on location update technology is proposed and applied to the problems of clustering analysis. The technology makes the algorithm fully use the information of current optimal solution and the worst solution to do further location update of current optimal solution after the search of onlookers. Experiments show that the ABC algorithm based on location update technology enhances the exploitation ability of applying the previous solutions to look for better solutions. The proposed algorithm also has better clustering performance compared with K-means algorithm, clustering algorithms based on particle swarm optimization and artificial bee colony.

Key words: location update technology, artificial bee colony algorithm, clustering analysis, exploitation ability

聚类分析是数据挖掘技术的重要组成部分,并被广泛应用于模式识别、图像分析和其他科学与工程领域.最常见的聚类算法是K-均值(K-means)聚类,其具有简单快速以及线性复杂度的特点.然而,K-均值算法对初始值敏感且易陷入局部最优.为克服这些缺点,很多启发式聚类算法被引入.1999年Krishma根据遗传算法的原理尝试以K-均值算子代替遗传算法中的交叉算子,提出一种混合遗传聚类算法[1];2000年Maulik采用聚类中心的浮点编码方式设计浮点数交叉和变异算法,提高了遗传聚类算法的搜索效率[2].但实验表明,当样本数目、维数和类别数较大时,这些算法常常过早地收敛于局部最优点,而且聚类问题的规模越大,这种早熟现象越容易发生.

近年来,用于搜索和优化的群智能算法已成为计算智能和人工智能领域的一个研究热点,并被成功地应用于聚类分析中.蚁群聚类方法最早由Deneubourg于1991年提出[3],并由Lumer和Faieta将该方法应用到数据分析领域[4].2002年Omran等提出一种基于粒子群优化算法(particle swarm optimization algorithm,PSO算法)的无监督图像分类方法[5],这是最早提出的基于PSO算法的聚类算法.2005年Karaboga成功地将蜜蜂采蜜原理应用于函数的数值优化并提出比较系统的人工蜂群算法[6](artificial bee colony algorithm,ABC算法).Zhang等[7]在2010年提出一种基于ABC的聚类算法,该算法将 n 个对象最佳分组到 k

收稿日期:2014-06-13.

基金项目:教育部人文社会科学研究青年基金(12YJCZH179)、国家自然科学基金项目(11371197).

通讯联系人:孙越泓,博士,副教授,研究方向:智能优化及图像处理,E-mail:05234@njnu.edu.cn

个集群中,其中Deb规则被用来选择每个候选解.2010年Karaboga和Ozturk[8]通过在模糊聚类上测试ABC算法的性能来表明ABC算法对模糊聚类的适用性.2011年Karaboga[9]提出基于ABC算法的新的聚类方法从而将ABC算法正式推广到聚类分析的求解.2014年Tan等提出一个基于关键初始化方法的EABC聚类算法来适应聚类的特殊解空间[10],Ozturk等采用ABC算法,提出一个新的目标函数对图像进行聚类[11].2015年Ozturk等提出一种改进的二进制人工蜂群算法研究动态聚类[12].

ABC及其相应算法都是采用轮盘赌机制进行选择更新,即算法更倾向于在比较好的解附近去搜索一个随机位置.为了让当前最优解得到更好的更新,本文将介绍一种新的基于定位更新技术的人工蜂群聚类算法,即在每次待工蜂搜索结束后加上一个定位更新的过程.本文选取UCI机器学习知识库(http://archive.ics.uci.edu/ml/)中5个典型数据集及2个人工数据集对新算法的性能进行测试,并将该算法的聚类结果与K-means、基于PSO和ABC的聚类算法的结果进行比较.

1　聚类问题

给定数据集 $X=(x_{ij})_{n\times p}$,其中 x_{ij} 对应于第 i 个对象的第 j 个实值特征,$i=1,2,\cdots,n;j=1,2,\cdots,p$.聚类算法是寻找一个分区 $C=\{C_1,C_2,\cdots\}$,当 $i,j=1,2,\cdots,k$ 时,满足(1) C_i 非空;(2)对 $\forall i\neq j$ 有 $C_i\cap C_j=\phi$;(3) $\bigcup_{i=1}^{k}C_i=X$.定义类内距离之和为:

$$Perf(X,C)=\sum_{i=1}^{n}\min\left\{\left\|x_i-v_j\right\|^2,j=1,2,\cdots k\right\},\tag{1}$$

其中 $\left\|x_i-v_j\right\|$ 表示对象 x_i 和集群 C_j 的中心 v_j 之间的相似性.本文使用欧氏距离作为相似性度量.总之,聚类问题就是寻找分 C^* 使 $Perf(X,C^*)$ 达到最小.

2　ABC算法和基于ABC的聚类算法

2.1　ABC算法

ABC算法是Karaboga于2005年为解决多变量函数优化问题而提出的一种模拟蜜蜂群采蜜行为的群智能算法[6].它非常简单、稳健,是一种基于群的随机优化算法.

在ABC算法中,蜂群主要分为3种:雇佣蜂、待工蜂和侦察蜂.雇佣蜂先去寻找食物源;待工蜂在舞蹈区等待雇佣蜂带回食物源的相关信息,并根据信息选择食物源;侦察蜂则完全随机寻找食物源.当某个食物源被雇佣蜂或待工蜂丢弃时,和此食物源对应的雇佣蜂变成侦察蜂.每个食物源的位置代表优化问题中目标函数的一个可能解,食物源的蜜源量对应于相应解的质量(或适应度),记为 fit,第 i 个蜂的适应度为 fit_i,

$$fit_i=\begin{cases}\dfrac{1}{1+f_i}, & f_i\geqslant 0,\\ 1+|f_i|, & f_i<0,\end{cases}\tag{2}$$

其中 f_i 是第 i 个蜂的目标函数值.

在ABC算法中,初始化蜜蜂总数为 N_s,雇佣蜂的数目为 N_e,待工蜂的数目为 N_u,N_e 与 N_u 相等且均为 $\frac{1}{2}N_s$.在算法的搜索过程中,首先在搜索空间中随机生成 N_s 个初始解(食物源位置),其 N_s 表示种群大小.每个解 $x_i(1,2,\cdots N_s)$ 是一个 D 维的向量,其中 D 是问题的维数.初始化后,整个蜂群将进行雇佣蜂、待工蜂和侦察蜂搜寻过程的重复循环,直到达到最大循环次数($MaxCycle$)或误差允许 ε.

在搜索过程开始阶段,每个雇佣蜂产生一个新解(新食物源位置):

$$v_{ij}=x_{ij}+\varphi_{ij}(x_{ij}-x_{kj}),\tag{3}$$

其中 $k\in\left\{1,2,\cdots,\frac{1}{2}N_s\right\},j\in\{1,2,\cdots,D\}$ 是随机选取的,$k\neq i,\varphi_{ij}$ 是 $[-1,1]$ 之间的随机数.雇佣蜂采用贪婪准则比较新解和旧解的质量.若新解的适应度值比旧解高,则雇佣蜂用新解取代旧解;否则,仍保留旧解.当所有的雇佣蜂完成搜索后,雇佣蜂在舞蹈区和待工蜂分享食物源信息.待工蜂由下式计算每个解的概率并选择食物源:

$$p_i = \frac{fit_i}{\sum_{j=1}^{\frac{1}{2}N_s} fit_j}, \tag{4}$$

其中,蜜源量大的雇佣蜂吸引待工蜂的概率大于蜜源量小的雇佣蜂.同样地,待工蜂在食物源附近仍采用式(3)产生一个新解,并根据贪婪准则检验.若新解的质量比旧解好,则待工蜂将保留新解;否则,仍保留旧解.在所有待工蜂完成搜索过程后,若一个解在有限次循环(*Limit*)内都不能被进一步改善,则该食物源会被丢弃.设食物源 x_j 被丢弃,则此食物源对应的雇佣蜂将变成侦察蜂,并产生一个新的蜜源代替 x_i :

$$x_{ij} = x_{\min j} + \text{rand}(0,1)(x_{\max j} - x_{\min j}), \tag{5}$$

其中 $j \in \{1,2,\cdots,D\}$.然后返回雇佣蜂搜索过程,开始重复循环.

在ABC算法中,有3个重要的参数:食物源数量($\frac{1}{2}N_s$)、阈值(*Limit*)以及最大迭代次数(*MaxCycle*).

2.2 基于ABC算法的聚类算法

蜂群的采蜜行为与聚类问题的对应关系如表1所示[9].

聚类的质量通过类内距离之和表示,即类内距离之和越小,聚类的质量越好;反之,聚类的质量越差.设样本数据集为 $X = \{x_1, x_2, \cdots x_n\}$,其中 x_i 为 D 维向量.ABC算法中的一个蜂(解)代表一个聚类中心集合.具体的算法步骤见算法1.

表1 蜂群的采蜜行为与聚类问题的对应关系

Table 1 The corresponding relationship between the behavior of honey bee colony and clustering problem

蜂群采蜜行为	聚类问题
食物源的位置	聚类中心
食物源的适应度	聚类中心的质量
寻找及采蜜的速度	求解速度
最大适应度	最佳聚类效果

算法1 基于ABC的聚类算法

Step 1 在数据空间内,根据(6)式随机产生 N_s 个初始解 $\{C_1, C_2, \cdots, C_{N_s}\}$:

$$C_i^j = X_{\min}^j + \text{rand}(0,1)(X_{\max}^j - X_{\min}^j), \tag{6}$$

其中每个 $C_i (i \in \{1,2,\cdots,N_s\})$ 都是一个 $M \times D$ 维的矩阵,代表一个聚类划分, M 表示总的聚类数目, D 为每个聚类中心的维数.用式(1)计算每个聚类中心的类内距离之和,将值较好(即较小)的前50%的蜂群作为初始时刻的雇佣蜂种群;

Step 2 每一个雇佣蜂在蜜源附近寻找新的聚类中心:

$$C_i^j = C_i^j + \varphi_i^j (C_i^j - C_k^j), \tag{7}$$

其中 $k \in \left\{1,2,\cdots,\frac{1}{2}N_s\right\}, j \in \{1,2,\cdots,D\}$都是随机产生的,且 $k \neq i, \varphi_i^j$ 是 $[0,1]$ 之间的随机数;

Step 3 每一个雇佣蜂根据贪婪准则比较新的聚类中心和原来的聚类中心质量,即若新的聚类中心的类内距离之和不变或变小,则用新的聚类中心代替原来的聚类中心;否则,仍保留原来的聚类中心;

Step 4 待工蜂按照概率 p_i 来选择聚类中心:

$$p_i = \frac{\min\limits_{k=1,\cdots,\frac{1}{2}N_s} Perf(X, C_k))}{Perf(X, C_i)}. \tag{8}$$

选择的原则是类内距离越小的聚类中心被选择的可能性越大.然后,每一个待工蜂按照自身类内距离之和的大小,采用轮盘赌选择机制,在其附近按照式(7)寻找新的聚类中心,并根据贪婪准则选择新的聚类中心;

Step 5 当某个雇佣蜂的位置周围搜索次数 *Bas* 达到一定阈值 *Limit* 而仍未找到更优位置时,利用公式(6)重新初始化该雇佣蜂的位置;

Step 6 若当前迭代次数达到预设的最大迭代次数 *MaxCycle* ,则停止迭代,输出最后一代找到的最优解,得到最后的聚类结果,即最佳聚类中心 C 以及此时的类内距离之和 *GlobalMin* ;否则转到Step 2,迭代次数加1.

3 基于定位更新技术的人工蜂群聚类算法

在ABC聚类算法中,待工蜂通过轮盘赌选择机制在其附近随机寻找新的解,即算法在一个较好的解

附近进行随机位置上的搜索.为了让较好的结果得到更好的、有目的的更新,在待工蜂搜索结束以后加上一个定位更新的过程,可以在一定程度上避免盲目搜索.首先在待工蜂搜索结束后,记录当前的最优解 $C_{bestnow}$ 和最差 $C_{worstnow}$;然后对解 $C_{bestnow}$ 和 $C_{worstnow}$ 的每一维进行比较,找出差距最大的一维,即影响解 $C_{bestnow}$ 和 $C_{worstnow}$ 产生差异贡献最大的一维;接着在所选出的维数上对目前最优解 $C_{bestnow}$ 进行更新.这种方法称为定位更新技术(location update technology,LUT).具体的算法流程见算法2.

算法2　定位更新技术

Step 1　在待工蜂搜索结束后,记录当前的最优解 $C_{bestnow}$ 和最差解 $C_{worstnow}$:

$$C_{bestnow}=\left\{C_k\middle|Perf(X,C_k)=\min(Perf(X,C_i)),i=1,2,\cdots,\frac{1}{2}N_s\right\},\tag{9}$$

$$C_{worstnow}=\left\{C_k\middle|Perf(X,C_k)=\max(Perf(X,C_i)),i=1,2,\cdots,\frac{1}{2}N_s\right\};\tag{10}$$

Step 2　计算解 $C_{bestnow}$ 和解 $C_{worstnow}$ 在每一维上元素的离差平方:

$$dis^{ij}=(C^{ij}_{bestnow}-C^{ij}_{worstnow}),\tag{11}$$

其中 $i\in\{1,2,\cdots,K\},j\in\{1,2,\cdots,D\},dis\in R^{M\times D}$;

Step 3　找出 dis 中元素最大值所在的位置,即当前最优解和最差解相对差距最大的位置.然后对当前的最优解在这一维上进行更新:

$$C^{ij}_{bestnow}=C^{ij}_{bestnow}+\varphi^j_i(C^{ij}_{bestnow}-C^{ij}_k),\tag{12}$$

其中 $k\in\left\{1,2,\cdots,\frac{1}{2}N_s\right\}$ 是随机产生的,且 $k\neq i$, 另外 ij 是 dis 中元素最大值所在的位置, φ^j_i 是[0,1]之间的随机数.

下面给出基于定位更新技术的人工蜂群(简记为LUT-ABC)聚类算法及其流程图(图1).

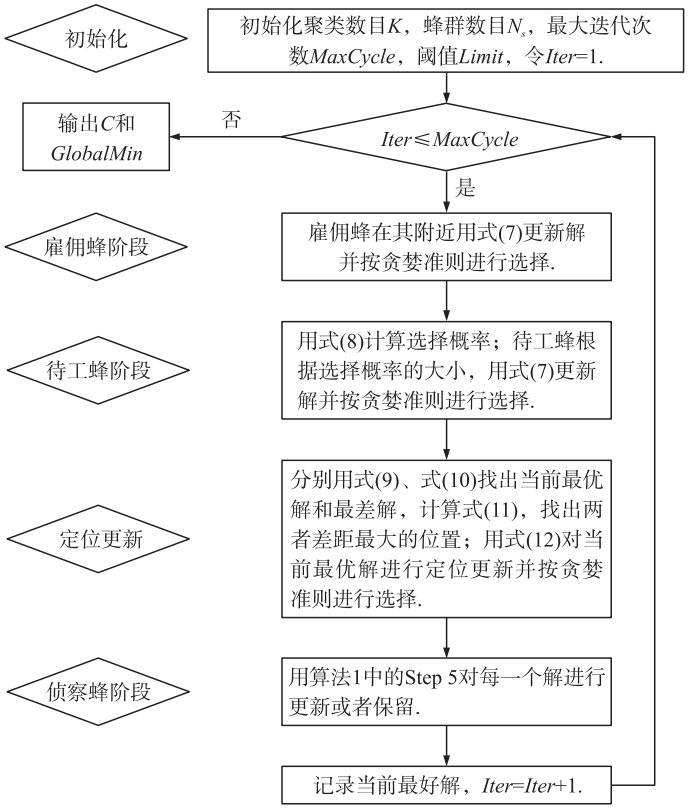

图1　LUT-ABC聚类算法的流程图

Fig.1　The flow chart of LUT-ABC clustering algorithm

算法3　基于定位更新技术的人工蜂群聚类算法

Step 1　初始化聚类数目 M,蜂群种群数目 N ,最大循环次数 $MaxCycle$,阈值 $Limit$,令 $Iter=1$;

Step 2　雇佣蜂阶段:雇佣蜂在其附近用式(7)更新解并按贪婪准则进行选择;

Step 3 待工蜂阶段:用式(8)计算选择概率;待工蜂根据选择概率的大小,用式(7)更新解并按贪婪准则进行选择;

Step 4 定位更新阶段:分别用公式(9)、(10)找出当前最优解和最差解,计算式(11),找出两者差距最大的位置;用式(12)对当前最优解进行定位更新并按贪婪准则进行选择;

Step 5 侦察蜂阶段:用算法1中的Step5对每一个解进行更新或者保留;

Step 6 $Iter = Iter + 1$;

Step 7 重复Step 2 ~ 6,当 $Iter > MaxCycle$ 时,输出 C 和 $GlobalMin$.

4 实验分析

4.1 测试数据集和参数设置

为验证LUT-ABC聚类算法的有效性,本文比较K-means算法、基于PSO的聚类算法[13]、基于ABC的聚类算法和LUT-ABC算法在7个典型数据集上的聚类性能. 这7个数据集分别是Artificial 1、Artificial 2、Iris、Wine、Cancer、CMC、Glass. 其中Artificial 1、Artificial 2为两组人工生成数据集,其余5组均选自UCI标准数据库[7]. 它们的组成和聚类特点见表2.

表2 测试数据集的组成和聚类特点

Table 2 The component and clustering features of the test data sets

数据集	维数d(特征数目)	数据个数	聚类数目M
Artificial 1	2	600	4
Artificial 2	3	250	5
Iris	4	150	3
Wine	13	178	3
Cancer	9	683	2
CMC	9	1 473	3
Glass	9	214	6

在表2中,Artificial 1数据集(如图2所示)包含600个样本数据,分为四个类,每一类均由150个服从正态分布的二维数据构成. 具体分布为:

$$N_2\left(\boldsymbol{\mu}=\begin{pmatrix} m_i \\ 0 \end{pmatrix},\ \boldsymbol{\Sigma}=\begin{bmatrix} 0.50 & 0.05 \\ 0.05 & 0.50 \end{bmatrix}\right),\ i=1,\cdots,4\ ,$$

其中 $\boldsymbol{\mu}$ 为均值向量,$m_1=-3, m_2=0, m_3=3, m_4=6$,$\boldsymbol{\Sigma}$ 为协方差矩阵.

Artificial 2数据集(如图3所示)包含250个样本数据,分为五个类,每一类均由50个服从均匀分布的三维数据构成. 具体分布为:$uniform(85,100)$,$uniform(70,85)$,$uniform(55,70)$,$uniform(40,55)$以及$uniform(25,40)$.

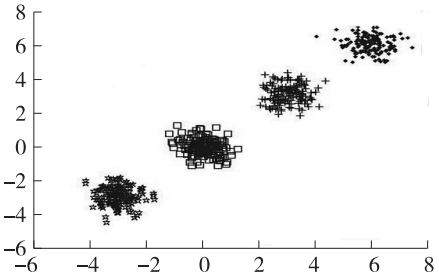

图2 Artificial 1数据集

Fig.2 Artifical 1 data set

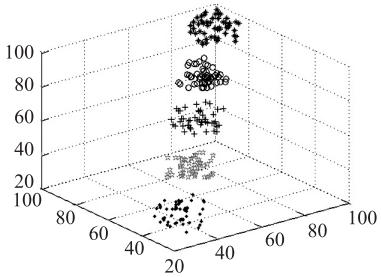

图3 Artificial 2 数据集

Fig.3 Artifical 2 data set

Iris数据集是通过提取3种不同类别花的花瓣和花萼等的特征所构成的数据集. 共有150个样本数据,每类各有50个样本,每个样本有4个特征.

Wine数据集是通过分析产于意大利同一地区不同种植园的3种葡萄酒所得的数据,包含178个样本数据,可以分成3类,每一个样本都代表一个产地的葡萄酒所包含的的13个化学特征.

Wisconsin Breast Cancer(Cancer)数据集共有683个样本,每个样本均包含丛厚度、平均细胞大小和平均细胞形状等9个特征. 该数据集分为两类,即恶性(444个样本)和良性(239个样本).

Contraceptive Mathod Choice(CMC)数据集是1987印尼国家避孕普及率调查集合的一个子集,包含1473个样本,每个样本各有9个特征.该数据被分为3类,其中没有使用避孕的占629个,长期使用避孕的占334个,短期使用避孕的占510个.

Glass数据集包含1214个样本数据,它取样于6种不同的类型的玻璃,其中建筑浮动处理窗户玻璃(70个样本),建筑非浮动处理窗户玻璃(76个样本),汽车浮动处理窗户玻璃(17个样本),容器玻璃(13个样本),餐具玻璃(9个样本),车头灯玻璃(29个样本).每个样本均有9个特征.

设蜂群种群数目 $N_s=40$,最大迭代次数 $MaxCycle=600$,阈值 $Limit=1\,000$.在以上7个数据集上的测试都是在相同的初始条件下运行的,计算机系统的配置如表3所示.

表3　计算机系统的配置

Table 3　Configuration of the computer

项目	配置
CPU	Inter®Core™2 P7350 @ 2.00GHz
内存	4.00GB DDR2 SDRAM
硬盘	320GB (5400RPM)
操作系统	Microsoft Windows XP

4.2　实验结果分析

采用类内距离之和与分类错误率(classification error rate,CER)两个标准,将LUT-ABC算法与K-means算法、基于PSO的聚类算法和基于ABC的聚类算法作比较.类内距离之和是指所有集群中的数据与所属集群中心的距离之和(式(1)).很明显,类内距离之和越小说明聚类的质量越高.分类错误率是指被分错类的数据数与测试集总数据数之比(式(13)).同样地,分类错误率越小说明聚类效果越好.

分别在上述7个数据集上对以上四种算法进行聚类性能分析.表4列出四种算法独立运行10次的类内距离之和的平均值、标准差和最好值,其中基于PSO的聚类算法的实验结果取自文献[13].对数据集Artificial 1,基于ABC的聚类算法和LUT-ABC算法所得的平均值和最好值明显比另外两种算法要小,尤其是LUT-ABC算法,而K-means算法、基于PSO以及ABC的聚类算法都会陷入局部最优.对数据集Artificial 2,因为数据有重叠,基于ABC的聚类算法和LUT-ABC算法所得的最好值虽然不如K-means算法和基于PSO的聚类算法,但LUT-ABC算法的平均值明显低于其他三种算法.对现实生活中的数据集,基于ABC的聚类算法和LUT-ABC算法在数据集Wine上的性能不如基于PSO的聚类算法,但在其余的4个数据集上,LUT-ABC算法均表现最好,其次是基于ABC的聚类算法.注意到除了数据集Cancer,LUT-ABC算法所得的标准差均是四种算法中最小的,这表明LUT-ABC算法比较稳定.而在数据集Cancer上,虽然

表4　7组数据的聚类结果

Table 4　Clustering results of 7 data sets

数据集	结果	K-means	基于PSO的聚类算法	基于ABC的聚类算法	LUT-ABC
Artificial 1	平均值	823.71	627.74	457.18	415.48
	标准差	218.10	180.24	186.33	143.77
	最好值	813.50	515.93	413.58	368.56
Artificial 2	平均值	2 335.55	2 217.20	2 084.40	1 828.20
	标准差	566.91	415.02	341.09	272.51
	最好值	1 763.17	1 743.20	1 761.40	1 760.50
Iris	平均值	106.05	103.51	99.44	96.92
	标准差	14.11	9.69	7.54	0.37
	最好值	97.33	96.66	97.59	96.55
Wine	平均值	18 061.00	16 311.00	16 567.00	16 504.00
	标准差	793.21	22.98	335.83	214.43
	最好值	16 555.58	16 294.00	16 300.00	16 295.00
Cancer	平均值	2 988.30	3 334.60	2 966.90	2 780.30
	标准差	0.46	357.66	32.67	25.51
	最好值	2 987.00	2 976.30	2 941.00	2 758.40
CMC	平均值	5693.60	5734.20	5564.90	5541.30
	标准差	473.14	289.00	56.81	8.99
	最好值	5 542.20	5 538.50	5 534.90	5 534.60
Glass	平均值	260.40	291.33	249.49	240.31
	标准差	36.82	12.33	13.11	9.80
	最好值	215.68	271.29	228.66	223.94

K-means算法所得的标准差比其他三种算法小得多,但是其平均值是最不理想的,这也说明K-means算法几乎每次都陷入局部最优.综上所述,LUT-ABC算法与其他三种算法相比寻优性能最好.为了更形象地观察出实验结果,图4和图5被用来描述K-means算法、基于ABC的聚类算法以及LUT-ABC算法在两个人工数据集上的收敛情况.

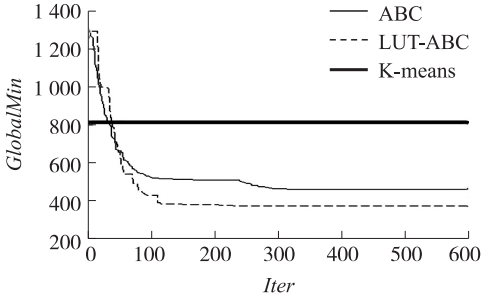

图4 Artificial 1数据集上的算法收敛图

Fig.4 Convergence chart of the algorithms for Artificial 1 data set

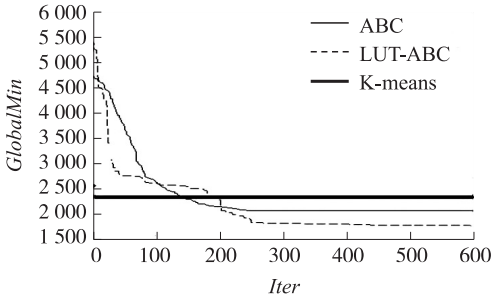

图5 Artificial 2数据集上的算法收敛图

Fig.5 Convergence chart of the algorithms for Artificial 2 data set

由图4和图5,可以发现K-means算法的初始值明显优于基于ABC的聚类算法和LUT-ABC算法,但是很快K-means就会陷入局部最优,即收敛曲线表现为水平直线.这也符合K-means算法的收敛特性:速度快但易陷入局部最优.另外,在两个人工数据集上,迭代开始时,两种人工蜂群算法表现差不多,但是随着迭代次数的增加,由于LUT-ABC算法采用定位更新技术,增强了算法的开采能力,跳出了局部最优.

下面用上述四个算法在测试数据集上的分类错误率对其进行直观的比较.这里使用著名的K-fold Cross Validation(简记为K-CV)方法.首先,将原始数据分成K组(一般是均分),依次将每组数据分别做一次测试集,其余的K-1组数据作为训练集.实际上,K-CV过程就是把实验重复做K次,每次试验都从K组数据中选择一组作为测试集,且每组只用一次,剩下的K-1组当作训练集进行实验,最后把K组实验所得到的分类错误率的平均值作为此K-CV方法下分类器的性能指标.这里采用最常用的10-CV方法.在每一次实验中,首先在训练集上运行算法算出相应的聚类中心位置;其次利用所得的聚类中心位置对测试集进行聚类,这里还是将每个数据划分到离它最近的聚类中心.最后,将算法输出的分类结果与原数据分类情况进行比较,得出被分错类的数据个数,并利用下式计算出相应的分类错误率:

$$CER = 100 \times \frac{\text{被分错类的数据个数}}{\text{测试集所含数据总个数}}. \tag{13}$$

这里以Artificial 2数据集为例,由于Artificial 2共有250个数据且被均分成5类.所以首先将Artificial 2数据集均分为10份,每份含有25个样本,需要注意的是,每一份都必须同等含有五个类的数据个数即含每类数据各5个.接着依次将10份中的每一份作为测试集,其余9份作为训练集来进行10次实验.类似可得其他数据集的训练集和测试集构成.

K-means算法、基于PSO的聚类算法、基于ABC的聚类算法以及LUT-ABC算法关于7个数据集运行10次得到的平均分类错误率在表5中给出,其中基于PSO的聚类算法的实验结果参见文献[13].对于数据Artificial 1,Artificial 2,Iris以及Cancer,基于ABC的聚类算法和LUT-ABC算法得到的平均分类错误率

表5 算法关于7个数据集的平均分类错误率

Fig.5 Average classification error rate of the algorithms of seven dta sets

数据集	算法			
	K-means	基于PSO的聚类算法[6]	基于ABC的聚类算法	LUT-ABC
Artificial 1	13.00	7.57	0.00	0.00
Artificial 2	34.00	22.00	0.00	0.00
Iris	17.80	12.53	9.00	9.00
Wine	31.12	28.71	22.30	21.00
Cancer	4.08	5.11	2.90	2.90
CMC	54.49	54.41	28.80	26.80
Glass	37.71	45.59	19.20	16.90

相同,明显优于K-means算法和基于PSO的聚类算法,特别是在两个人工数据集上,基于ABC的聚类算法和LUT-ABC算法的分类错误率均为0. 对数据集Wine、CMC以及Glass,LUT-ABC算法所得到的分类错误率最小,基于ABC的聚类算法其次,这表明LUT-ABC算法较其他三种方法可以更好地将测试数据集分类. 因此从分类错误率的角度来看,LUT-ABC算法的聚类效果更好.

5　结论

本文提出一种新的LUT-ABC算法并将其应用于数据聚类问题. 该算法有效结合了ABC算法和定位更新技术,即在每一次待工蜂搜索结束以后,充分利用当前最优解和最差解的信息,对最优解做进一步的更新. 实验结果表明,LUT-ABC算法较K-means算法、基于粒子群优化的聚类算法以及基于人工蜂群的聚类算法更适合进行聚类分析,通过利用先前的最优解和最差解寻找到可能更好的最优解来增强算法的开采能力,得到更好的聚类效果.

[参考文献]

[1] KRISHMA K, MURTY M N. Genetic K-means algorithm[J]. IEEE transactions on systems, man, and cybernetics. 1999, 29(3): 433-439.

[2] MAULIK U, BANDYOPADHAY S. Genetic algorithm-based clustering technique[J]. Pattern recognition, 2000, 33(9): 1455-1465.

[3] DENEUBOURG J L, GOSS S, FRANKS N. The dynamics of collective sorting: robot-like ants and ant-like robots[C]//Proceedings of the First International Conference on Simulation of Adaptive haviour, From Animals to Animals J, MIT Press, Cambridge MA, 1991: 356-365.

[4] LUMER E, FAIETA B. Diversity and adaptation in populations of clustering ants[C]//Proceedings of the Third International Conference on Simulation of Adaptive Behavior: From Animals to Animats, Vol. 3, MIT Press/Bradford Books, Cambridge, MA, 1994: 501-508.

[5] OMRAN M, Salman A, Engelbrecht A P. Image Classification Using Particle Swarm Optimization[C]//Proceedings of the 4th Asia-Pacific conference on simulated evolution and learning, Singapore, 2002.

[6] KARABOGA D. An idea based on honey bee swarm for numerical optimization[M]. Erciyes University, Engineering Faculty Computer Engineering Department, 2005.

[7] ZHANG C, OUYANG D, NING J. An artificial bee colony approach for clustering[J]. Expert Syst Appl, 2010, 37(7): 4761-4767.

[8] KARABOGA D, OZTURK C. Fuzzy clustering with artificial bee colony algorithm[J]. Sci Res Essay, 2010, 5(14): 1899-1902.

[9] KARABOGA D, OZTURK C. A novel clustering approach: Artificial Bee Colony algorithm[J]. Applied soft computing, 2011, 11(1): 652-657.

[10] TAN Q H, WU H J, HU B, et al. An Improved Artificial Bee Colony Algorithm for Clustering[C]//Proceedings of the 2014 conference companion on genetic and evolutionary computation companion, 2014: 19-20.

[11] OZTURK C, HANCER E, KARABOGA D. Improved clustering criterion for image clust-ering with artificial bee colony algorithm[J]. Pattern Anal Appl. 2014. http://dx.doi.org/10.1007/s10044-014-0365-y(in press).

[12] CELAL OZTURK, EMRAH HANCER, DERVIS KARABOGA. Dynamic clustering with improved binary artificial bee colony algorithm[J]. Applied soft computing, 2015(28): 69-80.

[13] KAO Y T, ZAHARA E, KAO I W. A hybridized approach to data clustering[J]. Expert systems with applications, 2008, 34(3): 1754-1762.

[责任编辑:陈　庆]