ISSN 1000-9825, CODEN RUXUEW
Journal of Software, Vol.21, No.10, October 2010, pp.2420−2431
doi: 10.3724/SP.J.1001.2010.03635

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

密度敏感的多智能体进化聚类算法*

潘晓英[1,2,3+], 刘 芳[4], 焦李成[2,3]

[1](西安邮电学院 计算机科学与技术系,陕西 西安 710061)

[2](西安电子科技大学 智能信息处理研究所,陕西 西安 710071)

[3](智能感知与图像理解教育部重点实验室,陕西 西安 710071)

[4](西安电子科技大学 计算机学院,陕西 西安 710071)

Density Sensitive Based Multi-Agent Evolutionary Clustering Algorithm

PAN Xiao-Ying[1,2,3+], LIU Fang[4], JIAO Li-Cheng[2,3]

[1](Department of Computer Science and Technology, Xi'an University of Post & Telecommunications, Xi'an 710061, China)

[2](Institute of Intelligent Information Processing, Xidian University, Xi'an 710071, China)

[3](Key Laboratory of Intelligent Perception and Image Understanding, Ministry of Education, Xi'an 710071, China)

[4](School of Computer Science and Engineering, Xidian University, Xi'an 710071, China)

+ Corresponding author: E-mail: xiaoying_pan@163.com

Pan XY, Liu F, Jiao LC. Density sensitive based multi-agent evolutionary clustering algorithm. *Journal of Software*, 2010,21(10):2420−2431. http://www.jos.org.cn/1000-9825/3635.htm

Abstract: By using the density sensitive distance as the similarity measurement, an algorithm of Density Sensitive based Multi-Agent Evolutionary Clustering (DSMAEC), based on multi-agent evolution, is proposed in this paper. DSMAEC designs a new connection based encoding, and the clustering results can be obtained by the process of decoding directly. It does not require the number of clusters to be known beforehand and overcomes the dependence of the domain knowledge. Aim at solving the clustering problem, three effective evolutionary operators are designed for competition, cooperation, and self-learning of an agent. Some experiments about artificial data, UCI data, and synthetic texture images are tested. These results show that DSMAEC can confirm the number of clusters automatically, tackle the data with different structures, and satisfy the diverse clustering request.

Key words: density sensitive distance; unsupervised clustering; multi-agent evolution; *k*-nearest neighbor mutation

摘 要: 采用密度敏感距离作为数据相似性度量,并基于多智能体进化的思想提出了一种密度敏感的多智能体进化聚类(density sensitive based multi-agent evolutionary clustering,简称 DSMAEC)算法.算法设计了一种基于连接的编码方式,通过解码过程可直接得到最终的聚类结果,无需事先确定聚类类别数,有效地克服了对领域知识的依赖.针对聚类问题,设计了 3 个有效的进化算子来模拟智能体间的竞争、合作和自学习行为,共同完成智能体的进化,最终达到对数据聚类的目的.分别对人工数据集、UCI 数据集以及合成纹理图像进行仿真,实验结果表明,该算法不但

* Supported by the National Natural Science Foundation of China under Grant Nos.60703107, 60703108 (国家自然科学基金); the National High-Tech Research and Development Plan of China under Grant No.2006AA01Z107 (国家高技术研究发展计划(863))

Received 2008-01-21; Accepted 2009-03-31

可以自动确定聚类类别数,而且能够应付不同结构的数据,适应不同的聚类要求,具有较强的实用价值.

关键词: 密度敏感距离;无监督聚类;多智能体进化;k 近邻变异

中图法分类号: TP18 **文献标识码:** A

聚类是一种重要的数据分析方法以及数据可视化的有效工具.给定一个数据集$\Gamma=\{x_1,x_2,\ldots,x_K\}\subset R^n$,聚类分析的目的在于根据某种相似性原则将$\Gamma$分成 C 个类别.聚类分析已经被广泛应用于计算机视觉、信息检索、数据挖掘和模式识别等领域.现有的聚类方法大致可以分为以下几种类型:基于划分的聚类算法[1]、层次聚类算法[2]、基于密度的聚类算法[3]、基于网格的聚类算法[4]、谱聚类算法[5].其中,基于划分的聚类算法由于将问题归结为一个优化问题,具有深厚的泛函基础,是聚类算法研究的重要分支之一,K 均值算法就是其中最典型的方法.该类算法的效率很高,但由于聚类目标函数是高度非线性和多峰的函数,因此标准的 K 均值算法在用梯度下降法优化目标函数时,搜索方向总是沿着能量减小的方向,算法易陷入局部最优点,只有当初始化较好时算法才能收敛到全局最优解.为了克服 K 均值算法中初始聚类中心对聚类结果的影响,Maulik 提出的遗传聚类算法(genetic algorithm-base clustering,简称 GAC)[6]通过进化的思想不断优化聚类中心,较好地克服了 K 均值算法易陷入局部最优的缺点.但与 KM(k-means)一样,GAC 仍采用了欧式距离作为相似性度量,这对于大多数具有特殊结构的聚类问题来说仍然无法得到满意的聚类结果.针对这一问题,王玲等人提出了一种新的距离测度——密度敏感距离测度方法[5,7],可以较好地体现聚类数据的空间分布特性;文献[8]将该距离测度应用到不同的聚类算法中,均取得了良好的聚类效果.但这些算法仍然无法摆脱该类算法需要事先输入聚类类别数的缺陷,给实际应用带来了一定的困难.另外,层次聚类算法一般以固定数目的点来表示一个聚类,提高了算法挖掘任意形状聚类的能力.基于密度的聚类算法的主要优点是能够发现任意形状的聚类和对噪音数据不敏感.谱聚类算法能够对高维数据进行聚类,且效率较高.因此,我们希望找到一种新的聚类算法,不但能够自动确定聚类类别数,可以对不同结构的数据进行聚类,而且能够推广到高维数据聚类.

鉴于多智能体进化所具有的快速收敛性和强大的优化能力[9],本文将其引入到无监督聚类问题中,提出了一种密度敏感的多智能体进化聚类算法.该方法在引入密度敏感距离测度和多智能体进化思想的基础上,设计了一种新的智能体编码方式以及有效的智能体进化算子,通过智能体间的竞争协作来实现数据聚类的目的.编码方式保证算法无需事先指定类别数,可在进化的过程中通过对智能体的解码提取出样本本身所含的结构,自动确定类别数.各进化算子协同作用,对各数据点之间的连接不断加以进化,达到对数据聚类的目的.

1 密度敏感距离及其多智能体进化模型

1.1 密度敏感距离

在现实世界的聚类问题中,数据的分布往往具有不可预期的复杂结构,导致了基于欧式距离的相似性度量无法反映聚类的全局一致性(即位于同一流形上的数据点具有较高的相似性).从图 1 所示的例子中可以形象地看出,我们期望数据点 1 与数据点 3 的相似性要比数据点 1 与数据点 2 的相似性更大,这样才有可能将数据点 1 和数据点 3 划分到同一类中.但是,在按照欧式距离进行相似性度量时,数据点 1 和数据点 2 的欧式距离要明显小于数据点 1 和数据点 3 之间的欧式距离,从而导致了数据点 1 与数据点 2 划分为同一类的概率要大于数据点 1 与数据点 3 划分为同一类的概率.也就是说,在用欧式距离作为相似性度量时,根本无法反映图 1 中所示数据的全局一致性.因此,对于现实世界中复杂的聚类问题,简单地采用欧式距离作为相似性度量会严重影响聚类算法的性能.

文献[5,7]提出了一种新空间一致性距离测度——密度敏感距离,该距离测度既能描述数据聚类的局部一致性特征,又能描述数据聚类的全局一致性特征,从而体现了聚类的空间分布特性.具体定义如下:

定义 1(密度可调节的线段长度). 空间中两点 x_i 和 x_j 之间的密度可调节线段长度 $L(x_i,x_j)$可按下式计算:

$$L(x_i,x_j)=\rho^{dist(x_i,x_j)}-1 \tag{1}$$

其中,$dist(x_i,x_j)$为数据点x_i与x_j之间的欧式距离,$\rho>1$为伸缩因子.由于满足聚类全局一致性的距离并不一定满足欧式测度下的三角不等式,也即满足该特性的距离能够使得两点间直接相连的路径长度不一定最短.很显然,密度可调节的线段长度即满足这一点,可用来描述聚类的全局一致性.

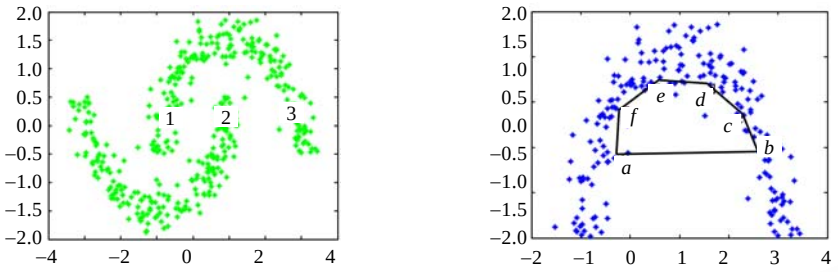


Fig.1 Euclidean distance can not reflect global consistency

图 1 欧式距离无法反映样本的全局一致性

定义 2(密度敏感距离测度). 将数据点看作是一个加权无向图 $G=(V;E)$的顶点 V,边集合 $E=\{W_{ij}\}$表示每一对数据点之间定义的相似度.令 $p\in V^l$ 表示图上一个长度为 $l=|p|-1$ 的连接点 p_1 与点 $p_{|p|}$的路径,其中边$(p_k,p_{k+1})\in E,1\leqslant k<|p|$.令 P_{ij}表示连接数据点x_i和x_j的所有路径的集合,则x_i和x_j之间的密度敏感距离测度 $D(x_i,x_j)$定义为

$$D(x_i,x_j)=\min_{p\in P_{i,j}}\sum_{k=1}^{|p|-1}L(p_k,p_{k+1}) \tag{2}$$

其中,$L(a,b)$表示 a,b 两点间密度可调节的线段长度.

密度敏感的距离测度可以度量沿着流形上的最短路径,这使得位于同一高密度区域内的两点可以用许多较短的边相连接,而位于不同高密度区域内的两点要用较长的边相连接,从而实现了放大位于不同高密度区域上数据点间的距离,而缩短位于同一高密度区域内数据点间距离的目的.

1.2 多智能体进化模型

在多智能体系统中,智能体 agent 是一个物理的或抽象的实体,它可作用于自身和环境,并能对环境做出反应.一般来说,agent 具有知识、目标以及通信能力、响应能力和协作能力.Agent 群体与环境之间相互作用、相互影响,群体内每个 agent 之间存在相互作用.因此,多智能体进化系统[9]的基本思想是:将传统进化算法中的每个个体形成智能体,每个智能体采用进化机制,能够同时与环境和其他智能体交换信息,互相影响彼此的进化过程,使各个智能体之间能够产生协作行为,最终形成各个智能体之间以及智能体与环境之间的共同适应.在多智能体进化模型中,将遵循以下的一些原则:每个 agent 都有初始能量;agent 具有局部性,其感知能力和行为只能针对有限的局部环境,即邻域;由于环境资源的有限性,agent 间存在着激烈的竞争,能量较低的 agent 将死亡,这一行为称为适者生存原则;由于 agent 死亡而空余出来的节点会由某些 agent 产生一个子 agent 来替代,这一行为称为弱肉强食.每个 agent 具有交配能力,agent 在其邻域内找到合适的配偶进行交配,把优良的基因传给下一代.另外,agent 具有知识,它可以利用知识进行启发式搜索,以提高自身的能量和对环境的适应能力.

2 密度敏感的多智能体进化聚类

利用多智能体进化的思想来解决聚类问题,不但需要解决智能体的编码问题,还需考虑智能体间如何进化以达到聚类的目的.

2.1 智能体定义

Maulik 提出的遗传聚类算法[6]将 K 个聚类中心编码为个体,对于 m 维的数据聚类问题,其编码长度为 $m\times K$,且这样的编码方式决定了该算法为一个连续空间的优化问题,且与维数相关,很难直接推广到高维数据聚类.

Gong 等人提出的进化聚类算法[8]采用了个体代表典型样本序号组合的编码方式,对于 K 类的聚类问题,个体长度为 K,且第 i 个基因位代表第 i 个类别的样本序号,该编码方式没有涉及数据的维数,搜索空间与数据维数无关,且为离散空间的优化问题,降低了搜索空间的大小.这两种编码方式虽然表示形式不同,但其实都是对 K 个聚类中心进行编码,典型样本序号组合的编码方式利用了已存在的数据点作为聚类中心.从本质上来说,个体进化的目的是为了确定一个较好的聚类中心,最终的类别判断仍依赖于 K 均值之类的传统算法.因此,这些算法仍需事先给定聚类类别数.

本文提出了一种基于连接的编码方式,智能体编码表示为 $a=(a_1,a_2,\ldots,a_N)$,其中 N 为参与聚类的数据点数目,a_i 表示和数据点 i 存在连接关系的点.在最终的聚类结果中,具有连接关系的数据点自动归为一类.当聚类数目不满足要求时,还可通过最小生成树断去较长的边,以获得满足要求的类别数.图 2 是该编码方式的一个简单例子(该例子包含了 8 个数据点).显然,这种编码方式无需事先给定一个确定的类别数目,可以在解码的过程中根据数据点的连接情况自动确定,减少了其对领域知识的依赖.

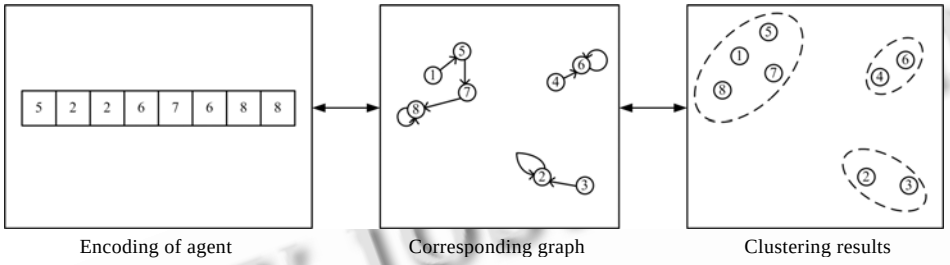


Fig.2 Connection based encoding diagram

图 2 基于连接的编码示意图

2.2 智能体能量

智能体的能量反映了智能体对问题的求解能力,针对聚类问题,智能体的能量为智能体所对应类别划分的目标函数值.完成解码过程后即可将所有样本数据划分到不同的类别中,则智能体的能量按公式(3)计算得到:

$$energy(a)=\frac{1}{1+\sum_{C_k\in C}\sum_{i\in C_k}D(i,\mu_k)} \tag{3}$$

其中,C 为智能体 a 对应的类别划分,μ_k 为 C_k 中心,$D(i,\mu_k)$为类别 C_k 中的第 i 个样本与聚类中心之间的密度敏感距离值.

2.3 智能体环境

在本文中,一个智能体代表了一种聚类结果.所有的智能体生存在规模为 $lat\times lat$ 的网格 *Lattice* 上,每个智能体占据一个格点且不能移动,智能体仅对其邻域具有局部感知能力,因此构成了如图 3 所示的网格结构.

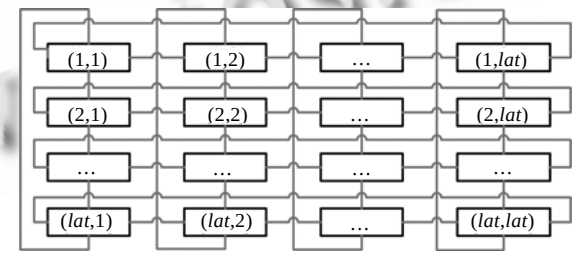


Fig.3 Sketch map of agent lattice

图 3 智能体网格结构示意图

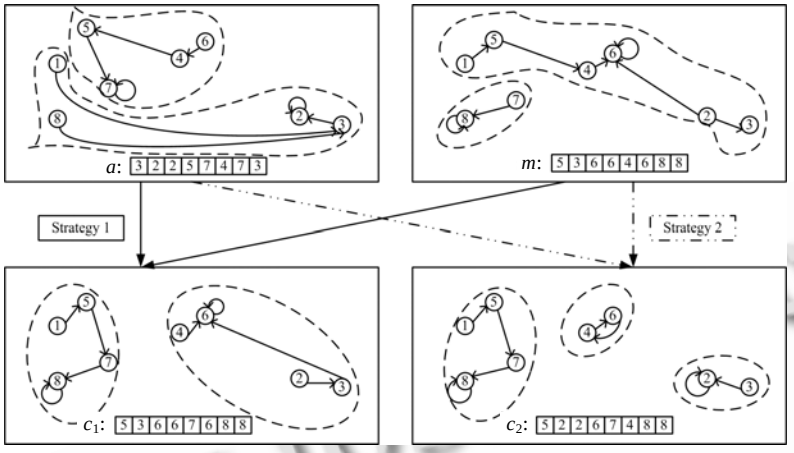
图 3 中,智能体的邻域包括其位置上相邻的 4 个智能体,$neighbor(i,j)=((i',j),(i,j'),(i'',j),(i,j''))$,其中,

$$i'=\begin{cases} i-1, & i\neq 1 \\ lat, & i=1 \end{cases},\ j'=\begin{cases} i-1, & j\neq 1 \\ lat, & j=1 \end{cases},\ i''=\begin{cases} i+1, & i\neq lat \\ 1, & i=lat \end{cases},\ j''=\begin{cases} i+1, & j\neq lat \\ 1, & j=lat \end{cases}.$$

2.4 智能体行为

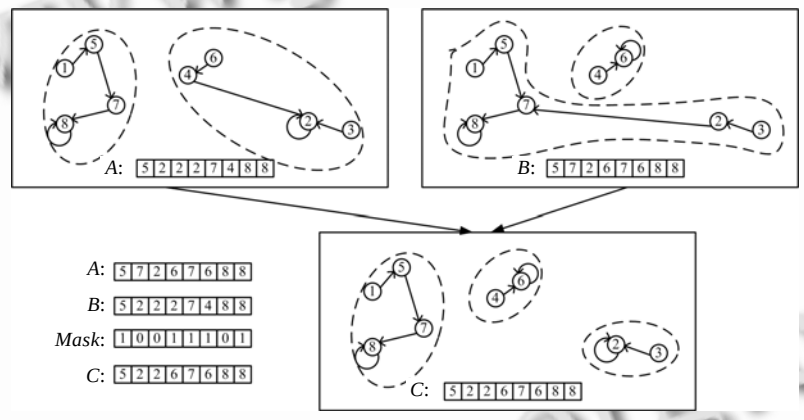
智能体具有一定的行为,以达到与环境和其他智能体交换信息、不断进化的目的.

针对聚类问题,设计了 3 种行为算子:竞争算子、协作算子和自学习算子(如图 4 所示).



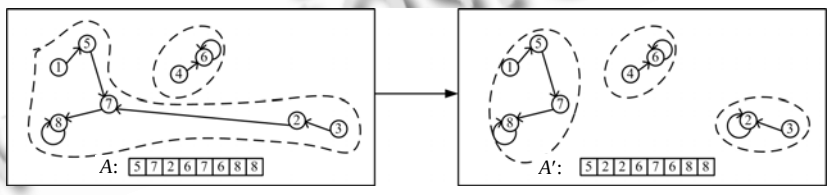
(a) Competition operator

(a) 竞争操作



(b) Cooperation operator

(b) 协作操作



(c) Self-Leaning operator

(c) 自学习操作

Fig.4 Schematic diagram of agent operation

图 4 智能体操作示意图

图 4 中:竞争算子和协作算子作用在 agent 与其局部环境中的 agent 之间,实现 agent 的竞争和协作行为;自学习算子作用在单个 agent 之上,实现 agent 利用自身知识的学习过程.

2.4.1 邻域竞争操作

竞争算子的作用是剔除网格上能量较低的智能体,提高整体的能量水平.假设执行竞争操作的智能体为 $a=(a_1,a_2,\ldots,a_N)$,其局部环境中能量最高的智能体为 $m=(m_1,m_2,\ldots,m_N)$,若 $energy(a)<energy(m)$,则执行竞争操作,产生新的智能体 $new=(e_1,e_2,\ldots,e_N)$.令 $D(i,a_i)=al_i,D(i,m_i)=ml_i,i=1,2,\ldots,N$,即 a 中各连接长度分别为$(al_1,al_2,\ldots,al_N)$,m 中各连接长度分别为$(ml_1,ml_2,\ldots,ml_N)$,则其竞争操作可采取两种策略,分别描述如下:

策略 1. 令 $new=m$,并找出 m 中长度最长的连接 $ml_i=\max(ml_1,ml_2,\ldots,ml_N)$,令 $e_i=a_i$.

策略 2. 令 $new=m$,从 a 中选取长度较短的 s 个连接,其位置为 $pos=(p_1,p_2,\ldots,p_s)$,则 $new(pos)=a(pos)$.

图 4(a)为智能体竞争操作的示意图,假设 $a=(3,2,2,5,7,4,7,3)$,$m=(5,3,6,6,4,6,8,8)$.策略 1 中,在 m 找到长度最长的连接 ml_s,并以 a 中第 5 位的连接点 7 替代 m 中原有的 4 时,即得到 $c_1=(5,3,6,6,7,6,8,8)$,其聚类结果也相应改变;策略 2 中,保留了智能体 a 中连接最短的 4 条边 $c_2=(\times,2,2,\times,7,4,\times,\times)$,标×的位置由 m 中对应的连接所代替,最终可得 $c_2=(5,2,2,6,7,4,8,8)$,其聚类结果也由原来的 2 类变成了 3 类.

2.4.2 邻域协作操作

假设参与协作的两个智能体分别为 $a=(a_1,a_2,\ldots,a_N)$,$b=(b_1,b_2,\ldots,b_N)$,随机产生一包含 0,1 位串的屏蔽向量 $Mask$,长度与智能体的长度相同,若 $Mask$ 为 1,则继承 a 中的连接,否则,继承 b 中的连接,其协作操作的例子如图 4(b)所示.图中参与协作的智能体分别为 $A=(5,2,2,2,7,4,8,8)$ 和 $B=(5,7,2,6,7,6,8,8)$,产生的屏蔽向量为 $Mask=(1,0,0,1,1,1,0,1)$,则最终产生的新智能体为 $C=(5,2,2,6,7,6,8,8)$,其聚类结果也发生了相应的改变.

2.4.3 自学习操作

智能体除了会与邻域内的智能体发生竞争协作操作之外,还可通过自身知识进行自学习操作,这里所采用的是 k 近邻变异法.以概率 Pm 随机选择智能体 $a=(a_1,a_2,\ldots,a_N)$上的某一基因位 a_i,在离 i 最近的 k 个数据点中任意选择一个来替代原有的 a_i.如图 4(c)所示,参与自学习的智能体为(5,7,2,6,7,6,8,8),选择第 2 个基因位进行自学习,以 2 来替代原有的 7,则自学习后的智能体为(5,2,2,6,7,6,8,8),其聚类结果即从 2 类变为 3 类.

2.5 密度敏感的多智能体进化聚类算法DSMAEC(density sensitive based multi-agent evolutionary clustering)

假设进行聚类的数据点数目为 N,智能体种群的规模为 $lat\times lat$,$Lattice^t$ 为第 t 代的智能体种群,交叉概率为 Pc,变异概率为 Pm,占据概率为 Po,最大迭代次数为 $maxgen$.

输入:智能体种群 $Lattice$,参数 Pc,Pm 和 Po;

输出:智能体网格 $Lattice$.

Step 1. 初始化智能体种群 $Lattice^1$,所有参数初始化,令 $t\leftarrow 1$;

Step 2. 根据智能体编码对数据点进行划分,并根据公式(3)计算各智能体的能量;

Step 3. 对种群 $Lattice^t$ 中的每个智能体以占据概率 Po 执行邻域竞争操作,得到 $Lattice^{t+1/3}$;

Step 4. 对 $Lattice^{t+1/3}$ 中的每个智能体以概率 Pc 执行邻域协作操作,得到 $Lattice^{t+2/3}$;

Step 5. 对 $Lattice^{t+2/3}$ 中的每个智能体,若 $U(0,1)<Pm$,则执行自学习操作,得到 $Lattice^{t+1}$;

Step 6. 令 $t=t+1$,若 $t\leqslant maxgen$,则转 Step 2,否则转 Step 7;

Step 7. 从 $Lattice^t$ 中选出合适的智能体进行解码,即得到聚类结果.

3 实验分析

为了考察密度敏感的多智能体进化聚类算法(DSMAEC)的性能,我们将其对人工数据集、UCI 数据集以及合成纹理图像分别进行了测试.同时,将该算法与原始的 K-均值算法(KM)、遗传聚类算法(GAC)[6]以及基于流形距离的进化聚类算法(density sensitive evolutionary clustering,简称 DSEC)[8]进行了性能的比较,各算法的参数设置见表 1.

对于算法的聚类性能,除了用可以聚类的错误率来表示之外,还可采用指标 Adjusted Rand Index 来衡量,它

将类别划分看作是样本之间的一种关系,每一对样本要么被划分在同一类,要么在不同类,通过统计正确决策对数来评价聚类算法的性能.对于一个有 n 个样本的数据集,Adjusted Rand Index 可以按照以下公式来计算:

$$R(U,V)=\frac{\sum_{lk}\binom{n_{lk}}{2}-\left[\sum_{l}\binom{n_{l\bullet}}{2}\bullet\sum_{k}\binom{n_{\bullet k}}{2}\right]\Big/\binom{n}{2}}{\frac{1}{2}\left[\sum_{l}\binom{n_{l\bullet}}{2}+\sum_{k}\binom{n_{\bullet k}}{2}\right]-\left[\sum_{l}\binom{n_{l\bullet}}{2}\bullet\sum_{k}\binom{n_{\bullet k}}{2}\right]\Big/\binom{n}{2}} \tag{4}$$

其中,n_{lk} 表示被划分到类属 l 和类属 k 的样本的个数.$R(U,V)\in[0,1]$,其数值越大,说明聚类效果越好.

Table 1 Parameter settings for KM, GAC, DSEC and DSMAEC

表 1 算法 KM,GAC,DSEC 以及 DSMAEC 的参数设置

Parameter	KM	GAC	DSEC	DSMAEC
Max generation	500	100	100	100
Population size	/	50	50	25
Cross probability	/	0.8	0.8	0.8
Mutation probability	/	0.1	0.1	0.1
Occupy probability	/	/	/	0.2
Number of clustering	Pre-Defined	Pre-Defined	Pre-Defined	/

3.1 人工数据集

首先以 6 个人工数据集的聚类问题对算法 DSMAEC 进行测试.该 6 个数据集分别为 two-moons,four-lines, multi-scales,squiggles,three-circles 以及 four-squares,它们具有不同的结构(如图 5 所示),可以用来考察算法对不同结构数据的聚类性能.

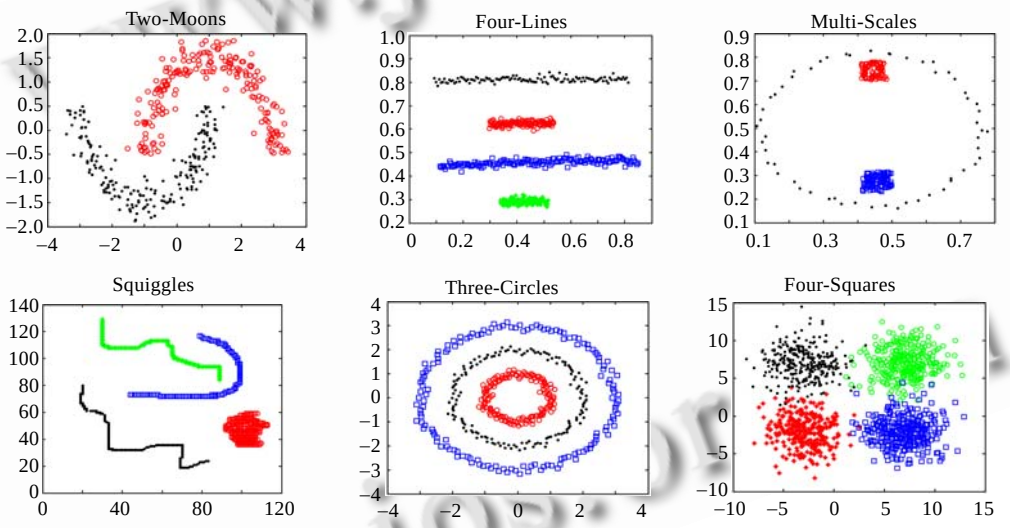


Fig.5 Sketch map of artificial datasets

图 5 人工数据集结构图

我们将算法对每个数据集分别独立运行 30 次,所得到的 Clustering Error 和 Adjusted Rand Index 两项指标的平均值见表 2.

表 2 中的客观聚类结果表明,DSMAEC 具有良好的聚类性能.在所测试的 6 个人工数据集中,DSMAEC 对其中的 4 个数据集达到了相对最好的聚类效果,对于其中的 four-lines 和 three-circles 数据集能够完全准确地聚类,对其他 4 个数据集的聚类错误率也保留在一个较低的水平.相对于同样采用密度敏感距离测度的 DSEC 来说,两者的性能比较接近,但 DSMAEC 的聚类错误率能够更小一些,而且无需事先确定聚类类别数.我们发现,对

于流形结构较为明显的前 5 个数据集,KM 和 GAC 的性能比较差,这是由于采用欧氏距离作为相似性度量所导致的.相对地,采用密度敏感距离作为相似性度量的两种算法 DSEC 和 DSMAEC 都能取得较好的聚类结果.另外,我们在设定算法参数时,DSEC 和 DSMAEC 的参数基本一致,但 DSMAEC 的种群规模仅为 DSEC 的一半,且聚类的数目是事先未给定的.

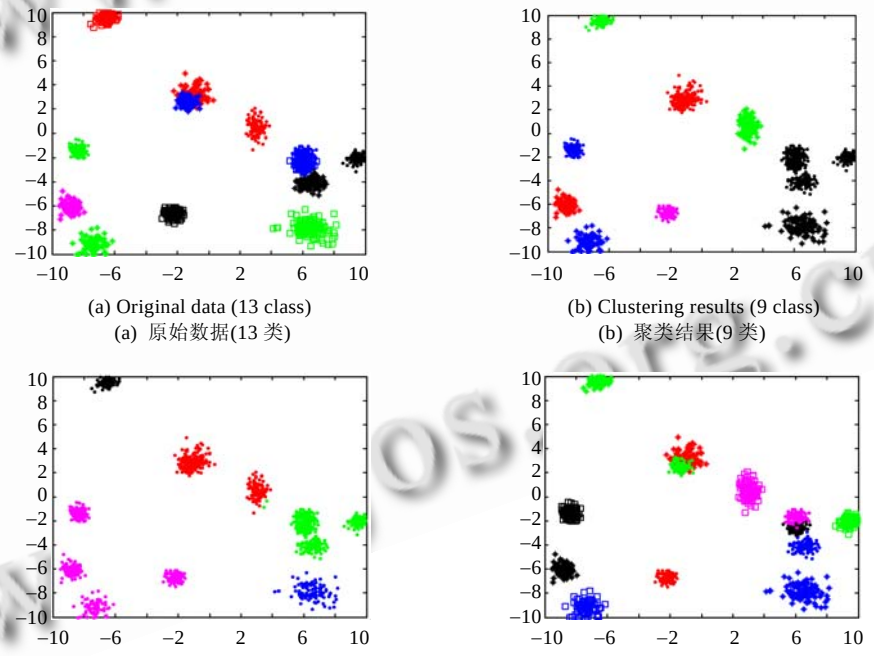
Table 2 Comparative results of four different algorithms for artificial datasets

表 2 4 种算法对人工数据集的聚类结果比较

Problem	Clustering error				Adjusted rand index			
	KM	GAC	DSEC	DSMAEC	KM	GAC	DSEC	DSMAEC
Two-Moons	0.282 5	0.274 3	0.017 4	**0.013 8**	0.415 3	0.457 6	0.921 4	**0.936 5**
Four-Lines	0.244 1	0.212 3	**0**	**0**	0.502 5	0.534 1	**1**	**1**
Multi-Scales	0.176 5	0.125 4	**0.042 7**	0.056 2	0.725 8	0.786 2	**0.895 2**	0.854 1
Squiggles	0.314 6	0.301 7	0.057 6	**0.038 1**	0.284 5	0.302 1	0.849 6	**0.901 7**
Three-Circles	0.331 4	0.312 8	**0**	**0**	0.218 5	0.285 4	**1**	**1**
Four-Squares	0.068 8	**0.061 2**	0.064 5	0.070 4	0.836 5	**0.897 3**	0.875 6	0.814 1

KM,GAC 以及 DSEC 从本质上来说都是 K 均值聚类算法,KM 仅在初始聚类中心较好时才能达到理想的聚类结果;GAC 与 DSEC 均从优化聚类中心的角度对 KM 进行改进,进一步改善聚类结果.对于一般聚类问题,GAC 可以达到较好的聚类效果,但对于具有较强流形形状的聚类问题仍无法正确求解;DSEC 改进了距离测度,因此对各种类型的聚类问题都可达到较好的效果,但它本质上仍采用 K 均值算法对数据点进行划分,无法对未知类别数的聚类问题进行自动聚类.本文算法 DSMAEC 从另外一个角度出发,采用连接的编码方式,通过对点连接的优化达到聚类目的,其聚类结果可通过解码过程直接得到,从而克服了其他算法需事先确定聚类数目的缺点.

从其设计思想出发,算法还应具备能够同时发现不同类型聚类的能力.因此,我们通过人工产生一个数据集来对算法进行测试,其参数设置同上,取其一次运行所得到的多个不同聚类结果,如图 6 所示.其中,图 6(a)为原图,图 6(b)~图 6(d)分别为一次运行所得到的 3 种不同聚类结果.



(a) Original data (13 class) (a) 原始数据(13 类)

(b) Clustering results (9 class) (b) 聚类结果(9 类)

(c) Clustering results (5 class) (c) 聚类结果(5 类)

(d) Clustering results (13 class) (d) 聚类结果(13 类)

Fig.6 Some different clustering results by one runs

图 6 一次运行所得到的多种不同聚类结果

对于如图 6(a)所示的数据集,通过 DSMAEC 一次运行可得到 3 种不同类别数的聚类结果,分别为 5 类、9 类和 13 类.也就是说,算法具备发现不同类型数据聚类的能力.在该测试集中,有几类数据距离很近,且存在某些重叠现象.当类别数较少时,这种数据会聚集到同一类中;而当类别数较多时,又会分开聚集到不同类别中.从直观上来看,这几种聚类结果均在能够接受的范围之内,但如图 6(d)所示的结果更接近于原始的类别分布.该例旨在说明算法能够同时适应不同的聚类要求,给出不同的聚类结果.

3.2 UCI数据集

为进一步测试算法 DSMAEC 的性能,我们从 UCI 数据集中选取了 3 个数据集(breast cancer,iris,glass)对其进行测试.其中,breast cancer 包括了 286 个数据,分为 2 类,分别包含 201 和 85 个数据,每个数据包含 9 个属性;iris 数据集包含 150 个数据,分为 3 类,每类 50 个数据,每个数据包含 4 个属性;glass 数据集包含 214 个数据,分为 6 类,每个数据包含 9 个属性.与人工数据集一样,我们采用 4 种算法对这 3 个数据集分别求解,对所涉及的参数同表 1 中设置,对 KM,GAC 和 DSEC 这 3 种算法事先指定聚类类别数,DSMAEC 则通过算法自动确定类别数.表 3 为 30 次独立运行的平均结果.

Table 3 Clustering results of four algorithms for UCI datasets

表 3 4 种算法对 UCI 数据集的聚类结果比较

Problem	Clustering error				Adjusted rand index			
	KM	GAC	DSEC	DSMAEC	KM	GAC	DSEC	DSMAEC
Breast cancer	0.436 8	0.396 4	0.038 2	**0.010 6**	0.098 4	0.132 5	0.896 5	**0.925 4**
Iris	0.106 7	0.096 7	0.013 7	**0.006 4**	0.804 4	0.831 2	0.912 7	**0.973 5**
Glass	0.523 4	0.482 5	0.102 1	**0.081 8**	0.032 4	0.084 2	0.810 6	**0.854 1**

从对 UCI 标准数据集中的聚类结果来看,DSMAEC 具有明显优于其他算法的性能,其聚类错误率 Clustering Error 全部小于其他算法所得到的结果,聚类指标 Adjusted Rand Index 要高于其他算法.我们所测试的 3 个 UCI 数据集的维数分别为 9 维、4 维、9 维,聚类类别数分别为 2 类、3 类、6 类,除数据集 iris 外,其他两个数据集中各类别的数目相差较大.可以发现,对于高维的且分布不均匀的现实数据聚类,K 均值的性能较差,其聚类错误率大约在 50%左右,而本文算法 DSMAEC 较好地克服了 K 均值算法的固有缺陷,具有较强的分类性能.更为重要的是,DSMAEC 可对未知类别的数据进行聚类,且可得到较优的结果.

3.3 合成纹理图像

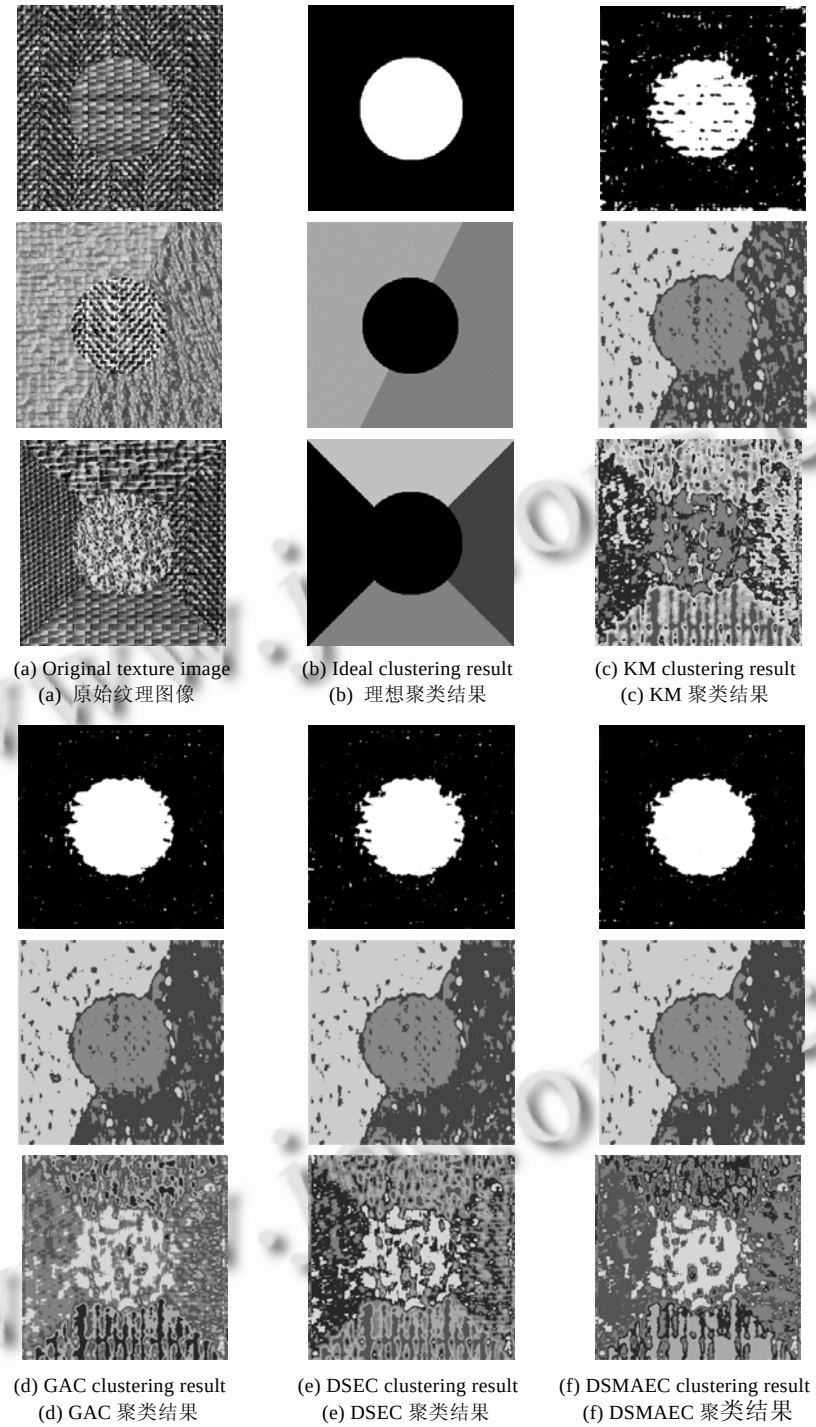
除了以人工数据集和 UCI 数据集对算法进行验证以外,我们还对合成纹理图像进行了测试.原始合成纹理图像如图 7(a)所示,对于图像中的每个像素,依据文献[13]对图像提取基于灰度共生矩阵的 3 维特征,4 种算法的参数设置仍同表 1 中设置.图 7(b)~图 7(f)按顺序分别为理想聚类结果、KM 聚类结果、GAC 聚类结果、DSEC 聚类结果和 DSMAEC 所得聚类结果.

由图 7 可以看出,对于前两个纹理图像,4 种算法都达到了较好的聚类效果,除 KM 算法所得到的图像中存在相对较多的错聚点外,其他 3 种算法的结果差别不大.对于第 3 个纹理图像,由于其所包含的纹理较多且比较复杂,4 种算法的聚类效果都不甚理想.但从图像的直观结果来看,DSMAEC 所得结果相对要好一些.另外,为进一步直观地获取各种算法对合成纹理图像的聚类性能,表 4 给出了 4 种算法对每个纹理图像分别独立运行 30 次所得到的平均结果.

由表 4 可见,4 种算法对图像 1(2 类)和图像 2(3 类)都取得了比较理想的结果,错误率非常低;而对于纹理较复杂的第 3 幅图像,4 种算法的性能都不够理想,其错误率均在 30%左右.但相对于其他 3 种算法,DSMAEC 的性能是最好的,其平均错误率为 21.47%.人工目视和客观评价指标的结果是类似的.

综合以上结果可知,密度敏感的多智能体进化聚类算法可对合成纹理图像进行有效的聚类,但并不存在较大的优势.算法设计的出发点是采用基于连接的编码方式和密度敏感距离测度,而纹理图像并不存在很强的流形形状,因而很难体现密度敏感距离测度的优势.另外,采用基于连接的编码方式能够自动确定聚类类别数,消除对领域知识的依赖,而我们在测试时对其他算法都指定了类别数,同样无法体现该算法的这一优点.因此,与

其他算法相比,能够在结果中体现的仅为多智能体进化的思想,说明在进化算法中结合多智能体协作的思想能够在一定程度上增强算法的优化能力.



(a) Original texture image (a) 原始纹理图像
(b) Ideal clustering result (b) 理想聚类结果
(c) KM clustering result (c) KM 聚类结果
(d) GAC clustering result (d) GAC 聚类结果
(e) DSEC clustering result (e) DSEC 聚类结果
(f) DSMAEC clustering result (f) DSMAEC 聚类结果

Fig.7 Clustering results for synthetic texture images

图 7 合成纹理图像聚类

Table 4 Clustering results of four algorithms for texture images

表 4 4 种算法对合成纹理图像的聚类结果

Texture images	Clustering error				Adjusted rand index			
	KM	GAC	DSEC	DSMAEC	KM	GAC	DSEC	DSMAEC
Image 1	0.123 7	0.094 6	0.098 3	**0.091 4**	0.745 8	0.825 5	0.815 3	**0.832 7**
Image 2	0.154 6	0.113 9	0.112 8	**0.110 8**	0.756 0	0.776 3	0.779 1	**0.782 4**
Image 3	0.325 9	0.295 6	0.259 4	**0.214 7**	0.281 4	0.401 3	0.464 1	**0.527 6**

3.4 算法性能分析

结合算法思想及 3 种不同数据集的测试结果,我们从以下几个方面对算法 DSMAEC 的性能进行简要分析:

1) 能够应付不同结构的数据.算法采用了密度敏感距离作为样本的相似性度量,很好地克服了欧式距离的局限性,能够反映数据的真实结构,因此对不同结构的数据都能达到良好的聚类效果;
2) 能够发现不同类型的聚类.算法采用了基于连接的编码方式,聚类结果可直接通过解码得到,且一次运行即可得到多个不同的聚类结果;
3) 对专业知识的要求降到最低.密度敏感的多智能体进化聚类无需事先指定类别数,而是在进化的过程中通过对智能体的解码提取出样本本身所含的结构,自动确定类别数,从而避免了算法受预先设定类别数的影响,减少了对领域知识的依赖性;
4) 对于数据的不同顺序不敏感.不同于一般的聚类算法,输入数据的顺序将不会对 DSMAEC 产生影响;
5) 时间复杂度.设数据集的规模为 N,计算数据点间距离的时间复杂度为 $O(N\times N)$,3 个进化操作(竞争操作、协作操作和变异操作)的时间复杂度均为 $O(N)$,解码分类过程的时间复杂度为 $O(N)$,因此,总的算法时间复杂度为 $O(N^2)$;
6) 可适用于高维空间的数据集.算法中所设计的编码方式与维数无关,只与参加聚类的数据点数有关,因此可直接推广到高维空间的数据聚类问题.

4 结 论

本文提出了一种密度敏感距离的多智能体进化聚类算法,通过设计一种基于连接的编码方式和 3 个有效的多智能体进化算子,充分利用了多智能体协作竞争的特点,快速将信息扩散到整个智能体种群当中,完成整个进化聚类过程.该算法具有良好的聚类性能,具有以下几方面的特点:

(1) 采用基于连接的编码方式,与数据维数无关,可以推广到高维数据聚类;
(2) 无需事先给定聚类类别数,其聚类类别在解码过程中自动确定;
(3) 采用密度敏感距离测度进行数据点间相似性的度量,能够更好地反映数据点的全局分布特性;
(4) 所设计的 3 个进化算子从不同方面促进智能体种群的进化,以达到最终聚类的目的;
(5) 一次运行可得到多种聚类结果,可同时满足不同的聚类需求.

对 3 种数据集的仿真实验结果表明,该方法具有良好的性能,能够应付不同结构的数据聚类以及不同的聚类需求,具有较强的实用价值.

References:

[1] Shen HB, Wang ST. Fuzzy kernel clustering with outliers. Journal of Software, 2004,15(7):1021−1029 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/15/1021.htm

[2] Seo J, Shneiderman B. Interactively exploring hierarchical clustering results. IEEE Computer, 2002,35(7):80−86.

[3] Sander J, Ester M, Kriegel H. Density-Based clustering in spatial databases: The algorithm GDBSCAN and its applications. Data Mining and Knowledge Discovery, 1998,2(2):169−194. [doi: 10.1023/A:1009745219419]

[4] Park NH, Lee WS. Statistical grid-based clustering over data streams. ACM SIGMOD Record, 2004,33(1):32−37. [doi: 10.1145/974121.974127]

[5] Wang L, Bo LF, Jiao LC. Density-Sensitive spectral clustering. ACTA Electronica Sinica, 2007,35(8):1577−1581 (in Chinese with English abstract).

[6] Maulik U, Bandyopadhyay S. Genetic algorithm-based clustering technique. Pattern Recognition, 2000,33(9):1455−1465. [doi: 10.1016/S0031-3203(99)00137-5]

[7] Wang L, Bo LF, Jiao LC. Density-Sensitive semi-supervised spectral clustering. Journal of Software, 2007,18(10):2412−2422 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/18/2412.htm [doi:10.1360/jos182412]

[8] Gong MG, Jiao LC, Wang L, Bo LF. Density-Sensitive evolutionary clustering. In: Proc. of the 11th Pacific-Asia Conf. on Knowledge Discovery and Data Mining. LNCS 4426, Nanjing: Springer-Verlag, 2007. 507−514.

[9] Zhong WC, Liu J, Xue MZ, Jiao LC. A multiagent genetic algorithm for global numerical optimization. IEEE Trans. on Systems, Man and Cybernetics, 2004,34(2):1128−1141. [doi:10.1109/TSMCB.2003.821456]

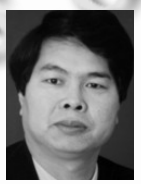
附中文参考文献:

[1] 沈红斌,王士同.离群模糊核聚类算法.软件学报,2004,15(7):1021−1029. http://www.jos.org.cn/1000-9825/15/1021.htm

[5] 王玲,薄列峰,焦李成.密度敏感的谱聚类.电子学报,2007,35(8):1577−1581.

[7] 王玲,薄列峰,焦李成.密度敏感的半监督谱聚类.软件学报,2007,18(10):2412−2422. http://www.jos.org.cn/1000-9825/18/2412.htm [doi:10.1360/jos182412]

潘晓英(1981－),女,浙江缙云人,博士,讲师,主要研究领域为智能信息处理,数据挖掘.



焦李成(1959－),男,博士,教授,博士生导师,CCF 高级会员,主要研究领域为智能算法,机器学习,非线性科学,小波理论及其应用.

刘芳(1963－),女,博士,教授,博士生导师,CCF 高级会员,主要研究领域为智能信息处理,模式识别.