多种群协同进化的 *K*-means聚类算法

曲建华[1], 邵增珍[2]

(1.山东师范大学管理与经济学院, 山东 济南 250014)
(2.山东师范大学信息科学与工程学院, 山东 济南 250014)

[摘要] 针对 *K* 均值聚类算法易陷入局部最小的缺点, 提出了一种多种群协同进化的微粒群和 *K* 均值混合聚类算法, 它将整个种群分解为多个子种群, 各子种群独立进化, 周期性地更新共享信息. 同时将此算法与现有的基于遗传算法的 *K* 均值聚类算法进行了比较. 实验结果证明, 该算法能有效地克服传统的 *K* 均值算法易陷入局部极小值的缺点, 同时全局收敛能力优于基于遗传算法的 *K* 均值聚类算法.

[关键词] 多种群, 微粒群算法, *K* 均值算法, 协同进化
[中图分类号] TP391 [文献标识码] A [文章编号] 1001-4616(2010)03-0001-05

Cooperative Evolutionary *K*-means Clustering Algorithm With Multi-populations

Qu Jianhua[1], Shao Zengzhen[2]

(1. School of Management and Economics, Shandong Normal University, Jinan 250014, China)
(2. School of Information Science and Engineering, Shandong Normal University, Jinan 250014, China)

Abstract This paper presents an mixed clustering algorithm based on cooperative evolution with multi-populations. It adopts cooperative evolutionary strategy with multi-populations to change the mode of traditional searching optimum solutions. The whole clustering process is divided into two stages. The first stage uses the cooperative evolutionary PSO algorithm to search the initial clustering centers. The second stage uses the *K*-means algorithm. The experiment proved that this method was able to extract the correct number of clusters with good clustering quality compared to the results obtained from other clustering algorithms like *K*-means and PSO clustering algorithm.

Key words multi-population, PSO, *K*-means, cooperative evolution

由 MacQueen提出的 *K* 均值算法是解决聚类分析问题的一种经典算法[1], 广泛应用于数据挖掘和知识发现领域中. 传统的 *K* 均值算法存在两个固有的缺点:

(1) 对于随机的初始值选取可能会导致不同的聚类结果, 甚至存在着无解的情况;

(2) 该算法是基于梯度下降的算法, 因此不可避免地常常陷入局部极优.

这两大缺陷大大限制了它的应用范围. 为了克服 *K* 均值的上述缺陷, 近几年很多文献[1-4]结合遗传算法对 *K* 均值算法进行改进. 但是, 实验表明, 当样本数目、维数和类别数较大时, 这些算法常常过早的收敛于局部极优点, 而且由于进化算法在进化过程中可能产生退化现象, 将导致迭代次数过长以及聚类准确率不高, 并且可能产生进化后期的波动现象.

粒子群优化算法(PSO)[5]不但具有遗传算法的全局寻优能力, 通过调整参数, PSO 还可同时具有较强的局部寻优能力, 由于没有个体杂交、变异等运算操作, PSO 的参数调整变得简单易行, 在大多数情况下, 比遗传算法更快的收敛于最优解, 而且可以避免完全随机寻优的退化现象. 因此将粒子群算法与传统的 K 均值算法相结合, 可以有效地改进聚类质量[6-12].

收稿日期: 2010-06-10
基金项目: 山东省科技攻关计划项目(2009GG10001008)、山东省软科学研究计划项目(2009RKA285)、济南市高校院所自主创新项目(200906001).
通讯联系人: 曲建华, 讲师, 研究方向: 数据挖掘信息系统. E-mail: qujh2003@sohu.com

在标准 PSO 算法的进化过程中, 每一代均更新信息, 即更新个体的历史最好位置和群体的历史最好位置. 这样就没有充分利用当前所搜索到的最好位置, 一旦粒子位置陷入局部最好值, 算法将无法继续进化, 从而影响 PSO 算法的全局收敛性. 如果将种群平均划分为若干个子种群, 各个子种群独立地用标准 PSO 进化. 间隔一定迭代次数后, 实施子种群之间的信息交换: 将 1个子种群中的最好解发送至相邻子种群, 取代其中适应值最差的粒子. 反复迭代, 以此改善 PSO 算法的收敛性. 为此, 将多种群协同进化的策略引入 PSO算法, 尽可能抑制早熟现象的发生.

本文研究多种群协同进化的混合聚类算法. 首先介绍了多种群协同进化的概念, 然后介绍了传统的 K 均值聚类算法和标准的 PSO算法, 在此基础上, 给出了一种多种群协同进化的混合聚类算法. 最后通过仿真实验证明了算法的有效性.

1 多种群的协同进化

多种群协同进化的策略改变了传统的用一个种群在解空间中搜索最优解的方式, 它将整个种群分解为几个子种群, 协同进化. 所谓的协同进化, 是指将解空间中的群体划分为若干子群体, 每个子群体代表求解问题的一个子目标, 所有子群体在独立进化的同时, 基于信息迁移与知识共享, 共同进化.

协同进化算法中最常见的协同模型是“孤岛模型”与“邻域模型”. 在这 2种模型中, 直接将群体中的个体划分为若干子群体, 每一子群体代表解空间中的一个子空间, 其中的每一个体均代表问题一个解. 所有子群体并行展开局部搜索, 所搜索到的优良个体将在不同子群体间进行迁移, 作为共享信息指导进化的进行, 从而有效提高算法的全局收敛效率.

2 传统的 K 均值算法和微粒群算法

2.1 传统的 K 均值算法

K 平均算法以 K 为参数, 把 n 个对象分成 K 个簇, 使簇内具有较高的相似度, 而簇间的相似度较低. 相似度的计算根据一个簇中对象的平均值即簇的质心来进行的. 基本思想: 首先, 随机选取 k 个对象作为初始的 K 个簇的质心; 然后, 将其余对象根据其与各个簇质心的距离分配到最近的簇; 最后重新计算每个簇的质心. 这个过程不断重复, 直到目标函数最小化为止. 通常采用的目标函数形式为平方误差准则函数: $E = \sum_{i=1}^{k}\sum_{p\in C_i} \| \boldsymbol{p} - \boldsymbol{c}_i \|^2$, 期中, $\boldsymbol{p}$ 为数据对象, $\boldsymbol{c}_i$ 表示簇 C_i 的质心, E 就表示数据集中所有对象的平方误差的和. 这个目标函数使生成的簇尽可能地紧凑和独立, 它使用的距离度量是欧几里得距离, 当然也可以采用其他距离度量. K 平均聚类算法的过程如下:

步骤 (1) 随机选取 k 个对象作为初始的簇的质心;

步骤 (2) 计算对象与各个族的质心的距离, 将对象划分到距离其最近的簇;

步骤 (3) 重新计算每个新簇的均值, 即质心;

步骤 (4) 若簇的质心不再变化, 则返回划分结果, 否则转步骤 (2).

2.2 标准的 PSO 算法

粒子群算法是由美国的 Kennedy和 Eberhar在 1995年提出的. 在 PSO算法中, 粒子通过不断调整自己的位置 X 来搜索新解. 每个粒子都能记住自己搜索到的最好解, 记做 P_{id}, 以及整个粒子群经历过的最好的位置, 即目前搜索到的最优解, 记做 P_{gd}. 每个粒子都有一个速度, 记作 V_{id},

$$V_{id}^{l} = \omega V_{id} + \eta_1 \mathrm{rand}(\,)(P_{id} - X_{id}) + \eta_2 \mathrm{rand}(\,)(P_{gd} - X_{id}), \quad (1)$$

其中 V_{id} 表示第 i 个粒子在第 d 维上的速度, ω 为惯性权重, $\eta 1$, $\eta 2$为调节 P_{id} 和 P_{gd} 相对重要性的参数, rand( ) 为随机数生成函数. 这样, 可以得到粒子移动的下一位置:

$$X_{id}^{l} = X_{id} + V_{id}, \quad (2)$$

PSO的基本算法步骤描述如下:

1) 初始化粒子群, 即随机设定各粒子的初始位置 X 和初始速度 V;

2) 计算每个粒子的适应度值;

3) 对每个粒子, 比较它的适应度值和它经历过的最好位置 P_{id} 的适应度值, 如果更好, 更新 P_{id};

4) 对每个粒子,比较它的适应度值和群体所经历的最好位置 P_{gd} 的适应度值,如果更好,更新 P_{gd};

5) 根据(1)式和(2)式调整粒子的速度和位置;

6) 如果达到结束条件(足够好的位置或最大迭代次数),则结束,否则转步骤(2).

3　多种群协同进化的混合聚类算法

3.1　算法模型

在多种群协同进化的混合聚类算法中,PSO算法采用的是实数编码方式,1个编码对应于 1个可行解.聚类算法采用的是基于聚类中心的编码方式,也就是每个粒子的位置是由 m 个聚类中心组成.每个粒子的状态由粒子的位置、速度和适应度函数值决定,用下面的三元组表示:$(\boldsymbol{S}_i, v_i, f_i)$, $1 \leqslant i \leqslant n$

这里,n 为粒子个数;$\boldsymbol{S}_i$ 是第 i 个粒子的位置;v_i 是第 i 个粒子在运动过程中的速度;f_i 是第 i 个粒子的适应度函数值.

$$\boldsymbol{S}_i \text{ 由 } m \text{ 个聚类中心组成: } \boldsymbol{S}_i = (p_{i1} p_{i2} \cdots p_{im});$$

当聚类中心确定时,聚类的划分由下面的最近邻法则决定:若数据 x_i 与聚类中心 p_j 的距离满足下面的公式,则 x_i 属于第 j 类.

$$d(x_i, p_j) = \min_{k=1, \cdots, m} (d(x_i, p_k)).$$

3.2　适应度计算

对于某粒子,按照以下方法计算其适应度:

1) 按照最近邻法则式,确定对应该粒子的聚类划分;

2) 根据聚类划分,计算类内离散度和 $\boldsymbol{J}_i$: $\boldsymbol{J}_i = \sum_{k=1}^{m} \sum_{x_j \in C_k} d(x_i, p_k)$.

3) 个体的适应度函数为: $f(\boldsymbol{S}_i) = 1/\boldsymbol{J}_i$, 其中 $\boldsymbol{J}_i$ 是总的类间离散度和.

这样个体的适应度与离散度和负相关,离散度和越小,个体适应度越大.

3.3　多种群协同进化策略

多种群协同进化策略:将整个种群 N 的粒子分解为 M 个子种群,各子种群依次进化.每个子种群内部都用标准 PSO算法进化,不断更新子种群内部粒子的速度和位置.当进化到第 R 代时(R 为更新周期),第一个子种群将其当前的最好值 P_{g1} 传给第二个子种群.第二个子种群依据 P_{g1} 进化,达到周期时,将它的当前最好值 P_{g2} 传给第三个子种群,依次类推.最后 1个子种群将 P_{gM} 传回给第 1个子种群.每次各个子种群得出该种群内的当前最好位置,传递给下 1个相邻的子种群的同时判断 P_{gi} ($i = 1, 2, 3, \cdots M$) 是否满足精度,若满足则停止,否则继续进化.各个子种群每隔 R 代,相邻的 2个子种群之间进行信息交换.循环进化,直到算法终止.见图 1.

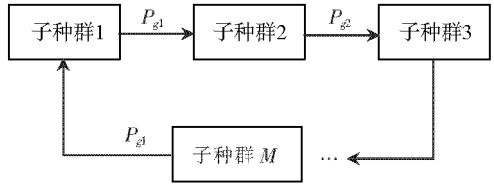

图 1　多种群协同进化策略

Fig.1　Cooperative evolutionary strategy

这样每个子种群进化时,与它相邻的前一个子种群总会直接给它一个当前的较好位置引导它的进化.在这样的引导之下,各个子种群内的粒子飞行方向就能很快地改变,朝着最好位置飞去,提高算法的收敛速度.

3.4　算法描述

1) 种群的初始化

在初始化粒子时,先将每个样本随机指派为某一类,作为最初的聚类划分,并计算各类的聚类中心,作为初始粒子的位置编码,计算粒子的适应度,并初始化粒子的速度.反复进行 N 次,共生成 N 个初始粒子群,将 N 个初始粒子群分为 M 个子群.

2) 对当前子群的每个粒子,比较它的适应度值和它经历过的最好位置 P_{id} 的适应度值,如果更好,更新 P_{id};

3) 对当前子群的每个粒子,比较它的适应度值和群体所经历的最好位置 P_{gd} 的适应度值,如果更好,更新 P_{gd};

4) 根据 (1) 式和 (2) 式调整粒子的速度和位置;

5) 新个体的 *K* 均值优化. 对于新一代粒子, 按照以下的 *K* 均值算法进行优化;

① 根据粒子的聚类中心编码, 按照最近邻法则, 来确定对应该粒子的聚类划分;

② 按照聚类划分, 计算新的聚类中心, 更新粒子的适应度值, 取代原来的编码值;

6) 如果到达迭代周期 *R*, 则将当前子群的最好 P_{id} 和 P_{gd} 传递给下一个子群, 循环进行;

7) 如果达到结束条件 (足够好的位置), 则结束, 否则转步骤 2).

在算法执行过程中, 如果出现空的聚类, 则随机地从其他某个非空的聚类中取出距离聚类中心最远的模式向量, 将该向量放入空聚类, 重复这个过程, 直到划分中没有空聚类为止.

基于粒子群的 *K* 均值算法在产生下一代解群有较大的随机性, 所以不容易陷入局部极小值, 而且每个子种群进化时, 与它相邻的前一个子种群总会直接给它一个当前的较好位置引导它的进化. 在这样的引导之下, 各个子种群内的粒子飞行方向就能很快地改变, 朝着最好位置飞去, 极大地提高算法的收敛速度.

4 仿真实验

为了检验算法的快速性和有效性, 我们进行了下面的仿真实验. 实验数据为 150个对象, 我们分别用 *K* 均值算法、PSO 算法和多种群协同进化算法进行测试. 多种群协同进化算法参数设置如下: $R = 50$, $\eta 1 = \eta 2 = 2.0$, $\omega = 0.72$ 聚类结果见图 2 ~ 5

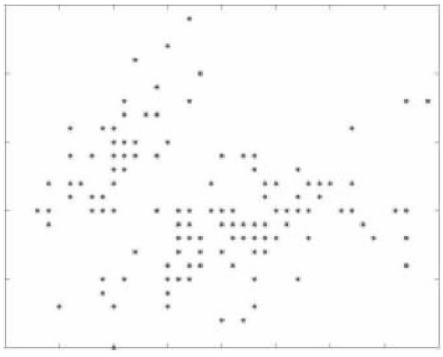

图 2 Iris 数据集

Fig.2 Iris data set

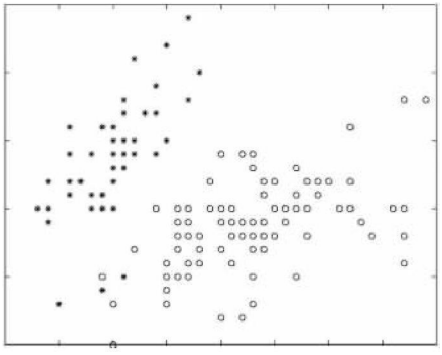

图 3 *K*-means 聚类

Fig.3 *K*-means clustering

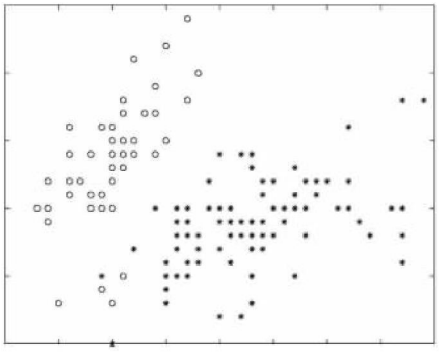

图 4 PSO 聚类

Fig.4 PSO Clustering

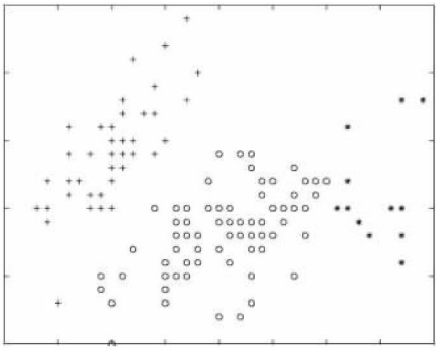

图 5 CEPSO 聚类

Fig.5 CEPSO clustering

从实验结果可以看出, 针对同样的聚类效果, 多种群算法的分类效果是最优的.

5 结论

多种群的协同进化混合聚类算法是在传统的 *K* 均值算法和 PSO 算法的基础上, 引入了多种群的协同进化策略, 大大改进了算法性能, 克服了传统聚类算法存在的问题, 全局寻优能力优于现有的 *K* 均值算法和 PSO 算法, 具有较快的收敛速度和好的聚类精度. 但在大规模数据集上, 算法是否具有同样的优越性,

有待于进一步研究和验证.

[参考文献]

[1] LiM J, NgM K, Cheung Y M, et al. Agglomerative fuzzy*K*-means clustering algorithm with selection of number of clusters [J]. IEEE Transactions on Knowledge and Data Engineering, 2008, 20(11): 1519-1534.

[2] Laszlo M, Mukherjee S. A genetic algorithm that exchanges neighboring centers for*K*-means clustering[J]. Pattern Recognition Letters, 2007, 28(16): 2359-2366.

[3] Arthur D, Vassilvitskii S. *K*-means++: the advantages of careful seeding[C]//Proceedings of the 18th annual ACM-SIAM Symposium on Discrete algorithms. New Orleans: Society for Industrial and Applied Mathematics, 2007: 1027-1035.

[4] Likas A, Vlassis N, Verbeek J. The global*K*-means clustering algorithm[J]. Pattern Recognition, 2003, 36(2): 451-461.

[5] Kennedy J, Eberhart R, Shi Y. Swarm Intelligence[M]. [S.l.]: Morgan Kaufmann Publishers, 2001.

[6] de Castro L N, Von Zuben F J. Recent Developments in Biologically Inspired Computing[M]. London: Idea Group Inc, 2004.

[7] Bonabeau E, Dorigo M, Theraulaz G. Swarm Intelligence: From Natural to Artificial Systems[M]. [S.l.]: Oxford University Press, 1999.

[8] Kennedy J, Eberhart R C. Particle swarm optimization[C]//Proceedings of IEEE International Conference on Neural Networks, IV. Perth: IEEE, 1995: 1942-1948.

[9] Van der Merwe D W, Engelbrecht A P. Data clustering using particle swarm optimization[C]//Proceedings of IEEE Congress on Evolutionary Computation 2003 (CEC 2003), Canberra: IEEE, 2003: 215-220.

[10] Shubham Agrawal, Panigrahi B K, Manoj Kumar Tiwari. Multiobjective particle swarm algorithm with fuzzy clustering for electrical power dispatch[J]. IEEE Transactions on Evolutionary Computation, 2008, 12(5): 529-541.

[11] Chang Dongxia, Zhang Xianda, Zheng Changwen. A genetic algorithm with gene rearrangement for*K*-means clustering[J]. Pattern Recognition, 2009, 42(7): 1210-1222.

[12] Lai J Z C, Huang T J, Liaw Y C. A fast*K*-means clustering algorithm using cluster center displacement[J]. Pattern Recognition, 2009, 42(11): 2551-2556.

[责任编辑: 顾晓天]