ISSN 1004-9037,CODEN SCYCE4
Journal of Data Acquisition and Processing Vol. 34,No. 2,Mar. 2019,pp. 341－348
DOI:10. 16337/j. 1004-9037. 2019. 02. 017

http://sjcj. nuaa. edu. cn
E-mail:sjcj@nuaa. edu. cn
Tel/Fax:＋86-025-84892742

结合蚁群算法的改进粗糙*K*均值聚类算法

刘　洋[1]　王慧琴[1,2]　张小红[2,3]

(1. 西安建筑科技大学信息与控制工程学院,西安,710055;2. 西安建筑科技大学管理学院,西安,710055;3. 西安科技大学通信与信息工程学院,西安,710054)

摘　要: 粗糙集理论是一种处理边界对象不确定的有效方法。将粗糙集与*K*均值结合的粗糙*K*均值聚类算法,具有简单高效且可处理聚类边界元素的特点,但同时存在缺陷。针对粗糙*K*均值聚类算法对初始点敏感,经验权重设置忽略数据差异性,阈值设置不合理导致聚类结果波动性大的缺陷,本文提出结合蚁群算法的改进粗糙*K*均值聚类算法,改进的算法中使用蚁群算法中随机概率选择策略和信息素更新的正负反馈机制,以及采用动态调整算法阈值和相关权重的方法,对粗糙*K*均值聚类算法进行优化。最后采用UCI的Iris、Balance-scale和Wine数据集分别对算法进行实验。实验结果表明,改进后的粗糙*K*均值聚类算法得到的聚类结果准确率更高。

关键词: 聚类;*K*均值;蚁群算法;粗糙集;目标函数

中图分类号: TP312　　**文献标志码:** A

An Improved Rough *K*-means Clustering Algorithm Combining Ant Colony Algorithm

Liu Yang[1], Wang Huiqin[1,2], Zhang Xiaohong[2,3]

(1. School of Information and Control Engineering, Xi′an University of Architecture and Technology, Xi′an, 710055, China; 2. School of Management, Xi′an University of Architecture and Technology, Xi′an, 710055, China;3. School of Communication and Information Engineering, Xi′an University of Science and Technology, Xi′an, 710054, China)

Abstract: Rough set theory is an effective method for dealing with uncertain boundary objects. The rough *K*-means clustering algorithm which combines rough set with *K*-means is simple and efficient. Though it can deal with clustering boundary elements, it has some drawbacks, for instance, the original rough *K*-means clustering algorithm is sensitive to the initial center, the set-up of empirical weigh ignores data difference, the unreasonable threshold setting engenders fluctuation of clustering results. To tackle these drawbacks, this paper proposed an improved rough *K*-means clustering algorithm combined with ant colony algorithm. The improved algorithm is optimized for rough *K*-means clustering by using random probability selection strategy and pheromone update of positive and negative feedback mechanisms in ant colony algorithm, and using dynamic threshold adjustment algorithm and associated weights method. Finally, the UCI's Iris set, Balance-scale set and Wine set are used for verification of the algorithm. The results show that this algorithm exhibits a higher clustering accuracy.

Key words: cluster; *K*-means; ant colony algorithm; rough sets; objective function

基金项目: 教育部归国留学人员科研扶持基金(K05055)资助项目;教育部高等学校博士学科点专项科研基金、博导类联合(20126120110008)资助项目。

收稿日期: 2017-08-26;**修订日期:** 2017-10-09

引 言

聚类是指按照一定的相似性度量准则将物理或抽象对象的集合划分成多组具有同类性质的子类(簇),同一子类(簇)中的对象相似度较高,而不同子类(簇)中的对象差别较大[1]。聚类在生态学领域,医学医疗领域,经济学领域,以及电子商务领域等很多领域都有着非常广泛的应用[2]。不同的领域中,有适用于该领域的聚类算法。基本的聚类算法可分为层次聚类、划分聚类、密度聚类、模型聚类和网格聚类[3]。K均值聚类算法是一种经典的划分聚类算法,它将对象按照欧氏距离来进行划分。在实际中样本对象之间没有固定的边界,往往会出现某个对象不是确定地属于某一类。

粗糙集理论[4]是Pawlak提出的一种可以有效处理不确定性问题的方法。Lingras提出了粗糙K均值聚类算法,它能较好地处理无法精确分类的样本数据[5]。粗糙K均值聚类算法根据样本对象到类(簇)心欧氏距离的不同,将其划分到上近似区域和下近似区域。上近似区域中的样本对象可能属于某类(簇),下近似区域中的样本对象确定属于某类(簇),因此粗糙K均值聚类算法能较好地处理无法精确分类的样本数据。粗糙K均值聚类算法存在以下不足:初始质心的选取是随机的,导致聚类结果可能不是全局最优而是局部最优;阈值的选择不合理导致聚类结果的波动性较大;经验权重的设置容易忽略了数据分布的差异性。

国内外文献针对粗糙K均值聚类算法的不足之处进行了研究与改进,文献[6]根据样本密度迭代地选取粗糙K均值算法的初始聚类中心,同时提出根据样本密度给每个样本赋以不同的权重计算类质心,但是没有考虑到,固定经验权重无法很好地适应聚类前期和后期的特点。文献[7]考虑了每个对象的密度和赋予不同的权重,但缺少对上近似和下近似整的体权重的考虑。文献[8]认为在聚类过程中同一类中的各个样本对类心的影响是不相同的,因此针对每一个样本都计算其所属类别的权重,但却没考虑阈值随聚类过程的变化对聚类结果的影响。文献[9]采用遗传算法与粗糙K均值结合虽得到全局最优解,却忽略了固定经验权重无法很好地适应聚类前期和后期的特点。文献[10]采用自适应获得参数的方法,避免了人为设置固定经验权重对算法影响的缺点,提高了算法的准确性,却没考虑到粗糙K均值聚类算法对初始点敏感的缺陷。文献[11]中通过构造Logistic增长曲线来计算下近似区域的权重,但当算法的迭代次数过大时下近似区域的权重几乎不再改变,对于不同的数据集忽略了其数据分布的差异性。

本文在前人对粗糙K均值聚类算法研究的理论基础上进行改进,结合蚁群算法中蚂蚁决策行为时的随机概率选择策略,以及蚁群算法的正负反馈机制,对粗糙K均值聚类算法的初始质心的选择进行优化,以改进基本粗糙K均值聚类算法初始质心随机选择导致聚类结果可能是局部最优而不是全局最优的缺点,同时采用动态调整阈值和相关权重的方法,避免人为设置对算法的影响。

1 粗糙K均值聚类算法

粗糙集的思想为确定属于某个类的样本点在该类的下近似区域中,可能属于某个类的样本点在该类的上近似区域中[7]。在K均值聚类算法的基础上引入粗糙集的思想,即为粗糙K均值聚类算法。

用x_i表示样本点,C_k表示k个类,每个类对应的的上近似表示为$\overline{C_k}$和下近似表示为$\underline{C_k}$,类C_k的边界区域表示为C_k^B,其中$\left|C_k^B\right|=\left|\overline{C_k}\right|-\left|\underline{C_k}\right|$,$m_k$表示类心。$m_k$计算公式为

$$m_k=\begin{cases}\omega_1\sum\limits_{x_i\in\underline{C_k}}\dfrac{x_i}{\left|\underline{C_k}\right|}+\omega_b\sum\limits_{x_i\in C_k^B}\dfrac{x_i}{\left|C_k^B\right|} & C_k^B\neq\varnothing\\ \sum\limits_{x_i\in\underline{C_k}}\dfrac{x_i}{\left|\underline{C_k}\right|} & C_k^B=\varnothing\end{cases}\tag{1}$$

式中:|·|表示样本点个数;ω_1表示下近似区域权重,ω_b表示边界区域的权重,且$\omega_1+\omega_b=1$;∅表示空集。

首先将每个$x_i(i=1,2,\cdots,N)$分到任意一个$C_k(k=1,2,\cdots,K)$的$\underline{C_k}$中,之后依照公式(1)计算m_k。接下来计算每个x_i与各个m_k的距离d_{ki},并找出与x_i距离最近的$m_{k'}$,即x_i与$m_{k'}$之间的距离为$\min(d_{k'i})$。若存在其他$m_k(k\neq k')$与x_i之间的d_{ki}与$\min(d_{k'i})$之差小于阈值,则把该x_i划分到这些类的$\overline{C_k}$,否则把x_i加入到$m_{k'}$所对应的类$C_{k'}$的$\underline{C_{k'}}$。再次更新类心,不断循环上述过程,直到每个类的m_k不变[10]。

2　结合蚁群算法的改进粗糙K均值聚类算法

蚁群算法是对真实蚂蚁觅食行为的一种抽象,算法中人为定义的蚂蚁代替了真实的蚂蚁。蚂蚁从一个样本点移动到聚类中心时,根据当前路径信息素量的大小依概率进行选择,蚂蚁在所经过的路径上留下信息素,以此影响本次迭代或者下次迭代蚂蚁选择该路径的概率。而信息素随着时间的推移不断挥发,这使蚂蚁不局限于过去的寻优结果,能跳出局部最优解,发现新的解。

结合蚁群算法中蚂蚁决策行为时的随机概率选择策略以及蚁群算法的正负反馈机制,以此增加聚类结果的多样性,且不断搜索更优结果,在避免算法陷入局部最优解的同时达到改进粗糙K均值聚类算法对初始点敏感的缺陷。同时粗糙K均值聚类算法中下近似与边界区域权重的通过上、下近似元素数来衡量,阈值采取自动获得,都避免了人为设计对聚类结果的影响,以此达到改进粗糙K均值聚类算法的目的。

2.1　随机概率选择策略增加解的多样性

在蚁群算法中,蚂蚁搜索所有样本所属的聚类中心时,根据当前信息素量的大小依概率进行随机选择,某样本与某一聚类中心之间的信息素量越大,该样本点被选择归入该类的概率越大。随机概率选择策略使算法拥有收敛的特性,同时有利于搜索更优的结果,避免算法陷入局部最优。

样本到聚类中心m_k的距离d_{ij}为欧式距离,启发式函数定义为距离的倒数,即

$$\eta_{kj}=\frac{1}{d_{kj}} \tag{2}$$

式中:$k=1,2,\cdots,K$,其中K表示聚类数目;$j=1,2,\cdots,N$,其中N表示数据集中样本总个数。蚂蚁在选择路径时,不仅利用启发式函数,而且用到了样本点与聚类中心之间的信息素。样本与聚类中心之间的信息素分布为τ_{kj},信息素分布在样本和聚类中心之间。

在算法的搜索过程中,样本点被分配到各个聚类中心的概率计算公式为

$$p_{kj}=\begin{cases}\dfrac{[\tau_{kj}(t)]^{\alpha}\times[\eta_{kj}(t)]^{\beta}}{\sum\limits_{s\in \mathrm{allow}_m(i)}^{M}[\tau_{ks}(t)]^{\alpha}\times[\eta_{ks}(t)]^{\beta}} & q\leqslant \mathrm{rand}\\ \max([\tau_{kj}(t)]^{\alpha}\times[\eta_{kj}(t)]^{\beta}) & \text{其他}\end{cases} \tag{3}$$

式中:α为信息素的相对重要程度,β为启发式因子的相对重要程度,M为蚂蚁总数($m\in[0,M]$),$q\in[0,1]$为给定参数,随机产生的$\mathrm{rand}\in[0,1]$,t为迭代次数。$\mathrm{allow}_m(i)$为禁忌表之外的样本集合,s为禁忌表中第s个元素,即蚂蚁m所走过的第s个样本。在算法的搜索过程中,为了加快收敛速度和增加搜索的多样性,蚂蚁随机选择一个$x_i\in\mathrm{allow}_m(i)$作为起始点,之后按照式(3)计算该样本选择各个聚类中心的概率p_{kj},最后根据轮盘赌的方法确定x_i所属的类,并将x_i归到相应的类中。之后,蚂蚁再随机选取一个不包含在禁忌表之内的样本,重复上述过程,直到将所有样本遍历,即形成一个解。

2.2 目标函数设计

在算法中所有蚂蚁完成构造解以后，通过目标函数进行评估，每次只保留目标函数值最优的蚂蚁所得到的聚类结果，因此粗糙K均值聚类可视为优化问题，优化的过程即为目标函数最小值的获得，即为最优的聚类结果。

$$J=\sum_{k=1}^{K}\left(\omega l_k\sum_{x_i\in\underline{C}_k}\left\|x_i-m_k\right\|^2+\omega b_k\sum_{x_i\in C_k^B}\left\|x_i-m_k\right\|^2\right) \tag{4}$$

式中：J表示类内距离，用来评价聚类的内聚程度；$\|\cdot\|^2$表示欧氏距离；ωl_k，ωb_k分别为第k簇下近似和边界区域的权重值。在聚类过程中动态地调整下近似与边界区域的权重，可以避免经验权重的设置导致忽略了数据分布的差异性。下近似与边界区域的元素个数可以衡量相对重要度比例

$$\frac{\omega l_k}{\omega b_k}=\frac{\left|\underline{C}_k\right|}{\left|\overline{C}_k\right|-\left|\underline{C}_k\right|},\quad \left|\underline{C}_k\right|\neq\varnothing \tag{5}$$

$$\omega l_k+\omega b_k=1 \tag{6}$$

考虑到聚类结果的好坏是由类内距离和类间距离共同作用，即尽量减小类内距离，增加类间距离。故本文采用能有效地均衡类间距离与类内距离的目标函数，当目标函数达到最小值时，得到最优的聚类结果。目标函数的计算公式为

$$F(t)=\ln(J/D) \tag{7}$$

$$D=\omega\sum_{i,j=1}^{K}\left\|m_i-m_j\right\|^2,\quad \omega=\frac{1}{K} \tag{8}$$

式中：D表示类间距离，用来评价类间分离程度，随着K增加内聚程度下降，而类间分离程度增加。为使类间距离和类内距离达到平衡，ω为类间分离程度权重[11]。

2.3 正负反馈机制

蚁群算法中各条路径上信息素在不断更新，最优路径上不断增加的信息素吸引更多的蚂蚁沿此路径前进，同时新增的蚂蚁在该路径上释放更多的信息素，不断加强该路径的吸引力，形成正反馈机制。差的路径上信息素被加强的程度较弱或者得不到加强，随着信息素不断挥发慢慢失去吸引力，形成负反馈机制[12]。

正反馈机制将更多的蚂蚁吸引到当前最优的路径上，促进了聚类算法的收敛。但当蚂蚁搜索得到的是局部最优解时，为防止更多的蚂蚁被吸引到该路径导致最终结果为局部最优解，负反馈机制则可以消除正反馈机制的作用。通过正负反馈机制对蚂蚁释放的信息素的作用，使蚂蚁在寻优过程当中，不局限于局部最优解，能不断发现新的解。

蚁群算法中的人工蚂蚁寻优利用的是蚁群的整体信息，即完成一次寻优后才进行残留信息素的全局更新。在蚁群算法中各路径上的信息素更新公式为

$$\tau_{kj}(t+1)=\rho\tau_{kj}(t)+\Delta\tau_{kj} \tag{9}$$

$$\Delta\tau_{kj}=Q/J_{F_{\min}} \tag{10}$$

式中：$J_{F_{\min}}$为目标函数取得最小值时的类内距离；$\Delta\tau_{kj}$为信息素的增量；Q为蚂蚁释放的信息素总量；$\rho(0<\rho<1)$为信息素挥发系数。

2.4 自动获取阈值

阈值T的作用是区分样本属于聚类上近似区域或下近似区域，阈值选择的适当才能确保上、下有充足的样本。阈值的取值使用文献[13]提出的方法，该方法可以自动获得阈值，避免人为设置对算法的

影响。

(1) 分别计算所有样本元素与K个聚类中心的距离$d(k,j)$。

(2) 找出$d(k,j)$中每列的最小值$d_{\min}(j)$。

(3) 计算各个样本元素和其他类心的距离与$d_{\min}(j)$的差值$d_t(k,j)$。

(4) 找出差值矩阵中每列的最小值$d_s(j)$且$d_s(j)\neq 0$,求这些最小值的平均值$d_s^{\text{mean}}(j)$。

(5) 阈值的取值为$T=\min(d_s^{\text{mean}}(j))$。

其中$k=1,2,\cdots,K$;$j=1,2,\cdots,N$。

2.5 生成新聚类中心

$$m_k=\begin{cases}\omega l_k\sum\limits_{x_i\in\underline{C}_k}\dfrac{x_i}{\left|\underline{C}_k\right|}+\omega b_k\sum\limits_{x_i\in C_k^B}\dfrac{x_i}{\left|C_k^B\right|} & C_k^B\neq\varnothing\\ \sum\limits_{x_i\in\underline{C}_k}\dfrac{x_i}{\left|\underline{C}_k\right|} & C_k^B=\varnothing\end{cases}\tag{11}$$

式中:$k=1,2,\cdots,K$;$j=1,2,\cdots,N$;ωl_k与ωb_k分别表示第k类的下近似权重和边界区域权重。

2.6 算法描述

蚂蚁随机选取一个样本点作为起点,参考启发式函数与路径上信息素的量,按照随机概率选择策略计算该样本选择各个聚类中心的概率,然后根据概率确定应该所属的类。然后蚂蚁再随机选取另外一个样本,重复上述过程,直到蚂蚁遍历所有样本,即形成了一个解。所有的蚂蚁构造完成解之后,使用目标函数评价出最优值并且保留,然后计算下近似区域权重和边界区域权重,最后重新计算各个簇的质心。信息素的更新为全局更新,在算法迭代的过程中,只聚类结果最优的蚂蚁的路径进行信息素增加,其余的蚂蚁的路径进行信息素衰减。

算法具体步骤如下:

(1) 设定各参数的值,初始信息素浓度$\tau_{kj}(0)=1$,求阈值。

(2) 初始聚类中心随机选取K个。

(3) 计算d_{kj}及η_{kj}。

(4) M只蚂蚁按照式(3)搜索方法独立地构造各自的解。

(5) 根据式(5,6)计算下近似权重和边界区域权重,根据式(11)重新计算各个簇的质心。

(6) 如果M只蚂蚁都构造完成各自的解,转步骤(7),否则转步骤(4)。

(7) 根据式(4,7,8)计算并比较M只蚂蚁求得的目标函数,保存其最优解和最优聚类结果。

(8) 根据式(9,10)进行全局信息素更新。

(9) 若满足结束条件(达到最大迭代次数),则输出最优解,否则进行迭代,转步骤(3)。

3 实　　验

3.1 评价指标

实验将聚类结果与实际通用UCI数据进行比较,以此评价聚类算法的性能的优良。常用的评价方法有准确率指标Rand[8]和Kappa系数[14]。

Rand计算公式为

$$\text{Rand}=\sum_{k=1}^{K}n_k/N\tag{12}$$

式中n_k表示正确划入第k簇的下近似中的样本的个数。聚类结果中的任意一个簇的下近似中,若其中含有k类别的样本数目最多,则认为该集合为第k类数据的分布。

Kappa系数公式如下

$$\text{Kappa}=\frac{N\sum_{k=1}^{K}n_k-\sum_{k=1}^{K}n_{k+}n_{+k}}{N^2-\sum_{k=1}^{K}n_{k+}n_{+k}} \tag{13}$$

式中:n_{k+}表示第k类真实的样本总数,n_{+k}表示第k类被分类的样本总数。

3.2 UCI数据集实验

采用UCI机器学习数据库中的3个经典数据集:Iris,Balance-scale和Wine数据集验证算法的性能,数据集具体信息如表1所示。实验环境为:Windows 7系统,i3处理器,2 GB内存,C语言编程。

表1 数据集信息

Tab. 1 Data sets information

数据集名称	包含样本点个数	样本维数	数据类数
Iris	150	4	3
Balance-scale	625	4	3
Wine	178	13	3

实验参数$\alpha=1,\beta=1,\rho=0.9,q=0.9,M=20,T=0.1$最大迭代次数$N_c=100,Q=100$。

为了验证本文算法的性能,分别采用粗糙K均值聚类算法、基于粒计算的粗糙集聚类算法[10]和本文提出的算法在所选UCI标准数据集上进行20次实验,然后取其平均值进行比较分析。在粗糙K均值聚类算法中ω_l和ω_b取值的不同会导致类心的变化,影响到聚类结果的准确率。当ω_l和ω_b取不同值时粗糙K均值聚类结果准确率如图1所示。下近似和边界区域的权重中的一个太大或者太小,都会造成类心的偏移,导致聚类结果准确率不高。因此在本次试验中粗糙K均值聚类算法中的下近似和边界区域权重的取值分别为$\omega_l=0.8$,$\omega_b=0.2$。

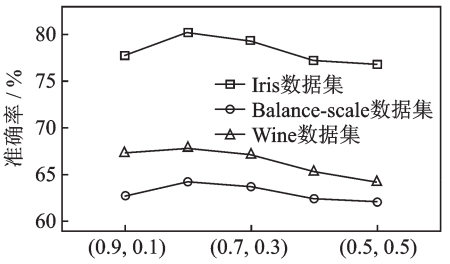

图1 ω_l和ω_b取不同值对应的聚类结果准确率

Fig.1 Accuracy rate of different ω_l and ω_b

3种算法在各数据集上聚类结果的准确率和Kappa系数如图2—4所示。3种算法在各个数据集上运行时间和迭代次数如图5,6所示。3种算法在各个数据集上运行后下近似中各类的类内距离如图7—9所示。

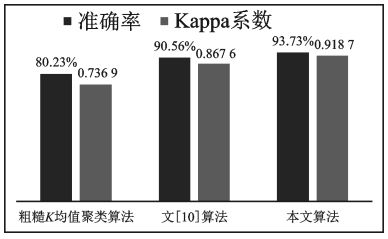

图2 Iris数据集准确率和Kappa系数

Fig.2 Accuracy rate and Kappa of Iris data sets

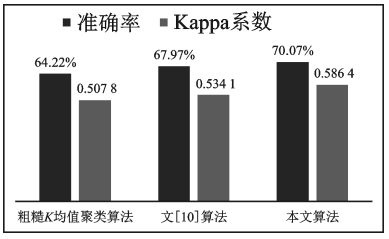

图3 Balance-scale数据集准确率和Kappa系数

Fig.3 Accuracy rate and Kappa of Balance-scale data sets

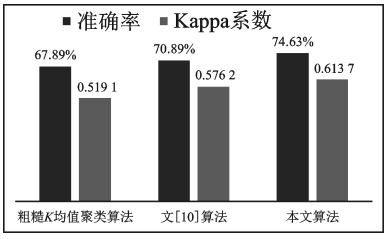

图4 Wine数据集准确率和Kappa系数

Fig.4 Accuracy rate and Kappa of Wine data sets

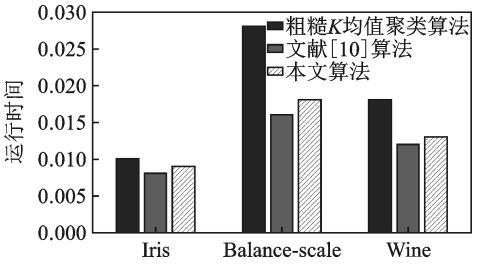

图5　3种算法运行时间

Fig.5　Running time of three algorithms

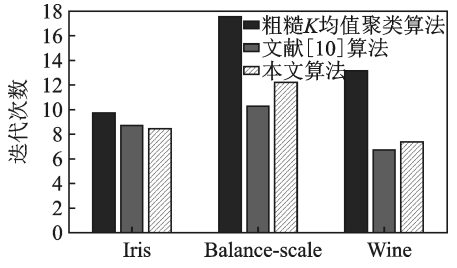

图6　3种算法运行次数

Fig.6　Running number of three algorithms

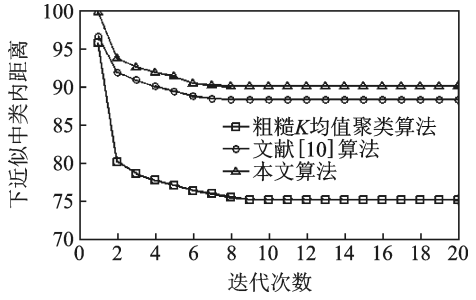

图7　Iris数据集收敛曲线

Fig.7　Convergence curves of Iris data sets

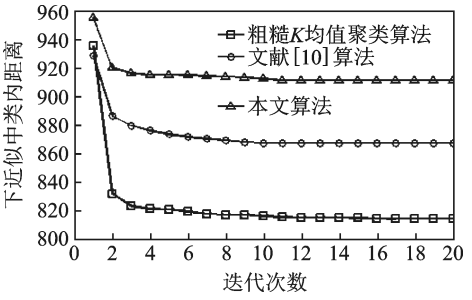

图8　Balance-scale数据集收敛曲线

Fig.8　Convergence curves of Balance-scale data sets

通过图2—4各项实验结果数据可知,本文算法在聚类准确率、Kappa系数方面有所改进。在Iris数据集上本文算法比粗糙K均值聚类算法与基于粒计算的粗糙集聚类算法的准确率分别提高了13.5%和3.17%,准确率效果显著,Kappa系数也得到了提高。在Balance-scale数据集和Wine数据集上,样本维数增加,本文算法的准确率与Kappa系数依然高于粗糙K均值聚类算法和基于粒计算的粗糙集聚类算法。通过图5和图6可知,本文算法与粗糙K均值聚类算法相比,运行时间和迭代次数都有所下降;本文算法与文献[10]中算法比较,虽然运行时间和迭代次数都略有增加,但是实验结果的准确率和Kappa系数都得到提升,因此为了提升实验结果增加迭代次数和运行时间也是值得的。下近似中各类的类间距离越大,即确定划分到该类的元素越多,算法的准确率越高。由图7—9可知,本文提出算法收敛的速度快于粗糙K均值聚类算法,与文献[10]算法相比收敛速度差别不大,但是准确率更高。由此可见,本文算法使用蚁群算法中随机概率选择策略增加解的多样性,使用信息素正负反馈机制促进算法收敛的同时跳出局部最优解,克服粗糙K均值聚类算法对初始点敏感的缺陷。同时根据数据集中数据的特点使用动态调整下近似权重,边界区域权重和阈值的方法,避免了人为设置参数对算法的影响,使得本文算法优于比较的其他算法。

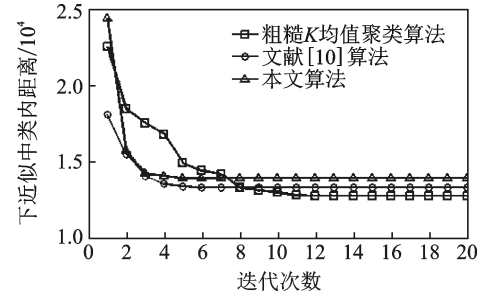

图9　Wine数据集收敛曲线

Fig.9　Convergence curves of Wine data sets

4　结束语

本文提出了一种结合蚁群算法的改进粗糙K均值聚类算法,该算法结合蚁群算法中的随机概率选择策略和正负反馈机制,优化粗糙K均值聚类算法对初始点敏感,导致聚类结果时常是局部最优而不是全局最优的缺陷,以及采用了动态调整下近似与边界区域的权重和算法阈值的方法,克服了人为设置参数对算

法的影响。实验证明这种方法,聚类结果的准确率和Kappa系数有很大提高。但该算法还存在着一些不足,如当有新的数据加入时需要在整个数据集上重新聚类。此项不足将是笔者下一步的研究工作重点。

参考文献:

[1] 孙权森, 纪则轩.基于模糊聚类的脑磁共振图像分割算法综述[J].数据采集与处理,2016,31(1): 28-42.
Sun Quansen, Ji Zexuan. Fuzzy clustering for brain MR image segmentation[J]. Journal of Data Acquisition and Processing, 2016, 31(1): 28-42.

[2] 田俊杰.基于功能磁共振数据的聚类研究[D].长沙:湖南师范大学,2015.
Tian Junjie. Cluster study based on functional magnetic resonance imaging data[D]. Changsha: Hunan Normal University, 2015.

[3] 张晓,张媛媛,高阳,等.一种基于密度的快速聚类方法[J].数据采集与处理,2015,30(4): 888-895.
Zhang Xiao, Zhang Yuanyuan, Gao Yang, et al. Fast density-based clustering approach[J]. Journal of Data Acquisition and Processing, 2015, 30(4): 888-895.

[4] Pawlak Z. Rough sets[J]. International Journal of Information and Computer Sciences, 1982, 11(5): 341-356.

[5] Lingras P, West C. Interval set clustering of web users with rough K-means[J]. Journal of Intelligent Information Systems, 2004, 23(1): 1635-1643.

[6] 谢娟英,张琰,谢维信,等.一种新的密度加权粗糙K-均值聚类算法[J].山东大学学报(理学版),2010,45(7): 1-6.
Xie Juanying, Zhang Yan, Xie Weixin, et al. A novel rough K-means clustering algorithm based on the weight of density[J]. Journal of Shangdong University (Natural Science), 2010, 45(7): 1-6.

[7] 郑超,苗夺谦,王睿智.基于密度加权的粗糙K均值聚类改进算法[J].计算机科学,2009,36(3): 220-222.
Zheng Chao, Miao Duoqian, Wang Ruizhi. Improved rough K-means clustering algorithm with weight based on density[J]. Computer Science, 2009, 36(3): 220-222.

[8] 周杨,苗夺谦,岳晓冬,等.基于自适应权重的粗糙K均值聚类算法[J].计算机科学,2011,38(6): 237-241.
Zhou Yang, Miao Duoqian, Yue Xiaodong, et al. Rough K-means clustering based on self-adaptive weights[J]. Computer Science, 2011, 38(6): 237-241.

[9] 洪亮亮,罗可.改进的基于遗传算法的粗糙聚类方法[J].计算机工程与应用,2010,46(25): 125-145.
Hong Liangliang, Luo Ke. Improved rough clustering method based on genetic algorithm[J]. Computer Engineering and Applications, 2010, 46(25): 125-145.

[10] 段文影,李向军,邱桃荣,等.一种具有自适应参数的基于密度加权的粗糙K-均值算法[J].南昌大学学报(理科版), 2012,36(5): 498-501.
Duan Wenying, Li Xiangjun, Qiu Taorong, et al. Rough K-means clustering algorithm with self-adaptive parameter and weighted-density[J]. Journal of Nanchang University (Natural Science), 2012, 36(5): 498-501.

[11] 李莲,罗可,周博翔,等.基于粒计算的粗糙集聚类算法[J].计算机应用研究,2013,30(10): 2916-2919.
Li Lian, Luo Ke, Zhou Boxiang, et al. Rough clustering algorithm based on granular computing[J]. Application Research of Coputers, 2013, 36(10): 2916-2919.

[12] 王超学,孔月萍,董丽丽,等.智能优化算法与应用[M].1版.西安:西北大学出版社,2012: 161-162.
Wang Chaoxue, Kong Yueping, Dong Lili, et al. Intellgent optimization algorithms and their applications[M]. 1st ed. Xi′an: Northwest University Press, 2012: 161-162.

[13] 杨柱天.基于机器学习的雷达辐射源分类识别技术研究[D].哈尔滨:哈尔滨工业大学,2012.
Yang Zhutian. Research on radar emitter classification and recognition based on machine learning[D]. Harbin: Harbin Institute of Technology, 2012.

[14] 李雪源,崔颖.基于二进制编码的烟花聚类算法[J].应用科技,2016,43(1): 36-39.
Li Xueyuan, Cui Ying. The binary encoding based fireworks clustering algorithm[J]. Applied Science and Technology, 2016, 43(1): 36-39.

作者简介:

刘洋(1990-),男,硕士研究生,研究方向:智能信息处理、数据挖掘,E-mail: Liuxauat@163.com。

王慧琴(1970-),女,教授,博士生导师,研究方向:数字图像处理、多媒体通信、数字建筑、信息安全,E-mail: hqwang@xauat.edu.cn。

张小红(1978-),女,博士研究生,讲师,研究方向:大数据分析、数据挖掘、机器学习、物联网等。

(编辑:张彤)