文章编号: 1001-0920(2018)01-0181-05 DOI: 10.13195/j.kzyjc.2016.1252

基于改进蜂群算法的K-means算法

于佐军†, 秦 欢

(中国石油大学(华东) 信息与控制工程学院, 山东 青岛 266580)

摘 要: 针对标准人工蜂群算法搜索效率低、收敛速度慢等缺点提出一种改进的人工蜂群算法. 通过引入算术交叉操作以及利用最优解指导搜索方向,增加算法收敛的速度. 在7个基准函数上的测试结果表明了算法的有效性. 在此基础上,针对K-means算法的缺点提出基于改进蜂群算法的K-means算法,并加入自动获得最佳聚类数的功能. 在人工数据集和UCI真实数据集上的测试验证了所提出算法的性能.

关键词: 人工蜂群算法; 聚类算法; 算术交叉; 最佳聚类数

中图分类号: TP301.6 **文献标志码:** A

K-means algorithm based on improved artificial bee colony algorithm

YU Zuo-jun†, QIN Huan

(College of Information and Control Engineering, China University of Petroleum(East China), Qingdao 266580, China)

Abstract: In order to overcome the disadvantage of the canonical artificial bee colony algorithm, which has low search efficiency and slow convergence, an improved artificial bee colony algorithm is proposed. This algorithm increases the convergence speed by introducing the arithmetic crossover operation and guiding the search direction by the global best solution. The proposed algorithm is proved to be effective with a test on seven benchmark functions. On the basis of previous work, according to the drawbacks of the K-means algorithm, the K-means algorithm based on the improved artificial bee colony algorithm is proposed, and the function of automatically selecting the best number of clusters is added. A test on the artificial data sets and UCI real data sets verifies the performance of the proposed algorithm.

Keywords: artificial bee algorithm; clustering algorithm; arithmetic crossover; best number of clusters

0 引 言

聚类作为分析数据的重要方法,被广泛用于数据挖掘、模式识别、图像分析等领域,其目的是将给定的数据集划分为若干类,使得同一类之间的数据相似性最大,不同类之间的数据相似性最小.

K-means算法是一种基于中心的聚类算法,结构简单、收敛速度快,但极易受初始聚类中心的影响,导致最终陷入局部最优解[1],且预先给定的聚类数对结果起着决定性作用. 然而,多数情况下很难预先获知最佳的聚类个数,因此算法本身能够获得最佳聚类数显得十分重要.

群智能算法是一种模拟生物群体行为的人工智能算法,常见算法有蚁群算法、粒子群算法、遗传算法. Karaboga于2005年提出人工蜂群算法[2],其概念简单、易于实现、控制参数少. 群智能算法由于具有强大的全局搜索能力而被广泛用于聚类领域,文献[3]提出了一种结合蚁群算法的聚类算法;文献[4]采用粒子群算法来解决聚类问题;文献[5]提出了结合人工蜂群算法的聚类算法.

鉴于人工蜂群算法和K-means聚类算法各自的优点,本文首先对人工蜂群算法作出改进,以提高算法的收敛速度;然后用其优化K-means算法中聚类中心的位置,消除初始聚类中心的影响和陷入局部最优解的可能,并在其中加入利用聚类准则函数获得最佳聚类数的环节.

1 聚类相关和基本人工蜂群算法

1.1 聚类相关

聚类问题可以描述为:给定一个样本数为n的数据集$Y=(y_1,y_2,\cdots,y_n)$,将其划分成k个类,即$C=(C_1,C_2,\cdots,C_n)$. 本文使用均方误差MES作为聚类的目标函数,定义为

$$\mathrm{MSE}=\frac{1}{n}\sum_{j=1}^{k}\sum_{y_i\in C_j}\|y_i-z_j\|^2, \tag{1}$$

收稿日期: 2016-10-05; **修回日期:** 2017-03-09.

作者简介: 于佐军(1961–), 男, 副教授, 从事工业过程建模、控制与优化技术等研究; 秦欢(1992–), 男, 硕士生, 从事工业过程建模、控制与优化技术的研究.

†通讯作者. E-mail: yuzj@upc.edu.cn

其中z_j表示聚类中心. MSE的值越小,相应的聚类效果越好.

1.1.1 K-means算法

K-means算法的具体步骤如下.

Step 1: 随机选取k个样本作为初始聚类中心;

Step 2: 计算每个样本与各个聚类中心的欧氏距离,将样本划分给最近的聚类中心;

Step 3: 利用下式更新聚类中心后转Step 2:

$$z_j = \sum_{y_i \in C_j} y_i / n_j, \tag{2}$$

其中n_j表示第j个类中数据的个数.

1.1.2 聚类评价函数

本文使用基于聚类分布的有效性度量方法[6]来构造聚类评价函数,通过此函数评价不同聚类数下的聚类效果,找出其中最佳的聚类数.

对于给定的数据集Y,类内方差定义如下:

$$\mathrm{var}(Y) = \sqrt{\sum_{i=1}^{n} d^2(y_i, \bar{y})/n}. \tag{3}$$

其中: $\bar{y} = \sum_{i=1}^{n} y_i/n$, $d(\,)$表示欧氏距离.

聚类密集性定义如下:

$$C_{\mathrm{var}} = \sum_{i=1}^{k} (\mathrm{var}(C_i)/\mathrm{var}(Y))/k. \tag{4}$$

聚类邻近性定义如下:

$$P_r = \frac{1}{k(k-1)} \sum_{i=1}^{k} \sum_{j \neq i, j=1}^{k} \exp\Big[-\frac{d^2(C_i, C_j)}{2\sigma^2}\Big], \tag{5}$$

其中$d(C_i, C_j)$表示类C_i与C_j中心点的距离.

最终构造出聚类有效性指标

$$P_C = 1 - [\lambda \times C_{\mathrm{var}} + (1-\lambda) \times P_r], \tag{6}$$

其中$\lambda \in [0,1]$. P_C值越大表示聚类效果越好.

1.2 基本人工蜂群算法

人工蜂群算法是一种模仿蜜蜂觅食行为的群智能算法,算法中的蜂群分为3个部分: 引领蜂、跟随蜂和侦查蜂[7]. 引领蜂对应特定的食物源,并携带与食物源有关的具体信息;跟随蜂在蜂巢的舞蹈区等待引领蜂分享关于食物源的相关信息,并据此选定某个食物源在其周边进一步探索;侦查蜂负责随机地搜索新的食物源.

基本人工蜂群算法主要分为以下4个阶段.

1) 初始化阶段.

在可行解空间内随机产生SN个食物源,每个食物源表示一个可行解,具体公式如下:

$$x_{i,j} = x_j^{\min} + \mathrm{rand}(0,1) \times (x_j^{\max} - x_j^{\min}). \tag{7}$$

其中: $i = 1, 2, \cdots, \mathrm{SN}$; $j = 1, 2, \cdots D$, D表示可行解的维数; $x_j^{\max}$和$x_j^{\min}$表示第j个参数的上限和下限. 此外,对每个食物源设置一个计数器,将其中的数值置0.

2) 引领蜂搜索阶段.

引领蜂在对应的食物源附近展开邻域搜索,寻找新的食物源v_i,搜索公式如下:

$$v_{i,j} = x_{i,j} + (-1 + 2 \times \mathrm{rand})(x_{i,j} - x_{k,j}). \tag{8}$$

其中: k表示随机选取的与i不同的食物源, $k \neq i$. 对于新、旧两个食物源x_i和v_i,采用“贪婪选择”算法,即对新、旧食物源的质量进行比较,如果新的食物源的质量好,则保留新的食物源,令其计数器置0;否则,保留旧的食物源,令其计数器中的数字加1.

3) 跟随蜂搜索阶段.

跟随蜂按轮盘赌的方式从当前食物源中选中一个食物源. 每个食物源被选择的概率如下:

$$P_i = \mathrm{fit}_i \Big/ \sum_{j=1}^{\mathrm{SN}} \mathrm{fit}_j. \tag{9}$$

其中: fit_i表示食物源的质量,计算公式如下:

$$\mathrm{fit}_i = \begin{cases} 1/(1+f_i), & f_i \geqslant 0; \\ 1 + \mathrm{abs}(f_i), & \mathrm{otherwise}. \end{cases} \tag{10}$$

其中f_i代表目标函数的值.

跟随蜂选定食物源之后,按照式(8)的方式进行邻域搜索,然后进行贪婪选择的相关操作.

4) 侦查蜂搜索阶段.

为了避免在进化过程中丧失种群多样性,蜂群算法中加入了特有的侦查蜂搜索模式. 当某个食物源所对应的计数器中的数值大于某个预先设置的阀值limit时,可以认为当前食物源已经耗尽,放弃此食物源,相应的引领蜂变为侦查蜂,采用式(7)在可行解空间内随机产生新的食物源.

2 改进的人工蜂群算法

基本人工蜂群算法存在如下两个缺点: 1) 邻域搜索是一种单维搜索模式,收敛速度慢; 2) 邻域搜索是一种随机的搜索方式,缺乏方向性并且效率较低. 这些缺点引起了众多学者的关注,各种新算法也相继被提出. Yan等[8]提出了结合遗传算法独有的交叉操作的混合人工蜂群(HABC)算法; Zhu等[9]提出了结合粒子群算法优点的GABC算法; Zou等[10]提出了利用虚拟蜜蜂获取各个维数上最优解的协同人工蜂群(CABC)算法; Cong等[11]受差分算法中突变

操作与HABC算法的交叉操作的启迪,提出了EABC算法.

本文结合遗传算法中的算术交叉操作与粒子群算法中最优解指导搜索方向的思想,对蜂群的搜索方式作出改进. 针对邻域搜索收敛速度慢的缺点,引入算术交叉操作,具体形式为

$$\text{child} = \text{rand} \times \text{parent1} + \text{rand} \times \text{parent2}. \quad (11)$$

其中: child表示新的解, parent表示已存在的解. 这种对所有维数同时进行操作的搜索方式收敛速度更快. 针对邻域搜索缺乏方向性的缺点,利用最优解来指导搜索方向,使蜜蜂沿着蜜源质量升高的方向进行搜索,增加寻找到全局最优解的概率.

2.1　改进的引领蜂搜索方式

改进的引领蜂搜索方式有两种. 在算法的早期阶段,搜索方式如下: 首先判断当前所选择的引领蜂所代表的食物源是否为最优,是则采用下式:

$$v_{i,j} = x_{i,j} + (-1 + 2 \times \text{rand})(x_{i,j} - x_{k,j}); \quad (12)$$

不是则采用下式:

$$v_{i,j} = x_{i,j} + \text{rand} \times (x_{\text{better},j} - x_{i,j}) + \text{rand} \times (G_{\text{best},j} - x_{i,j}). \quad (13)$$

其中: x_{better}表示比当前食物源x_i的质量高的任一食物源; G_{best}表示当前质量最高的食物源,即当前全局最优解. 在算法的后期阶段,搜索方式如下: 首先判断当前所选择的引领蜂所代表的食物源是否为最优,是则采用下式:

$$v_i = x_i + \text{rand} \times (x_i - x_k); \quad (14)$$

不是则采用下式:

$$v_i = \text{rand} \times x_i + \text{rand} \times x_{\text{better}}. \quad (15)$$

从上面的公式可以看出,对代表当前最优解的蜜蜂采取随机搜索方式,增加算法跳出局部最优的能力,对其余蜜蜂则利用全局最优解和相对最优解引导搜索方向,提高搜索效率.

在算法的不同阶段采取不同搜索策略的原因为: 早期阶段产生的解的随机性比较大,与全局最优解相差较大,此时使用全维搜索方式(即式(14)、(15))的弊端在于会忽略单个维数的变化趋势,使得算法无法收敛到全局最优解; 后期阶段产生的解接近全解最优解,使用全维搜索方式有助加快算法收敛,且不必担心会忽略单维的变化趋势.

2.2　改进的跟随蜂搜索方式

跟随蜂的作用主要是对引领蜂代表的蜜源进行更进一步的精细搜索,因此只采用式(12)、(13)所描述的单维交互信息的搜索方式.

2.3　改进的侦查蜂搜索方式

在侦查蜂搜索的基本模式中加入额外的判断条件,即放弃某个食物源之前进行一次判断,如果其是当前的最优解,则不放弃该食物源,否则放弃此食物源并将相应引领蜂变为侦查蜂.

3　基于改进蜂群算法的K-means算法

基本思想: 按照K-means算法的聚类原则,利用改进蜂群算法优化聚类中心,得到不同聚类数下的聚类划分,然后利用聚类评价函数获得最佳聚类数.

利用改进蜂群算法优化聚类中心,是由于蜂群算法是一种随机寻优算法,不受初始解的干扰,且具备优越的跳出局部解的性能. 这些优点保证其能够在整个搜索空间内得到使聚类目标函数尽可能小的聚类中心,虽然其随机性使得结果不一定为最优,但是与最优解之间的误差极小. 对整个数据集进行聚类划分时,是按照样本与聚类中心的欧氏距离最近的原则划分的,聚类中心位置的细微变动不会对划分结果产生显著影响,因此可以认为用改进蜂群算法优化聚类中心能够得到最佳的聚类结果. 在此基础上,利用聚类评价函数可得到相应结构标准下的最佳聚类数.

在算法的实现中有两个部分需要注意.

1) 初始化阶段.

确定聚类数的搜索范围并进行食物源初始化操作. 聚类数的搜索范围为[2,$C_{\max}$],其中$C_{\max} \leqslant \sqrt{n}$. 食物源的形式为$x_i = (z_1, z_2, \cdots, z_k)$,表示对数据集的一种划分,其中$z_j$为聚类中心.

2) 目标函数计算阶段.

计算目标函数之前,首先按照样本与当前聚类中心之间的欧氏距离最小原则将所有样本重新划分类别,然后利用式(1)计算目标函数.

算法的具体步骤如下.

Step 1: 确定聚类数的搜索范围,令初始聚类数$k = 2$;

Step 2: 按照当前聚类数完成食物源初始化,计算食物源的目标函数并选出当前最优解G_{best};

Step 3: 循环执行改进的人工蜂群算法的3个搜索阶段,循环结束后输出最优的聚类结果,并据此计算聚类有效性指标P_C;

Step 4: 令$k = k+1$,如果$k < C_{\max}$,则转到Step 2,否则转到Step 5;

Step 5: 根据聚类有效性指标P_C找出最佳的聚类数,并输出相应的聚类结果.

4 实验仿真与分析

算法的性能验证主要包含两个部分: 1) 改进蜂群算法寻找最优解的性能; 2) 基于改进蜂群算法的K-means算法寻找最优聚类数的性能.

4.1 改进蜂群算法性能分析

将本文提出的改进蜂群算法与基本人工蜂群算法(ABC)、协同人工蜂群算法(CABC)、混合人工蜂群算法(HABC)、遗传算法(GA)、粒子群算法(PSO)相比较,利用一系列的基准函数作为测试算法性能的基准,基准函数详细表达见文献[12].

4.1.1 相关算法参数设置

实验中,所有变量维数设为30,种群大小设为100, limit $=$ 100,其余参数与文献[12]一致. 使用函数运行次数FEs(function evaluations)作为测量基准,算法的终止条件是FEs $>$ 100 000.

4.1.2 仿真结果及分析

表1展示了每种算法运行30次之后获得的基准函数值的平均值和标准差(除ABC算法和本文算法结果外,其余算法结果均出自文献[12]). 从表1可以看出,在7个基准函数中,HABC算法在4个基准函数上性能最佳,本文提出的算法在3个基准函数上性能最佳,其余算法总体性能稍差. 将HABC算法与本文算法性能进行单独比较,本文算法在5个基准函数上的性能均优于HABC算法.

值得注意的一点是,HABC算法在基准函数f_2上最终会困于局部最优解处,连续运行30次,均未能跳出局部最优解. 原因可能为:对全维进行操作可能会忽略某些单维方面的变化趋势,导致算法无法收敛到最优解. 在此基准函数上,本文算法虽然在多次运行期间也出现过陷于局部最优的情况,但整体而言仍然可以找到全局最优解.

表1 本文方法与ABC、CABC、HABC、GA、PSO算法效果比较

函数		ABC	CABC	HABC	GA	PSO	本文算法
Sphere(f_1)	均值	9.194e-15	3.122e-13	0	1.390e+00	1.035e-05	1.001e-18
	标准差	5.177e-15	3.343e-13	0	50 156e-01	7.015e-06	1.001e-18
	排名	3	4	1	6	5	2
Rosenbrock(f_2)	均值	3.363e-01	6.205e+00	2.812e+01	1.440e+03	3.405e+01	3.023e+00
	标准差	3.676e-01	1.432e+01	5.434e-01	7.813e+02	2.111e+01	6.736e+00
	排名	1	3	4	6	5	2
Quadric(f_3)	均值	1.082e+02	8.578e+01	0	2.058e+02	4.030e-02	9.622e-19
	标准差	1.847e+01	2.011e+01	0	5.777e+01	1.961e-02	1.246e-18
	排名	5	4	1	6	3	2
Rastrigin(f_4)	均值	7.060e-02	2.843e-07	0	1.380e+02	5.998e+01	0
	标准差	2.526e-01	6.281e-07	0	3.247e+01	1.651e+01	0
	排名	3	2	1	5	4	1
Schwefel(f_5)	均值	1.942e+02	3.848e-04	7.916e+02	1.023e+03	6.005e+03	4.738e+01
	标准差	1.235e+02	6.918e-06	3.549e+02	4.212e+02	8.005e+02	6.671e+01
	排名	3	1	4	5	6	2
Ackley(f_6)	均值	7.458e-06	2.618e-05	8.882e-16	1.919e+01	2.116e+00	8.882e-16
	标准差	3.253e-06	1.192e-05	2.006e-31	8.464e-01	4.708e-01	0
	排名	3	4	2	6	5	1
Griewank(f_7)	均值	2.144e-08	4.138e-04	0	5.060e+00	3.138e-02	0
	标准差	8.414e-08	1.842e-03	0	1.634e+00	1.455e-01	0
	排名	2	3	1	5	4	1

整体而言,对于文中采用的7个基准函数,本文提出的算法性能最优.

4.2 算法寻找最佳聚类数性能分析

为了测试本文算法的有效性,利用人工和真实数据集验证算法寻找最优聚类数的效果.

4.2.1 测试数据集

测试数据集包括两个服从高斯分布的人造数据集S1、S2(见图1和图2)和来自UCI机器学习数据库的数据集Iris, glass, cmc,数据集特征如表2所示.

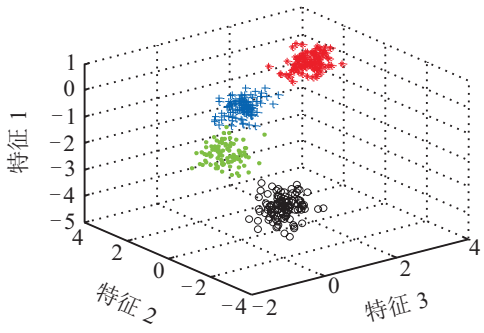

图1 服从高斯分布的人造数据集S1

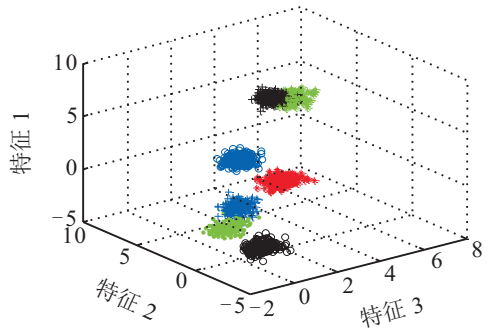

图 2　服从高斯分布的人造数据集 S2

表 2　测试数据集特征描述

数据集	大小	类别	维数	单类对象数目
S1	400	4	3	100
S2	1 400	7	3	200
Iris	150	3	4	50
Glass	214	6	9	70, 76, 17, 13, 9, 29
Cmc	1 473	3	10	629, 333, 511

4.2.2　实验结果与分析

检验本文算法聚类准确率的性能时, 预先给定每个数据集正确的聚类数; 检验本文算法获取最优聚类数的性能时, 重复运行算法 10 次, 记录算法得到的正确最优聚类数的次数, 结果如表 3 所示.

表 3　算法运行 10 次结果

数据集	聚类准确率 / %	K 值正确次数
S1	100	10
S2	100	10
Iris	90	9
Glass	51.8	5
Cmc	39.5	0

由上述结果可以看出: 在聚类准确率方面, 本文提出的算法对于分离性良好的球状人工数据簇 S1、S2 能够实现正确的聚类, 而对于呈带状簇的真实数据集的聚类效果较差; 在寻找最佳聚类数方面, 本文提出的算法对于分离性良好的球状簇能够找到正确的聚类数, 对非球状簇的真实数据集难以找到理论上的最优聚类数. 总而言之, 本文算法的泛化能力较差, 适用于球状簇组成的数据集的聚类.

5　结　论

本文首先提出了一种改进的蜂群算法, 分别从引领蜂和跟随蜂的搜索模式方面进行了改进, 提高了算法收敛的速度, 增加了算法跳出局部最优解的能力. 然而, 实验结果表明, 该算法在求解某些函数时仍有陷于局部最优的情况, 如何在保证收敛速度的情况下增加跳出局部最优的能力有待进一步研究. 随后, 结合改进蜂群算法和 K-means 聚类, 提出了一种能自动寻找最优聚类数的聚类算法, 并利用相应数据集进行了聚类测试, 验证了算法的可行性.

参考文献 (References)

[1] 曹永春, 蔡正琦, 邵亚斌. 基于 K-means 的改进人工蜂群聚类算法 [J]. 计算机应用, 2014, 34(1): 204-207.
(Cao Y C, Cai Z Q, Shao Y B, Improved artificial bee colony clustering algorithm based on K-means[J]. J of Computer Applications. 2014, 34(1): 204-207.)

[2] Karaboga D. An idea based on honey bee swarm for numerical optimization[R]. Engineering Faculty, Computer Engineering Department, Erciyes University, 2005.

[3] Shelokar P S, Jayaraman V K, Kulkarni B D. An ant colony approach for clustering[J]. Analytica Chimica Acta, 2004, 509(2): 187-195.

[4] Van der Merwe D W, Engelbrecht A P. Data clustering using particle swarm optimization[C]. The 2003 Congress on Evolutionary. Canberra: IEEE, 2003, 1: 215-220.

[5] Karaboga D, Ozturk C. A novel clustering approach: Artificial bee colony(ABC) algorithm[J]. Applied Soft Computing, 2011, 11(1): 652-657.

[6] 杨燕, 靳蕃, Mohamed K. 聚类有效性评价综述 [J]. 计算机应用研究, 2008, 25(6): 1630-1632.
(Yang Y, Jin F, Mohamed K. Survey of clustering validity evaluation[J]. Application Research of Computers, 2008, 25(6): 1630-1632.)

[7] Karaboga D, Basturk B. A powerful and efficient algorithm for numerical function optimization: Artificial bee colony(ABC) algorithm[J]. J of Global Optimization, 2007, 39(3): 459-471.

[8] Yan X, Zhu Y, Zou W, et al. A new approach for data clustering using hybrid artificial bee colony algorithm[J]. Neurocomputing, 2012, 97: 241-250.

[9] Zhu G, Kwong S. Gbest-guided artificial bee colony algorithm for numerical function optimization[J]. Applied Mathematics and Computation, 2010, 217(7): 3166-3173.

[10] Zou W, Zhu Y, Chen H, et al. A clustering approach using cooperative artificial bee colony algorithm[J]. Discrete Dynamics in Nature and Society, 2010, 2010: 1-16.

[11] Tran D C, Wu Z, Wang Z, et al. A novel hybrid data clustering algorithm based on artificial bee colony algorithm and K-means[J]. Chinese J of Electronics, 2015, 24(4): 694-701.

[12] Yan X, Zhu Y, Zou W, et al. A new approach for data clustering using hybrid artificial bee colony algorithm[J]. Neurocomputing, 2012, 97: 241-250.

（责任编辑：闫　妍）