计算机科学
COMPUTER SCIENCE

基于人工蜂群的三支*k*-means聚类算法

徐天杰，王平心，杨习贝

引用本文

徐天杰，王平心，杨习贝.基于人工蜂群的三支*k*-means聚类算法[J]．计算机科学, 2023, 50(6): 116-121.

XU Tianjie, WANG Pingxin, YANG Xibei. Three-way *k*-means Clustering Based on Artificial Bee Colony [J]. Computer Science, 2023, 50(6): 116-121.

计算机科学 COMPUTER SCIENCE
http://www.jsjkx.com
DOI:10.11896/jsjkx.220800150

基于人工蜂群的三支 *k*-means 聚类算法

徐天杰[1]　王平心[2]　杨习贝[1]
1 江苏科技大学计算机学院　江苏 镇江 212003
2 江苏科技大学理学院 江苏 镇江 212003
(tianjie_xu@163.com)

摘　要　聚类在数据挖掘技术中起着至关重要的作用。传统的聚类算法都是硬聚类算法,即对象要么属于一个类,要么不属于一个类,在处理不确定数据时,强制划分会带来决策错误。三支 *k*-means 聚类算法可以对边界不确定数据进行更加合理的分类,但仍然存在对初始聚类中心敏感的问题。为解决这一问题,将人工蜂群算法与三支 *k*-means 聚类算法相结合,提出了一种基于人工蜂群的三支 *k*-means 聚类算法。通过定义类内聚集度函数和类间离散度函数来构造蜜源的适应度函数,引导蜂群向高质量的蜜源进行全局搜索。利用蜂群之间不同角色的相互协作与互换,对数据集进行多次迭代聚类,找到最优的蜜源位置,作为初始聚类中心,并在此基础上交替迭代聚类。实验证明,该方法对聚类结果的性能指标有所提高。在 UCI 数据集上的实验验证了该算法的有效性。

关键词:三支 *k*-means 聚类算法;人工蜂群算法;适应度函数;初始聚类中心;蜜源
中图法分类号　TP391

Three-way *k*-means Clustering Based on Artificial Bee Colony

XU Tianjie[1], WANG Pingxin[2] and YANG Xibei[1]
1 School of Computer, Jiangsu University of Science and Technology, Zhenjiang, Jiangsu 212003, China
2 School of Science, Jiangsu University of Science and Technology, Zhenjiang, Jiangsu 212003, China

Abstract　Clustering plays an important role in data mining technology. Traditional clustering algorithms are hard clustering algorithms, namely, objects either belong to a class or do not belong to a class. However, when dealing with uncertain data, forced division will lead to decision-making errors. Three-way *k*-means clustering algorithm can divide the data into several groups with uncertain boundary reasonably. But it is still sensitive to the initial clustering center. In order to solve this problem, this paper presents a three-way *k*-means clustering algorithm based on artificial bee colony by integrating artificial bee colony algorithm with three-way *k*-means clustering algorithm. The fitness function of honey source is constructed by class cohesion function and inter class dispersion function to guide the bee colony to search for high-quality honey source globally. Using the cooperation and exchange of different roles between bee colonies, the data set is clustered repeatedly to find the optimal honey source location, which is used as the initial clustering center, and on this basis, iterative clustering is carried out alternately. Experiments show that this method improves the performance index of clustering results. The effectiveness of the algorithm is verified on UCI data set.

Keywords　Three-way *k*-means, Artificial bee colony algorithm, Fitness function, Initial cluster center, Nectar

1　引言

聚类算法的目的是将数据集划分成多个类簇,使得在同一个类簇中的对象尽可能相似,不同类簇中的对象不相似。聚类算法在许多领域应用广泛,如生物信息学[1]、安全保障[2]、图像处理[3]等。聚类又被称为无监督学习,其与监督学习的不同之处在于,在类簇中没有可用于表示数据类别的分类和分组的信息,也就是说,在对一个数据集进行聚类的过程中,我们无须知道样本的标签,仅仅利用一些聚类的算法就可以对样本进行分类。目前在大部分的聚类算法中,算法的选择取决于聚类的目的、数据的类型和具体应用。为了将一些抽象的问题变成可行解,研究人员提出了基于划分的聚类方法、基于层次的聚类方法、基于密度的聚类方法、基于网格的聚类方法和基于模型的聚类方法[4]。

k-means 是一种基于划分的聚类算法,其对象和各类簇之间只存在两种关系:属于该类簇和不属于该类簇。如果样本在该类簇中,则属于该类簇,否则不属于该类簇。*k*-means 因简单、高效、收敛速度快、易于实现而被广泛应用,但也存在

到稿日期:2022-08-15　返修日期:2022-11-25
基金项目:国家自然科学基金(62076111,61773012);江苏省高校自然科学基金(15KJB110004)
This work was supported by the National Natural Science Foundation of China(62076111,61773012) and Natural Science Fund for Colleges and Universities in Jiangsu Province(15KJB110004).
通信作者:王平心(pingxin_wang@hotmail.com)

需随机地选取聚类中心、类簇需要人为给定、对噪声点和离群点敏感、很难发现非凸现状的簇、易陷入局部最优[5-6]等问题。另外，k-means 聚类是一种硬聚类，类簇之间有清晰的边界，但在处理不确定数据时，如果强制将某个对象划分到类簇中，会带来较高的决策风险，降低聚类精度。为了解决这一问题，Wang 等[7]将三支决策[8-9]和 k-means 相结合，提出了三支 k-means 算法。三支 k-means 算法的主要思想是在 k-means 迭代过程中引入容许误差，将每个类簇的结果用核心域和边界域来表示，较好地解决了不确定性数据的处理问题。然而，三支 k-means 算法和 k-means 算法一样，存在聚类结果和随机选取的聚类中心有关、对噪声点和离群点敏感、易陷入局部最优的问题。

为了解决上述问题，本文将人工蜂群算法[10]引入到三支 k-means 算法中，利用人工蜂群算法更新迭代寻找聚类中心。人工蜂群算法(Artificial Bee Colony Algorithm，ABC)是一种群智能算法[10]，是由 Karaboga 于 2009 年提出的一种模拟蜜蜂群体寻找最优蜜源的仿生智能算法。其特点是控制参数少、简单、易于实现、有较强的全局寻优能力，因而被广泛应用到聚类领域中。本文通过定义类内聚集度函数和类间离散度函数来构造蜜源的适应度函数，引导蜂群向高质量的蜜源进行全局搜索。利用蜂群之间不同角色的相互协作与互换对数据集进行多次迭代聚类，找到最优的蜜源位置，作为初始聚类中心，并在此基础上进行迭代更新。采用三支决策规则，在处理类内孤立数据和类与类的重叠数据时，有效地提高了聚类结果的准确性，降低了决策风险。将人工蜂群算法[10]引入到三支 k-means 算法中可以较好地解决三支 k-means 算法对初始聚类中心敏感的问题，提高算法的准确性。

本文的其余部分安排如下：第 2 节介绍了三支聚类的基本概念和三支 k-means 算法的相关内容；第 3 节介绍人工蜂群算法并在此基础上提出了基于人工蜂群的三支 k-means 聚类算法；第 4 节介绍了本文实验分析采用的聚类评价指标；第 5 节对算法进行实验分析，在 13 个 UCI 数据集上对本文提出的算法进行了有效性验证；最后总结全文。

2 相关工作

2.1 三支聚类

三支决策[8-9]是姚一豫教授在决策粗糙集[11]和概率粗糙集[12]的研究基础上为解决不确定性问题提出的理论，其核心思想是将待决策项拓展为正域决策、负域决策和边界域决策。对有充分把握、信息全面的事物，直接给出接受或者拒绝的判断，对信息不充分的事物，做延迟决策。假设 U 是一个非空、有限实体集，A 表示一个有限条件集，三支决策将 U 划分成 3 个互不相交的域，这 3 个域分别是 POS(正域)、NEG(负域)、BND(边界域)。根据这 3 个域给出了三支决策的规则，正域生成正规则，对对象做出接收决策；负域生成负规则，对对象做出拒绝决策；边界域生成边界规则，对对象做出延迟决策。

近年来，在不确定性信息处理方面，三支决策理论得到了广泛应用和推广。Yu 等[13]将三支决策的理论应用到了聚类中，提出了三支聚类理论。基于这一理论，Wang 等[14]提出了基于动态邻域的三支聚类方法；Wang 等[15]将数学形态的侵蚀和膨胀思想引入聚类中，提出了 CE3 框架。

传统的聚类方法大多是二支决策，即决策一个元素属于一个类或者不属于一个类。然而在处理不确定性信息时，强制将其中的元素划分到一个类中，往往容易带来较高的决策风险。与二支聚类不同，三支聚类[16]使用 $Co(C)$，$Fr(C)$，$Tr(C)$3 个集合，即核心域、边界域和琐碎域来表示一个类簇，其中 $Co(C)$，$Fr(C)$，$Tr(C)$满足 $Tr(C)\cup Co(C)\cup Fr(C)=U$。每个类簇将数据集分成核心域、边界域和琐碎域 3 部分，其中核心域中的元素确定属于类 C，边界域中的元素可能属于类 C，琐碎域中的元素确定不属于类 C，这样可以有效地降低决策风险。假设 $U=\{x_1,x_2,\cdots,x_n\}$是一个非空、有限集，n 表示数据集中的 n 个对象，三支决策聚类的结果可以表示为：

$$T=\{(Co(c_1),Fr(c_1)),(Co(c_2),Fr(c_2)),\cdots,(Co(c_k),Fr(c_k))\}$$

其中，$Co(c_i)(i=1,2,\cdots,k)$表示第 i 类的核心域，$Fr(c_i)(i=1,2,\cdots,k)$表示第 i 类的边界域。核心域和边界域满足以下 3 个条件：

(1)$Co(C_i)\neq\varnothing$；

(2)$\bigcup_{i=1}^{k}(Co(C_i)\cup Fr(C_i))=U$；

(3)$Co(C_i)\cap Co(C_j)=\varnothing,i\neq j$。

条件(1)要求任意类簇不能为空。条件(2)要求数据集 U 中的任意样本对象至少属于某个类簇中的核心域或者边界域，可能存在某个样本元素 $x\in U$ 属于多个类簇的情况。条件(3)要求不同类簇之间的核心域是没有交集的。如果 $Fr(C_i)=\varnothing$，则类簇 $C_i=Co(C_i)$且 $Tr(C_i)=U-Co(C_i)$，此时该聚类结果是一个二支聚类结果。因此，三支决策聚类形式是传统二支聚类方法的推广，这也是对一些不确定数据聚类问题提出的一种解决方案。针对根据目前已知信息难以聚类的对象，我们无法确定其所属类别时将其归为某些类的边界域，等待新的信息以帮助进一步决策。

2.2 三支 k-means 算法

传统的 k-means 算法是一种迭代求解的聚类分析算法，其步骤是先随机选取 k 个对象作为初始的聚类中心，然后计算每个对象与各个子聚类中心之间的距离，把每个对象分配给距离它最近的聚类中心，通过不断更新迭代聚类中心直到满足某个终止条件。在 k-means 算法的处理过程中，样本和各类簇之间只存在两种关系：属于该类簇和不属于该类簇。如果样本在该类簇则属于该类簇，否则就不属于该类簇。为了消除不确定性信息在 k-means 聚类过程中带来的风险，三支 k-means(TWKM)聚类算法[7]将三支决策理论与 k-means 算法相结合，在 k-means 迭代过程中引入容许误差，将每个类簇的结果用核心域和边界域来表示，较好地解决了不确定性数据的处理问题。三支 k-means 聚类算法主要分为两步：

第一步　以其他聚类中心与对象的距离与最小距离的差小于阈值 q 为依据，把对象分配到每一类簇的上界中。假设有一个对象 v，随机选取 k 个质心，通过计算对象 v 到 k 个质心的最短距离，即 $d(v,x_i)=\min_{i\leqslant j\leqslant k}d(v,x_j)$，计算得到集合 $\{j:d(v,x_j)-d(v,x_i)\leqslant q\wedge i\neq j\}$，$q$ 参数的值是给定的，会产生如下两种情况：

(1)假如 $U=\varnothing$，则对象 $v\in C_i^u$；

(2)假如 $U\neq\varnothing$，则对象 $v\in C_i^u\wedge v\in C_j^u$。

我们可以通过以上两种方式获得每个类簇的上近似区域的对象，然后利用式(1)更新每个类簇的质心。

$$x_i=\frac{\sum_{v\in c_i^u} v}{|c_i^u|},i=1,2,\cdots,k \tag{1}$$

其中，v 是类簇 C_i^u 上近似区域中的所有对象，C_i^u 为类簇中对象的个数。

第二步　通过扰动分析法，将每个类簇的上界分成两个区域，即核心域和边界域。我们针对不同的类型采用不同的策略。

类型 $1=\{v\in C_i^u \mid \exists j=1,2,\cdots,k,j\neq i,v\in C_j^u\}$

类型 $2=\{v\in C_i^u \mid \forall j=1,2,\cdots,k,j\neq i,v\notin C_j^u\}$

如果一个对象 v 满足类型 1，就将这个对象分配到类 i 的上界，即该对象至少属于一个类。如果这个对象 v 满足类型 2，即该对象只能属于一个类，则在类 i 的上界中加入 n_i 个相同的对象 v，n_i 表示类 i 中对象的个数，得到类 i 新的上界 $C_i^{u^*}$。然后通过式(1)计算 $C_i^{u^*}$ 的质心 x_i^*，比较新旧质心之间的距离即 $|x_i-x_i^*|$，如果 $|x_i-x_i^*|\leqslant p$，p 参数的值是给定的，对象 v 被赋给类 i 的核心区域；反之，对象 v 被赋给类 i 的边界区域。算法的步骤如算法 1 所示。

算法 1　TWKM 算法

输入：$V=\{v_1,v_2,\cdots,v_n\}$，聚类数目 k，参数 q，p

输出：$C_1,C_2,\cdots,C_k$

1. 随机选择 k 个聚类中心 $x_1,x_2,\cdots,x_n$；
2. for k←1 ton do
3. repeat
4. 计算 v_i 离最近的聚类中心 x_i 的距离：$d(v_i,x_i)=\min\limits_{1\leqslant j\leqslant k} d(v_i,x_j)$，得到集合 $U=\{j:d(v_i,x_j)-d(v_i,x_i)\leqslant q\wedge i\neq j\}$；
5. if U≠Ø then
6. 将对象 v_i 聚到类 i 的上界 $v\in C_i^u$；
7. else
8. 将对象 v_i 同时聚到类 i 和类 j 的上界，即 $v_i\in C_i^u\wedge v_i\in C_j^u$；
9. end if
10. end for
11. 使用式(1)重新计算聚类中心；
12. for i←1 to k do
13. 样本 $v\in C_i^u$，定义集合 $Y=\{v\in C_i^u \mid \exists j=1,2,\cdots,k,j\neq i,v\in C_j^u\}$；
14. if U≠Ø then
15. 将样本 v 分配到类 C_i 的边界域 $v\in Fr(c_i)$；
16. else
17. 则在类 i 的上界中加入 n_i 个相同的对象 v，n_i 表示类 i 中对象的个数，得到类 i 新的上界 $C_i^{u^*}$，使用式(1)计算 $C_i^{u^*}$ 的质心 x_i^*，然后计算新旧质心之间的距离即 $|x_i-x_i^*|$；；
18. if $|x_i-x_i^*|\leqslant p$ then
19. 将 v 归到类 i 的核心域即 $Co(c_i)$；
20. else
21. 将 v 归到类 i 的边界域即 $Fr(c_i)$；
22. end if
23. end if
24. end for

3　基于人工蜂群的三支 *k*-means 聚类算法

三支 *k*-means 聚类算法可以对边界不确定数据进行更加合理的分类，但仍然存在对初始聚类中心敏感的问题。为了解决这一问题，本节将人工蜂群算法与三支 *k*-means 聚类算法相结合，提出了一种基于人工蜂群的三支 *k*-means 聚类算法。

3.1　人工蜂群

大自然中许多生物都有一定的群智能行为，如蜜蜂、蚂蚁、鸟、鱼群等，它们通过群体之间的相互合作与竞争来获得食物。自然界的蜜蜂总能在任何环境下以极高的效率找到优质蜜源。人工蜂群算法(ABC 算法)是由 Karaboga 等[10]于 2009 年模拟自然界中蜂群的采蜜行为来优化代数问题而提出的一种群智能算法。算法简单，参数少，容易实现，具有较强的全局搜索能力与局部搜索能力，适合对多维数据的处理。文献[17]提出了人工蜂群算法来解决聚类的问题；文献[18]采用粒子群算法来解决聚类问题；文献[19]提出了蚁群算法来解决聚类问题。人工蜂群算法的主要思想是将蜜蜂系统分为蜜源、引领蜂、跟随蜂、侦察蜂，将蜜源位置转化成优化问题的可行解，蜜源的含蜜量对应优化问题的适应度函数，蜂群寻找蜜源的过程是求最优解的过程。通常引领蜂的个数和蜜源的个数相等，一个蜜源对应一个引领蜂。人工蜂群算法主要分为以下 4 个阶段。

(1)初始化阶段：在搜索空间中随机产生 N 个食物源(即蜜源的位置)$\{x_1,x_2,\cdots,x_n\}$，它是一个 D 维的向量，每个食物源代表一个可行解。引领蜂和跟随蜂各为 N，蜂群的最大迭代次数为 $Maxgen$，每个蜜源的最大搜索次数为 $Limit$。

(2)引领蜂阶段：引领蜂按式(2)在蜜源位置附近寻找新的蜜源，引领蜂采用一种贪婪准则，比较记忆中的最优解与邻域搜索解，这个邻域搜索解就是我们已知的蜜源邻域搜索解。当邻域搜索解大于最优解时更新蜜源位置，反之不变。

$$V_{ij}=x_{ij}+\varphi(x_{ij}-x_{kj}) \tag{2}$$

其中，V_{ij} 表示新的蜜源位置，j 代表解的某一个维度，i 代表目前的引领蜂，k 是除了 i 之外的某个引领蜂，φ_{ij} 是介于$[-1,1]$之间的随机数。

(3)跟随蜂阶段：根据引领蜂的蜜源信息采用轮盘赌的方法选择引领蜂进行跟随。跟随蜂利用式(3)计算得到的概率选择引领蜂。同样，跟随蜂根据式(2)在蜜源的附近邻域搜索产生一个新的蜜源。蜜量大的引领蜂吸引跟随蜂的概率大于蜜量小的引领蜂，适应度越高蜜源越好。

一个蜜源由跟随蜂选择的概率表示为：

$$p_i=\frac{fit_i}{\sum_{k=1}^{N} fit_i} \tag{3}$$

其中，fit_i 是第 i 个解的适应度值，N 是解的个数，p_i 是第 i 个解的概率。当引领蜂在某个蜜源搜索的次数达到 $Limit$ 后，蜜源不再发生变化，此时引领蜂就离开该蜜源，不再对这个蜜源进行开采，转而变成侦察蜂，随机产生新的蜜源。

(4)侦察蜂阶段：若某处的蜜源经过 $Limit$ 次后仍未找到邻近更优的蜜源，此时该蜜源已经陷入了局部最优解，则调用侦察蜂利用式(4)随机生成一个新的蜜源位置。

$$x_{ij}=x_{ij}+rand*(x_j^{\max}-x_j^{\min}) \tag{4}$$

其中，$i=1,2,\cdots,N$，N 表示食物源的个数；$i=1,2,\cdots,D$，D 表示解的维数；$x_j^{\max}$ 表示第 j 维的上限，$x_j^{\min}$ 表示第 j 维的下限，相当于全局搜索的能力，让它跳到另一个地方继续开采。

3.2　构造适应度函数

在人工蜂群算法中，吸引蜂群的主要因素取决于蜜源含蜜量的多少。蜜源的含蜜量通常由适应度函数决定，同时适应度函数也决定着蜂群的进化方向、迭代次数以及解的

优劣。适应度越高，蜜源质量越好，就越能吸引蜂群。我们通过定义类内聚集度函数和类间离散度函数来构造适应度函数，使得同一类别内的对象尽可能相似，不同类别之间的对象最大程度相异。

定义 1(类内聚集度函数) 假设一个数据集 $U=\{x_1,x_2,\cdots,x_n\}$，n 个数据对象，有 K 个聚类中心 $C_k=\{m_1,m_2,\cdots,m_k\}$，因为核心域和边界域都对类内聚集度函数有影响，因此我们定义了一个权重，则权重和类内聚集度函数为：

$$w\_core=|icore|/(|icore|+|ifrine|),w\_ifringe=1+w\_core \tag{5}$$

$$J(C_K)=\sum_{k=1}^{K}w\_core\sum_{x_i\in icore}\sqrt{|x_i-m_k|^2}+w\_core\sum_{x_i\in ifringe}\sqrt{|x_i-m_k|^2} \tag{6}$$

其中，$J(C_K)$表示每个对象到对应聚类中心的所有距离总和，x_i 表示每个要计算的对象，m_k 表示聚类中心，w_core 表示核心域的权重，$w\_ifringe$ 表示边界域的权重，$|icore|$ 和 $|ifringe|$ 分别表示核心域和边界域中对象的个数。

定义 2(类间离散度函数) 假设一个数据集 $U=\{x_1,x_2,\cdots,x_n\}$，n 个数据对象，有 K 个聚类中心 $C_k=\{m_1,m_2,\cdots,m_k\}$，则聚类中心的类间离散度函数为：

$$D(C_K)=\sum_{k=1}^{K}w\sum_{j=i+1}^{k}\sqrt{|m_i-m_j|^2},w=\frac{1}{\sqrt{k}} \tag{7}$$

其中，$D(C_K)$表示所有聚类中心之间的距离总和，m_i 和 m_j 表示类 c_i 和类 c_j 的聚类中心。

当迭代次数不断增加时，相同类中的聚集度会不断下降，不同类中的离散度会不断扩大。为了有效防止聚类中心成为异常点或在稀疏区域，通过式(7)定义一个权重系数 w，使得类内聚集度和类间离散度间的距离更加平衡，从而避免聚类中心成为异常点或在稀疏区域，更加符合数据分布的特点。

定义 3(适应度函数)

$$fit_i=\frac{1+D(C_K)}{1+J(C_K)} \tag{8}$$

其中，fit_i 表示适应度的函数值，$D(C_K)$表示类间离散度函数值，$J(C_K)$表示类内聚集函数值。当 $D(C_K)$的值越大，$J(C_K)$的值就越小，fit_i 的值越大，获得的聚类效果就越好。

3.3 算法实现的步骤

鉴于人工蜂群算法和三支 k-means 算法各自的特性，本文提出了一种基于人工蜂群的三支 k-means 聚类算法。该算法的主要思想是利用蜂群之间的相互协作与互换对数据集进行多次迭代，找到最优的蜜源位置作为聚类中心，并在此基础上进行三支 k-means 聚类。通过构造适应度函数 fit_i 来引导蜂群向高质量的蜜源进行搜索，适应度函数代表蜜源的质量。由式(8)可知，类内聚集度距离越小，类间离散度距离越大，适应度函数 fit_i 越高，蜜源质量越好，就越能吸引蜂群，在寻找到最优的聚类中心的同时找到最优的聚类结果。研究表明该方法进一步提高了聚类结果的稳定性与精度，提高了算法的鲁棒性，使得聚类的性能指标 DBI 值更小，AS 和 ACC 的值更大，进一步避免了因为盲目聚类而带来的决策风险。算法的步骤如算法 2 所示。

算法 2 基于人工蜂群的三支 k-means 聚类算法

输入：数据集 $V=\{v_1,v_2,\cdots,v_n\}$，参数 N，Maxgen，Limit，聚类数目 k

输出：聚类结果 $T=\{(Co(c_1),Fr(c_1)),(Co(c_2),Fr(c_2)),\cdots,(Co(c_k),Fr(c_k))\}$

1. 初始化种群，随机产生 N 个蜜源位置，选取 k 个蜜源 $z_1,z_2,\cdots,z_k$ 作为聚类中心；
2. for gen←1 to Maxgen do
3. for i←1 to n do
4. 根据算法 1 的第 1－10 步计算样本对象 v_i 到各个聚类中心的距离，并将 v_i 规划到对应类别的上界中；
5. end for
6. for i←1 to n do
7. 根据式(8)计算各个蜜源的适应度函数；
8. end for
9. for i←1 to n do
10. 引领蜂按式(2)在蜜源位置附近寻找新的蜜源，计算这个新蜜源位置的适应度函数值，采用贪婪的原则保留适应度函数较高的蜜源；
11. if($fit_{new}>fit_{old}$)
12. 更新蜜源位置，并保留下来；
13. else
14. Limit=Limit+1；
15. end for
16. for i←1 to N do
17. 引领蜂完成搜索后由式(3)计算每个蜜源的概率；
18. end for
19. for i←1 to N do
20. 根据概率 p_i，跟随蜂利用轮盘赌规则选择引领蜂，完成选择后，按式(2) 进行邻域搜索，同样采用贪婪的原则保留适应度较高的蜜源；
21. end for
22. 所有跟随蜂完成搜索后，将得到的蜜源位置作为新的聚类中心，对于任意对象 v_i 根据算法 1 的第 12－24 步，将对象 v_i 分配到对应类别的 $Fr(c_k)$，否则划分到对应类别的 $Co(c_k)$中；
23. 根据第 16 步得到的划分结果利用式(1)重新计算聚类中心；
24. 若引领蜂在搜索 Limit 次后仍未找到更优的蜜源位置，则变为侦察蜂，利用式(4)随机产生一个新的蜜源位置；
25. 当算法达到最大迭代次数 Maxgen 后，算法结束并输出最终的聚类结果，否则转到步骤 2，Maxgen=Maxgen+1；
26. end for
27. return $T=\{(Co(c_1),Fr(c_1)),(Co(c_2),Fr(c_2)),\cdots,(Co(c_k),Fr(c_k))\}$.

4 聚类评价指标

(1) 准确率

ACC 是最常见的一种外部评价指标，其对预测的结果与真实值做对比，值越高说明聚类效果越好。

定义 4(ACC[20])

$$ACC=\frac{1}{N}\sum_{i=1}^{k}U_i \tag{9}$$

其中，N 表示样本总数，k 表示类簇数量，U_i 表示正确聚类到类 i 的样本数量。本文的三支聚类算法在实验中所计算的 ACC 是使用核心域对象来计算的。

(2)DBI 评价指标

DBI 又被称为分类适确性指标，是 Davies 等[21]提出的用于评估聚类算法优劣的指标，该指标的核心思想是度量每个类簇最大相似度的均值。

定义 5(DBI[21])

$$DBI=\frac{1}{N}\sum_{i=1}^{N}\max_{j\neq i}\left(\frac{\bar{s_i}+\bar{s_j}}{\|\bar{\omega}_i-\bar{\omega}_j\|_2}\right) \tag{10}$$

其中，N 表示类簇数量，$\bar{s}_i$ 表示类内样本到簇中心 $\bar{\omega}_i$ 的平均距离，$\| \bar{\omega}_i - \bar{\omega}_j \|_2$ 表示类 i 和类 j 之间的欧氏距离。

(3) Average Silhouette Index

1986 年 Rousseeuw 提出了一个用于评价聚类好坏的内部指标，称作轮廓系数(Silhouette Index)。

定义 6(单个样本点 x_i 的轮廓系数 S_i[22])

$$S_i = \frac{b_i - a_i}{\max(a_i, b_i)} \tag{11}$$

其中，a_i 又被称为类内相似度，表示样本点 v_i 与其所属类簇中其他样本的平均距离，该值越大，说明这个样本属于该类簇的可能性就越大；b_i 又被称为类间相似度，$b_i = \min\{D(v_i - c_j)\}$ 表示样本 v_i 到类 c_j 中所有样本的最新距离，该值越大，说明这个样本属于其他类的可能性越小。

定义 7(平均轮廓系数 AS[22]) 平均轮廓系数就是对所有样本点的轮廓系数求平均，轮廓系数的值介于[-1,1]之间。平均轮廓系数是衡量聚类结果好坏的一个指标，这个值越大，说明样本属于该类簇的可能性就越大。

$$AS = \frac{1}{N}\sum_{i=1}^{N} S_i \tag{12}$$

其中，i 表示样本的数量，S_i 表示第 i 个样本的轮廓系数。

5 实验分析

为了对本文提出的基于人工蜂群的三支 k-means 聚类算法的有效性进行验证，本文选取了 13 组 UCI 数据集。数据集如表 1 所列。

表 1 实验中使用的数据集

Table 1 Datasets used in experiments

	Datasets	Numbers	Dimensions	Categories
1	Wine	178	13	3
2	Class	214	9	6
3	Congressional Voting	435	16	2
4	Iris	150	4	3
5	Bank	1372	4	2
6	Fertility	100	9	2
7	Hill_valley_without_noise	1212	100	2
8	Libras_Nor	360	90	15
9	Zoo	101	16	7
10	Contraceptive	1473	9	3
11	Caffeine_consumption	1885	12	7
12	Molecular Biology	106	57	2
13	Magic04	19020	10	2

本文的聚类结果用核心域表示，利用核心域中的对象来代表聚类，分别计算 DBI，AS，和 ACC 的值，从而进行评价。DBI 值越小，AS 和 ACC 值越大，聚类效果就越好。为了能更好地体现出本文算法在 DBI，AS 和 ACC 性能指标上有所提升，将所提算法与 k-means，FCM 和 TWKM 算法进行聚类性能指标比较。在本文算法中，阈值是非常重要的参数，如何设置阈值也是一项重要研究内容。本文根据文献[7]，通过前期大量的实验得到了一个适合的阈值。通过实验得到参数 $p=0.004$，$q=0.26$。

表 2 UCI 数据集上的结果

Table 2 Experimental results on UCI dataset

ID	Algorithm	DBI	AS	ACC	ID	Algorithm	DBI	AS	ACC
1	k-means	1.3053	0.4763	0.9550	8	k-means	1.9240	0.3519	0.0861
	FCM	1.3210	0.4748	0.9577		FCM	2.0066	0.3128	0.0889
	TWKM	1.2572	0.4952	0.9634		TWKM	1.6372	0.4092	0.0883
	本文算法	**1.1846**	**0.5013**	**0.9636**		本文算法	**1.4174**	**0.4816**	**0.0969**
2	k-means	1.0050	0.5309	0.5981	9	k-means	1.1639	0.5011	0.7623
	FCM	1.2670	0.4895	0.5934		FCM	1.4737	0.4303	0.7821
	TWKM	1.1432	0.5073	0.6106		TWKM	1.1690	0.5206	0.7742
	本文算法	**0.9606**	**0.5678**	**0.6419**		本文算法	**1.1400**	**0.5237**	**0.7979**
3	k-means	1.4865	0.4407	0.8666	10	k-means	1.2706	0.4236	0.2145
	FCM	1.4857	0.4401	0.8643		FCM	1.2775	0.4175	0.2124
	TWKM	1.3602	0.4728	0.8326		TWKM	1.2673	0.4159	0.1925
	本文算法	**1.3031**	**0.5010**	**0.8886**		本文算法	**1.2537**	**0.4348**	**0.2183**
4	k-means	0.7609	0.6959	0.8866	11	k-means	1.9008	0.3160	0.1999
	FCM	0.7872	0.6711	0.9000		FCM	**0.9273**	0.3159	0.2074
	TWKM	0.7327	0.6952	0.9072		TWKM	1.6932	**0.3627**	0.2159
	本文算法	**0.7183**	**0.7293**	**0.9118**		本文算法	1.8069	0.3499	0.2236
5	k-means	1.1913	0.5000	0.5758	12	k-means	4.9545	0.0565	0.5815
	FCM	1.1922	0.4986	0.5860		FCM	5.5119	0.0491	0.5999
	TWKM	1.1529	0.5204	0.5735		TWKM	4.6351	0.0583	0.6001
	本文算法	**1.1181**	**0.5294**	**0.6235**		本文算法	**4.1019**	**0.0608**	**0.6128**
6	k-means	1.7968	0.3647	0.5100	13	k-means	1.2056	0.4008	0.4068
	FCM	1.9373	0.3206	0.5232		FCM	1.2324	0.4137	0.4622
	TWKM	1.7534	0.3697	0.5163		TWKM	0.8367	0.6337	0.6591
	本文算法	**1.6653**	**0.3714**	**0.5344**		本文算法	**0.7963**	**0.6527**	**0.6828**
7	k-means	0.4047	0.9376	0.4636					
	FCM	0.4067	0.9370	0.4628					
	TWKM	0.3841	0.9485	0.4569					
	本文算法	**0.3678**	**0.9522**	**0.4680**					

本实验对每组数据做了 100 次实验，并取平均值作为实验结果，比较算法的总体性能，实验结果如表 2 所列。

根据表 2 的实验结果可以发现，本文算法在大数据和高维数据集上聚类性能指标都显著优于 k-means，FCM 和 TWKM 算法，唯独在 Caffeine_consumption 数据集上，本文算法没有达到预期的效果，FCM 和 TWKM 算法在该数据集上的性能指标 DBI 和 AS 的平均值优于本文算法。尽管本文算法在数据集 Caffeine_consumption 上的性能指标不如 FCM 和 TWKM 算法，但在大部分数据集上本文算法的总体性能指标要优于其他对比算法。由此可见，本文提出的基于人工蜂群的三支 k-means 聚类算法在大部分数据集上都能提高 AS 和 ACC 的值，降低 DBI 的值，能有效地避免盲目聚类带来的决策风险，克服了聚类中心因随机选取而导致聚类结果易陷入局部最优的问题。综上所述，本文算法能够有效地提高聚类精度，显示出更好的聚类结果。

结束语 本文将人工蜂群算法与三支 k-means 聚类算法相结合，提出了基于人工蜂群的三支 k-means 聚类算法。通过人工蜂群的全局寻优能力和局部寻优能力对聚类中心进行更新迭代，有效地避免了因收敛速度快而陷入局部最小值。采用三支决策规则，在处理类内孤立数据和类与类的重叠数据时，有效地提高了聚类结果的准确性，降低了决策风险。本文算法的不足之处在于，k 的取值是根据已知数据集给定好的，k 取不同值时对聚类结果影响较大，因此后续将研究自适应寻找最佳类簇 k。另外，本文算法在边界域的处理上还不太理想，这也是接下来要研究的主要内容。

参 考 文 献

[1] LU D, TRIPODIS Y, GERSTENFELD L C, et al. Clustering of temporal gene expression data with mixtures of mixed effects models with a penalized likelihood[J]. Bioinformatics, 2019, 35(5): 778-786.

[2] KALYANI S, SWARUP K S. Particle swarm optimization based k-means clustering approach for security assessment in power systems [J]. Expert Systems with Applications, 2011, 38(9): 10839-10846.

[3] SONG L H, ZHANG X F. Improved pixel relevance based on Mahalanobis distance for image segmentation [J]. International Journal of Information and Computer Security, 2018, 10(2/3): 237-247.

[4] SUN J G, LIU J, ZHAO L Y. Clustering algorithms research [J]. Journal of Software, 2008, 19(1): 48-61.

[5] WU X, KUNMAR V, QUINLAN J R. Top 10 algorithms in data mining [J]. Knowledge and Information Systems, 2008, 14(1): 1-37.

[6] LEI X F, XIE K Q, LIN F, et al. An efficient clustering algorithm based on local optimality of k-meams [J]. Journal of Software, 2008, 19(7): 1683-1692.

[7] WANG P X, SHI H, YANG X B, et al. Three-way k-means: integrating k-means and three-way decision[J]. International Journal of Machine Learning and Cybernetics, 2019, 10: 2767-2777.

[8] YAO Y Y. The superiority of three-way decisions in probabilistic rough set models[J]. Information Science, 2011, 181(6): 1080-1096.

[9] YAO Y Y. An outline of a theory of three-way decisions[C]// International Conference on Rough Sets and Current Trends in Computing. Berlin, Heidelberg: Springer, 2012.

[10] KARABOGA D, BASTURK B. A comparative study of artificial bee colony algorithm [J]. Applied Mathematics and Computation, 2009, 214(1): 108-132.

[11] YU H, WANG G Y, YAO Y Y. Current research and future perspectives on decision-theoretic rough sets [J]. Journal of Computer, 2015, 38(8): 1608-1639.

[12] YAO Y Y, DENG X F. Sequential three-way decisions with probabilistic rough set[C]// Proceedings of the 10th IEEE International Conference on Cognitive on Cognitive Informatics & Cognitive Computing. Banff, Canada, 2011: 120-125.

[13] YU H. A framework of three-way cluster analysis[C]// International Joint Conference on Rough Sets. Cham: Springer, 2017.

[14] WANG P X, LIU Q, YANG X B, et al. Three-way Cluster-ring Analysis based on Dynamic Neighborhood [J]. Computer Science, 2018, 45(1): 62-66.

[15] WANG P X, YAO Y Y. CE3: A three-way clustering method based on mathematical morphology[J]. Knowledge-Based Systems, 2018, 155: 54-65.

[16] YU H, ZHANG C, WANG G Y. A tree-based incremental overlapping clustering method using the three-way decision theory [J]. Knowledge-based systems, 2016, 91(C): 189-203.

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

[18] VAN DER MERWE D W, ENGELBRECHT A P. Data clustering using particle swarm optimization[C]// The 2003 Congress on Evolutionary. Canberra: IEEE, 2003: 215-220.

[19] SHELOKAR P S, JAVARAMAN V K, KULKAMARNI B D. An ant colony approach for clustering [J]. Analytica Chimica Acta, 2004, 509: 187-195.

[20] SCHOLKOPF B, PLATT J, HOFMANN T. A local learning approach for Clustering [C]// International Conference on Neural Information Processing Systems. Vancouver, Canada: MIT Press, 2007: 1529-1536.

[21] DAVIES D L, BOULDIN D W. A cluster separation measure [J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 1979, 1(2): 224.

[22] BEZDEK J C, PAL N R. Some new indexes of cluster validity [J]. IEEE Transactions on Systems, Man, and Cybernetics, 1998, 28(3): 301-315.

XU Tianjie, born in 1996, postgraduate. His main research interests include rough sets and three-way decision.

WANG Pingxin, born in 1980, Ph.D, associate professor, master supervisor. His main research interests include matrix analysis, three-way decision, and rough set.

（责任编辑：杨雪敏）