第 26 卷第 10 期
2009 年 10 月

控 制 理 论 与 应 用
Control Theory & Applications

Vol. 26 No. 10
Oct. 2009

文章编号: 1000−8152(2009)10−1175−05

基于微粒群优化聚类数目的K–均值算法

巩敦卫[1], 蒋余庆[1], 张　勇[1], 周　勇[2]

(1. 中国矿业大学 信息与电气工程学院, 江苏 徐州 221008; 2. 中国矿业大学 计算机科学与技术学院, 江苏 徐州 221008)

摘要: K–均值算法是广泛使用的聚类算法, 但该算法的聚类数目难以确定, 且聚类结果对初始聚类中心比较敏感. 本文提出一种基于微粒群优化聚类数目的K–均值算法, 该算法采用聚类中心的坐标和通配符表示微粒位置, 通过定义微粒更新公式中新的加减运算符, 动态调整聚类中心的数目及坐标, 此外, 以改进的聚类有效性指标Davies-Bouldin准则作为适应度函数. 5个人工和真实数据集的聚类结果验证了所提算法的优越性.

关键词: 聚类; K–均值算法; 微粒群优化; 微粒更新

中图分类号: TP181; TP301　　**文献标识码**: A

K-mean algorithm for optimizing the number of clusters based on particle swarm optimization

GONG Dun-wei[1], JIANG Yu-qing[1], ZHANG Yong[1], ZHOU Yong[2]

(1. School of Information and Electrical Engineering, China University of Mining and Technology, Xuzhou Jiangsu 221008, China;
2. School of Computer Science and Technology, China University of Mining and Technology, Xuzhou Jiangsu 221008, China)

Abstract: K-mean algorithm is a widely used clustering method, but it is difficult to determine the number of clusters; and the clustering result is sensitive to initial cluster centers. We present a novel K-mean algorithm for optimizing the number of clusters based on particle swarm optimization. The algorithm denotes the position of a particle with the coordinates of cluster centers and wildcards. The coordinates of cluster centers are dynamically adjusted by defining the new plus and new minus operators in the particle update formula. In addition, an improved Davies-Bouldin index is employed to evaluate the efficiency of a clustering result. Experimental results of 5 sets of artificial and real-world data validate the advantages of the proposed algorithm.

Key words: clustering; K-means algorithm; particle swarm optimization; particle update

1　引言(Introduction)

聚类分析是模式识别、机器学习和数据挖掘等领域的非常重要的研究课题, 具有广阔的应用前景. 目前, 有关聚类算法的研究已经取得了丰硕的成果. 根据对象间的相似性度量和聚类评价准则等的不同, 聚类方法一般包括:划分的方法、层次的方法、基于密度的方法、基于网格的方法, 以及基于模型的方法等[1], 其中, K–均值算法作为一种经典的划分方法, 因其简单易行、快速有效, 而受到众多研究者的青睐. 但是, 该算法的聚类数目通常作为先验知识由用户给定. 实际上, 当数据量非常大或者先验知识难以获取时, 用户很难确定准确的聚类数目. 因而, 根据实际的聚类问题, 采用合适的方法优化聚类数目是十分必要的. 此外, K–均值算法对初始聚类中心十分敏感, 这使得对于相同的数据集, 不同的初始聚类中心有可能得到完全不同的聚类结果. 这些缺陷大大限制了K–均值算法在实际问题中的应用.

微粒群优化(particle swarm optimization, PSO)算法是一种基于群体智能的随机全局优化技术, 由于该算法收敛速度快, 需要设定的参数少, 且实现简单, 因而, 近年来得到学术界的广泛关注[2]. 最近, 研究人员结合PSO的全局优化性能, 提出了多种行之有效的微粒群优化聚类算法[3~6], 大大降低了聚类结果对初始聚类中心的敏感度. 但是, 这些算法仍然没有解决聚类数目的优化问题.

能否自动确定聚类数目, 是评判聚类算法是否优秀的重要标准之一. 鉴于以上分析, 本文提出一种基于微粒群优化聚类数目的K–均值算法, 对人工和真实数据集的聚类结果验证了所提算法的有效性.

收稿日期: 2007−10−22; 收修改稿日期: 2009−01−04.
基金项目: 江苏省自然科学基金资助项目(BK2008125); 教育部新世纪优秀人才支持计划资助项目(NCET–07–0802).

2 相关工作(Related work)

2.1 问题描述(Description of problems)

设$O = \{o_1, o_2, \cdots, o_S\} \subset \mathbb{R}^d$为待聚类数据集, $o_i = (o_{i1}, o_{i2}, \cdots, o_{id})$为$O$中的一个样本或特征向量. O的聚类问题就是寻找一个满足性能指标J的最优划分$C = \{C_1, C_2, \cdots, C_K\}, 2 \leqslant K \leqslant S$, 使得

$$\begin{cases} \bigcup\limits_{i=1}^{K} C_i = O, \\ C_i \neq \emptyset,\ i = 1, 2, \cdots, K, \\ C_i \cap C_j = \emptyset,\ i, j = 1, 2, \cdots, K,\ i \neq j, \end{cases} \tag{1}$$

J为某一聚类有效性指标, 如总的误差平方和准则、Davies-Bouldin(DB)准则[1]等.

2.2 K–均值算法及其研究现状(K-means algorithm and state-of-the-art)

K–均值算法于1967年由McQueen提出, 该算法的思想是: 基于总的类内误差平方和最小准则, 随机初始化K个聚类中心, 通过迭代搜索K个最优的聚类中心. K–均值算法的聚类结果对初始聚类中心比较敏感. 面向解决该问题, 研究人员提出了很多改进方法, 如遗传优化聚类算法[7]、自适应进化聚类算法[8], 以及谱方法[9]等. 到目前为止, 优化聚类数目的方法并不多见. 李永森等对空间聚类算法中的聚类数目优化问题进行了初步研究, 提出了距离代价函数, 证明了当该函数达到最小值时, 对应的聚类数目即为最佳的[10]. 但是, 该方法的复杂度随着聚类数目的增大也将大大增加. 此外, Bandyopadhyay等将聚类数目融入到染色体编码中, 提出聚类数目自动进化的遗传优化聚类算法[11], 该算法虽然能够自动确定最佳的聚类数目, 但效率有待进一步提高.

2.3 基本微粒群优化算法(Basic particle swarm optimization algorithms)

记第t代的微粒群为$x(t)$, 其第i个微粒为$x_i(t)$. 在PSO中, 微粒$x_i(t)$通过两个极值进行速度和位置更新, 一个是微粒$x_i(t)$本身到目前为止找到的最佳位置$p_i(t) = (p_{i1}(t), p_{i2}(t), \cdots, p_{id}(t))$, 称为微粒$x_i(t)$的个体极值; 另一个是整个微粒群到目前为止找到的最佳位置$p_{\mathrm{g}}(t) = (p_{\mathrm{g}1}(t), p_{\mathrm{g}2}(t), \cdots, p_{\mathrm{g}d}(t))$, 称为微粒全局极值. 微粒更新公式如下:

$$\begin{cases} v_{ij}(t+1) = wv_{ij}(t) + c_1 r_1 (p_{ij}(t) - \\ \qquad\qquad x_{ij}(t)) + c_2 r_2 (p_{\mathrm{g}j}(t) - x_{ij}(t)), \\ x_{ij}(t+1) = x_{ij}(t) + v_{ij}(t+1). \end{cases} \tag{2}$$

式中: $i = 1, 2, \cdots, N$, N为微粒群规模, $j = 1, 2, \cdots, d$, d为微粒的维数, c_1, c_2是加速度系数, r_1, r_2是$[0, 1]$之间的随机数, w是惯性权重, $v_{ij} \in [-v_{\max}, v_{\max}]$, $v_{\max}$由用户事先设定.

Merwe首次利用微粒群优化算法进行聚类, 采用聚类中心的坐标表示微粒位置, 提出2种和K–均值算法结合的微粒群优化聚类方法[3]. 在此基础上, 刘靖明等对其中的微粒编码方法和目标函数进行了改进[4]. 高尚等借鉴遗传算法中交叉、变异等思想, 给出了若干种基于交叉、变异策略的微粒更新方法, 并提出了求解聚类问题的混合微粒群算法[5]. Omran等将已有的微粒群优化聚类算法用于图像分割问题中[6]. 总之, 基于微粒群优化算法解决聚类问题已取得了可喜的研究成果, 但是, 上述方法的聚类数目都是基于先验知识人为给定的, 没有进行聚类数目的优化. 实际上, 由于聚类问题的复杂性以及人的认知能力的限制, 准确的聚类数目往往是很难甚至无法给出的, 有必要根据实际问题和某种性能指标进行优化. 因此, 如何利用PSO优化聚类数目成为亟需解决的关键问题.

3 基于微粒群优化聚类数目的K–均值算法(K-means algorithm for optimizing number of clusters based on PSO)

3.1 微粒编码(Particle coding)

Xu等给出了如图1所示的聚类数目经验区间$\tilde{K} = [2, K_{\max}]$[1], 其中: $K_{\max} = \lceil\sqrt{|O|}\rceil$, $\lceil\cdot\rceil$表示向上取整. $\tilde{K}$的合理性已从理论上得到证明[10].

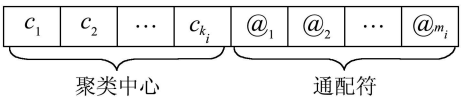


图 1 微粒编码结构

Fig. 1 Structure of particle coding

为使聚类数目K能够在$\tilde{K}$中得到优化, 将K融入到微粒群优化算法的微粒编码中, 采用聚类中心的坐标和通配符“@”表示微粒的位置. 图1给出了笔者提出的微粒编码结构, 包含$K_{\max}$个位置, 其中, 前半部分为聚类中心的坐标, 后半部分为通配符.

3.2 微粒群初始化(Initialization of particle swarm)

基于上述编码, 采用如下的微粒群初始化方法: 考虑第i个微粒$x_i(0)$, $i = 1, 2, \cdots, N$, 首先, 随机产生$[2, K_{\max}]$的整数K_i; 然后, 从待聚类数据集中随机选取K_i个样本作为初始聚类中心; 最后, 选择$K_{\max} - K_i$个“@”对剩余位置进行补充. 这样, 微粒$x_i(0)$就代表一种初始聚类结果, 整个微粒群$x(0)$就代表N种初始聚类结果.

3.3 适应度评价(Fitness evaluation)

DB准则是一类经典的聚类有效性评价指标, 采用类内紧致性和类间分离性评价聚类结果的好坏, 定义为[1]

$$DB = \frac{1}{K}\sum_{i=1}^{K} R_i, \tag{3}$$

其中:

$$R_i = \max_{j=1,2,\cdots,K;j\neq i}\{\frac{S_i+S_j}{d_{ij}}\},$$

$$S_i = \frac{1}{|C_i|}\sum_{o\in C_i}\|o-c_i\|, d_{ij} = \|c_i - c_j\|.$$

S_i用来评价类内的紧致性, d_{ij}用来评价类间的分离性. 因为一个好的聚类划分应使类间的分离性和类内的紧致性尽可能的大, 因此, DB取最小值时对应最佳划分. 这样一来, 可以定义如下的目标函数:

$$f(x) = \frac{1}{DB(x)+1}. \tag{4}$$

显然, $f(x)$越大, 聚类效果越好. 因此, 本文考虑的聚类问题可以转化为求式(4)表示的最大值问题.

3.4 微粒更新(Particle update)

微粒更新是PSO的关键环节. 与传统微粒更新策略不同的是, 本文采用实数和通配符混合编码. 如果仍然利用公式(2)更新微粒, 就会出现实数和通配符之间进行四则运算, 这在以前是没有定义的. 另外, 微粒更新不仅要更新聚类中心的坐标, 还要更新聚类数目, 这在以前也是未曾处理的. 为解决上述问题, 保证微粒更新能够顺利进行, 首先对式(2)中的“$-$”和“$+$”运算进行重新定义, 并分别记为“$\widehat{-}$”和“$\widehat{+}$”, 在此基础上, 给出新的微粒更新公式.

考虑微粒群$x(t)$的两个微粒$x_i(t)$和$x_j(t)$, $i,j = 1,2,\cdots,N$, 其第q维分量分别为$x_{iq}(t)$和$x_{jq}(t)$, $q = 1,2,\cdots,d$. 首先, 给出“$\widehat{-}$”的定义:

定义 1　定义$\widehat{-}$:

$$x_{iq}(t)\widehat{-}x_{jq}(t) = \begin{cases} x_{iq}(t)-x_{jq}(t), \text{ 如果, } x_{iq}(t), x_{jq}(t)\neq @;\\ x_{lq}(t), \text{ 如果 } x_{iq}(t)=@,\ \ x_{jq}(t)\neq @,\ r\geqslant\lambda;\\ @, \quad \text{如果 } x_{iq}(t)=@,\ \ x_{jq}(t)\neq @,\ \ r<\lambda;\\ @, \quad \text{如果, } x_{iq}(t)=@,\ x_{jq}(t)=@. \end{cases} \tag{5}$$

式中: r为$[0,1]$内的随机数, λ为人为设定的阈值, $x_{lq}(t)$为从$x(t)$中随机选取的微粒.

同理, 这样定义“$\widehat{+}$”:

定义 2　定义$\widehat{+}$:

$$x_{iq}(t)\widehat{+}x_{jq}(t) = \begin{cases} x_{iq}(t)+x_{jq}(t), \text{ 如果 } x_{iq}(t), x_{jq}(t)\neq @;\\ v_{lq}(t), \text{ 如果 } x_{iq}(t)=@,\ x_{jq}(t)\neq @,\ r\geqslant\lambda;\\ @, \quad \text{如果 } x_{iq}(t)=@,\ x_{jq}(t)\neq @,\ r<\lambda;\\ @, \quad \text{如果 } x_{iq}(t)=@,\ x_{jq}(t)=@. \end{cases} \tag{6}$$

与“$\widehat{-}$”定义不同的是, $v_{lq}(t)$从$[-v_{\min}, v_{\max}]$中随机选取.

基于上述定义, 给出如下新的微粒更新公式:

$$\begin{cases} v_{ij}(t+1) = wv_{ij}(t)\widehat{+}c_1r_1(p_{ij}(t)\widehat{-}x_{ij}(t))\widehat{+}\\ \qquad\qquad c_2r_2(p_{gj}(t)\widehat{-}x_{ij}(t)),\\ x_{ij}(t+1) = x_{ij}(t)\widehat{+}v_{ij}(t+1). \end{cases} \tag{7}$$

式(7)采用传统的微粒更新公式框架, 通过定义其中的“$\widehat{-}$”和“$\widehat{+}$”运算, 不仅实现了聚类中心坐标的更新, 更重要的是, 实现了聚类数目的更新. 另外, 采用的随机策略保证了微粒群的多样性, 提高了算法的全局寻优能力.

3.5 算法步骤(Steps of the proposed algorithm)

算法步骤如下:

Step 1　按照3.2节方法初始化微粒群;

Step 2　对微粒群中的每一微粒, 执行K–均值算法, 得到新的聚类中心坐标, 更新原来的微粒编码;

Step 3　按照式(4)评价微粒的适应值;

Step 4　更新微粒个体极值和全局极值;

Step 5　根据式(7)调整微粒的速度和位置;

Step 6　如果满足终止条件, 算法结束, 输出最优聚类划分; 否则, 转Step 2.

本文算法通过随机初始化一组可行解, 利用PSO算法的速度–位置更新模型, 并与K–均值算法基于的梯度下降迭代过程相结合, 很好地保持了广度搜索和深度搜索的平衡, 降低了聚类结果对初始聚类中心的敏感度. 更重要的是, 该算法可以自动优化聚类数目.

4 算例(Examples)

为了验证笔者提出的方法的有效性, 对人工和真实数据集进行聚类实验, 作为比较, 笔者也同时给出了K–均值算法、PSOK算法[4]、GCUK算法[11]的聚类结果. 所有实验均在CPU为P4/2.66 GHz、内存为256 MB的PC机上进行.

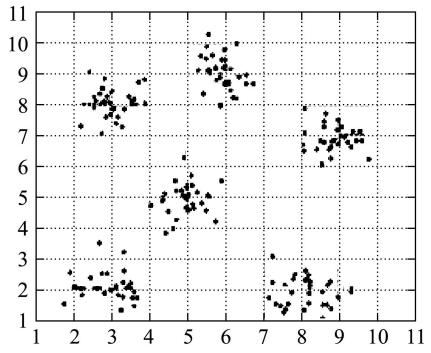
4.1 测试数据集(Data sets)

表1描述了各测试数据集的特征. 其中, S_1, S_2和S_3为3个服从高斯分布的人工数据集, 如图2(a)∼(c)所示, S_1和S_2分别为2维和3维的分离性较好的球形簇集, 而S_3是边缘比较模糊的含有较多类的2维簇集. Iris和Wine为真实数据集, 来自权威的UCI机器学习数据库[12], 经常被用来检验聚类算法的有效性. Iris数据集由3类4维空间的150个样本组成, 样本的4个分量分别表示Iris数据的花瓣长度、花瓣宽度、萼片长度和萼片宽度. 在此, 选取其中的前3个属性, 给出其样本空间分布, 如图2(d)所示, 从中可以看出, 其中1个类与其他2类完全分离, 而另外2类之间有部分交叉. Wine数据集由3类共178个样本组成, 每个样本有13个属性, 其中, 3个类不存在交迭, 但划分不太明显.

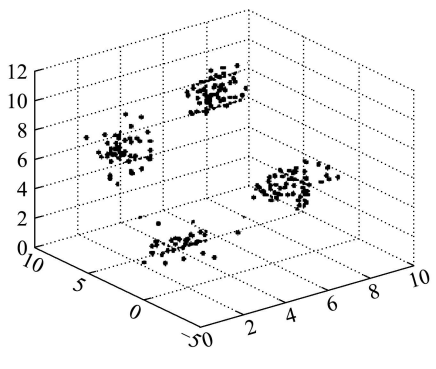
表 1 数据集描述

Table 1 Description of data sets

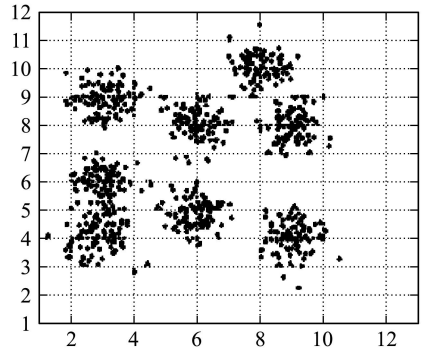
数据集	大小	类别数	维数	各类包含对象数目
S_1	200	4	2	50
S_2	180	6	3	30
S_3	720	8	2	90
Iris	150	3	4	50, 50, 50
Wine	178	3	13	59, 71, 48



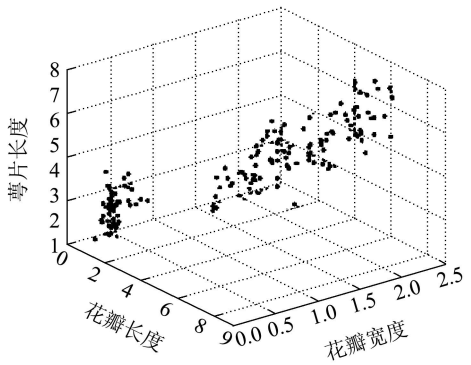
(a) 数据集 S_1



(b) 数据集 S_2



(c) 数据集 S_3



(d) 数据集 lris

图 2 测试数据集分布

Fig. 2 Distribution of data sets

4.2 结果及分析(Results and analysis)

为了评价聚类结果, 引入聚错率和纯度两个指标. 其中, 纯度值在0和1之间, 越大越好. 两者皆刻画了聚类结果的准确性. 本文算法的参数设置如下: 微粒群规模为50, 最大进化代数为100, 加速度系数$c_1 = c_2 = 2$, 惯性权重w随进化代数从0.9线性减少到0.5, 最大飞行速度$v_{\max} = 0.01$.

通过实验设定微粒更新的概率阈值λ. 以S_1, S_2和S_3为例, 按照式(8)选择的λ值分别做20次试验, 得到使聚类纯度最高的3个概率阈值, 然后取平均值, 即为本文算法采用的λ值:

$$\lambda_i = 0.05 + 0.05(i-1),\ i = 0, 1, 2, \cdots, 20. \quad (8)$$

按照上述参数设置, 对每个数据集分别做10次聚类实验. 表2给出了本文算法10次聚类实验结果的纯度、聚错率和聚类数目正确的次数, 其中, 括号前的数字表示平均值, 括号内的数字表示标准差. 可以看出, 本文算法能够对离散性较好的数据集S_1和S_2实现正确的聚类;对于结构较为复杂的数据集Iris和Wine, 虽不能实现完全正确的聚类, 但皆能得到正确的聚类数目, 并且具有较高的纯度和较低的聚错率.

表 2　10次实验结果

Table 2　Results of 10 trials

数据集	纯度	聚错率/%	K值正确次数
S_1	1(0)	0(0)	10
S_2	1(0)	0(0)	10
S_3	0.901(0.004)	2.11(0.002)	10
Iris	0.894(0.002)	2.44(0.003)	10
Wine	0.857(0.013)	2.59(0.005)	10

作为比较, 表3给出了上述4种算法对于Iris数据集的10次聚类结果. 其中, 前两种方法的聚类数目人为设定, 均为$K = 3$. K–均值算法基于梯度下降寻优, 虽然收敛速度很快, 但由于对初始聚类中心敏感, 造成聚类结果波动性较大, 聚类质量较差;而本文算法、PSOK及GCUK算法, 很大程度上降低了聚类结果对初始聚类中心的敏感度, 具有较高的聚类质量. 从运行时间上看, 本文算法优于GCUK算法, 这正体现了PSO在实际优化中具有比GA更快速的寻优性能; 相比于PSOK算法, 本文算法在优化聚类中心的同时还要优化K值, 因而其运行时间要高于PSOK算法.

表 3　不同算法的聚类结果

Table 3　Clustering results of different algorithms

算法	纯度	聚错率/%	运行时间/s
K–均值	0.653(0.184)	15.6(1.023)	0.57
PSOK	0.897(0.003)	2.45(0.005)	2.31
GCUK	0.891(0.009)	2.51(0.004)	3.83
本文算法	0.894(0.002)	2.44(0.003)	2.88

5　结语(Conclusions)

在分析K–均值算法及传统微粒群聚类算法缺点的基础上, 笔者提出一种基于微粒群优化聚类数目的K–均值算法. 采取的微粒编码方法及更新策略能动态地优化聚类数目, 实现了聚类数目的自动获取, 而不再借助于先验知识由用户人为设定. 5个人工和真实数据集的聚类结果验证了所提算法的有效性.

需要进一步研究的问题包括: 将所提算法应用于不确定数据集的聚类中, 并采用新的指标, 评价所提算法的聚类效果.

参考文献(References):

[1] XU R, DONALD W. Survey of clustering algorithms[J]. *IEEE Transactions on Neural Networks*, 2005, 16(3): 645 – 678.

[2] KENNEDY J, EBERHART R C. Particle swarm optimization[C] //*Proceedings of the 1995 IEEE International Conference on Neural Networks*. Piscataway, NJ: IEEE, 1995: 1942 – 1948.

[3] MERWE D W, ENGELBRECHT A P. Data clustering using particle swarm optimization[C] //*Proceedings of the 2003 Congress on Evolutionary Computation*. Piscataway, NJ: IEEE, 2003, 1: 215 – 220.

[4] 刘靖明, 韩丽川, 侯立文. 基于粒子群的K-均值聚类算法[J]. 系统工程理论与实践, 2005, 6(3): 55 – 58.
(LIU Jingming, HAN Lichuan, HOU Liwen. Clustering analysis based on particle swarm optimization algorithm[J]. *System Engineering–Theory and Practice*, 2005, 6(3): 55 – 58.)

[5] 高尚, 杨静宇. 求解聚类问题的混合粒子群优化算法[J]. 科学技术与工程, 2005, 23(5): 1792 – 1795.
(GAO Shang, YANG Jingyu. Solving clustering problem by hybrid particle swarm optimization algorithm[J]. *Science Technology and engineering*, 2005, 23(5): 1792 – 1795.)

[6] OMRAN M, ENGELBRECHT A P, SALMAN A. Particle swarm optimization method for image clustering [J]. *International Journal of Pattern Recognition and Artificial Intelligence*, 2005, 19(3): 297 – 321.

[7] CHIOU Y C, LAWRENCE W. Genetic clustering algorithms [J]. *European Journal of Operational Research*, 2001, 135(6): 413 – 427.

[8] ELIZABETH L, OLFA N, JONATAN. ECSAGO:evolutionary clustering with self adaptive genetic operators[C] //*Proceedings of the 2006 IEEE Congress on Evolutionary Computation, BC, Canada*. Piscataway, NJ: IEEE, 2006: 1768 – 1775.

[9] 钱线, 黄萱菁, 吴立德. 初始化K-means的谱方法[J]. 自动化学报, 2007, 33(4): 342 – 346.
(QIAN Xian, HUANG Xuanjing, WU Lide. A spectral method of K-means initialization[J]. *Acta Automatic Sinica* 2007, 33(4): 342 – 346.)

[10] 李永森, 杨善林, 胡笑旋, 等. 空间聚类算法中的K值优化问题研究[J]. 系统仿真学报, 2006, 18(3): 574 – 576.
(LI Yongsen, YANG Shanlin, HU Xiaoxuan, et al. Optimization study on K value of spatial clustering[J]. *Journal of System Simulation*, 2006, 18(3): 574 – 576.)

[11] BANDYOPADHYAY S, MAULIK U. Genetic clustering for automatic evolution of clusters and application in image classification[J]. *Pattern Recognition* , 2002, 35(2): 1197 – 1208.

[12] Data sets for clustering techniques[EB/OL]. http://www.unikoeln.de/themen/Statistik/data/cluster. 2007-9-20.

作者简介:

巩敦卫　(1970—), 男, 研究方向为智能优化与控制等, E-mail: dwgong@vip.163.com;

蒋余庆　(1982—), 男, 研究方向为进化计算和聚类分析, E-mail: jyqingfly2008@126.com;

张　勇　(1979—), 男, 研究方向为微粒群优化, E-mail: yongzh-401@126.com;

周　勇　(1979—), 男, 研究方向为数据挖掘等, E-mail: yzhou@cumt.edu.cn.