一种改进的 *K*-means 蚁群聚类算法

李 振,贾瑞玉
(安徽大学 计算机科学与技术学院,安徽 合肥 230601)

摘 要:现有的 *K* -means 蚁群聚类算法,首先进行 *K* -means 聚类算法操作,快速、粗略地确定初始聚类中心,接着根据上一步获得的聚类中心再进行蚁群算法聚类操作,有效地解决蚁群聚类算法收敛速度过慢的问题。研究发现,现有的 *K* -means 蚁群聚类算法并没有改善算法在迭代后期易出现收敛于非全局最优的缺陷。针对这一问题,提出一种改进的 *K* -means 蚁群聚类算法。每次迭代结束时,随机选择一个或多个簇,再从选中的簇里选择含有信息素最小的节点进行变异操作,把选中的节点变异到其他簇,计算评价值判断变异是否进行。仿真实验结果表明,用 *F* 值表示的平均值和最差结果都比原有的算法较好,有效解决了原有算法易收敛于非全局最优及早熟问题,但由于变异操作使算法运行时间相对较长。

关键词:聚类; *K* -means 算法;蚁群聚类算法;聚类组合;变异
中图分类号:TP301.6 **文献标识码**:A **文章编号**:1673-629X(2015)12-0028-04
doi:10.3969/j.issn.1673-629X.2015.12.007

An Improved *K*-means Ant Colony Clustering Algorithm

LI Zhen, JIA Rui-yu
(School of Computer Science and Technology, Anhui University, Hefei 230601, China)

Abstract: Existed *K* -means ant colony clustering algorithm carries out *K* -means algorithm operation, fast and roughly determines the clustering center, then according to rough clustering center, ant colony clustering algorithm is conducted again to solve the problem of low convergence speed effectively. The research shows that the existed *K* -means any colony clustering algorithm doesn't improve the defect of converging to non-global optimal in late iteration. In order to solve this problem, a modified *K* -means ant colony clustering algorithm is presented. At the end of each iteration, randomly select one or more clusters, and then choose the point from the selected cluster with minimum pheromones for mutation, the mutation selecting node to another cluster, evaluation value is calculated to judge whether to mutate. Experimental results show that the average and worst results indicated by *F* value are better than the original algorithm, effectively solving the problem that is easy to converge to non-global optimal and premature, but it takes a longer running time.

Key words: clustering; *K* -means algorithm; ant colony clustering algorithm; clustering combination; variation

0 引 言

数据聚类是数据挖掘研究中的重要内容之一,在许多领域得到了广泛应用,包括机器学习、模式识别、图像分析等[1]。聚类是把具有相似属性的数据对象通过一些聚类方法聚成不同的组别或者簇,这样就把具有相似属性高的成员归并到一个簇中,而数据对象属性差别较大的聚到不同的簇中,形成不同的簇。聚类分析[2]中目前比较流行的聚类方法包括层次聚类算法、划分聚类算法、基于网格聚类算法,以及基于密度的聚类算法等[3]。

K -means 算法是一种基于划分的聚类算法。由于具有算法简单且收敛速度快的优点,得到较普遍的应用,同时也出现了许多改进的 *K* -means 算法。

冯波等针对传统 *K* -means 算法对初始聚类中心敏感的问题,提出基于数据样本分布情况的动态选取初始聚类中心的改进 *K* -means 算法[4]。蚁群算法是一种新型的模拟生物进化算法,最早详细提出是在 M. Dorigo 的博士论文中,通过模拟蚂蚁的一些行为特征来解决多种优化问题,算法通用性和鲁棒性较强[5]。蚁群算法运用到聚类问题主要分为基于蚂蚁觅食原理聚类[6]和基于蚂蚁堆积尸体原理聚类[7]。为了解决早熟等问题,研究人员对蚁群算法进行了改进。Goss

收稿日期:2015-03-02 **修回日期**:2015-06-05 **网络出版时间**:2015-11-04
基金项目:国家自然科学基金资助项目(61202227)
作者简介:李 振(1991-),男,硕士生,研究方向为数据挖掘;贾瑞玉,副教授,硕士生导师,研究方向为智能计算与数据挖掘。
网络出版地址:http://www.cnki.net/kcms/detail/61.1450.TP.20151104.0953.072.html

等[8]对蚂蚁系统进行了改进,在后继节点选择时引入额外的启发信息,避免环的记忆功能。朱峰等在文献[9]中从蚂蚁捡起对象、放下对象的策略、参数的自适应改变策略及游离对象的处理策略四个不同方面对现有的蚁群算法进行改进。杨燕等在文献[10]中将运动速度类型各异的多个蚁群,独立而并行地进行聚类分析,然后组合其聚类结果为超图,再用蚁群算法对超图进行 2 次划分对算法进行改进。张蕾等在文献[11]中提出一种局部最邻近运动原则来指导蚂蚁的移动和自适应调整蚂蚁移动阈值的方法。针对早期蚁群聚类算法的缺点,邢猛等提出动态调整的蚁群聚类算法,降低蚁群移动的随意性,减少蚂蚁的搜索时间,提高聚类性能[12]。

文中介绍 K -means 和蚁群算法组合的聚类算法[13],通过用 K -means 算法得到粗略的聚类中,然后其结果再用蚂蚁算法进行聚类,改善了蚂蚁聚类收敛过慢的现象,但并没有改善蚂蚁算法迭代后期出现非全局最优和早熟现象。因此提出改进的 K -means 蚁群聚类算法。根据每只蚂蚁聚类情况选择聚到某类中信息素较小的数据对象进行变异操作,变异结果较好的进行信息素增益操作和更改簇别。改进的 K -means 蚁群聚类算法,提高了搜索效率,改善了容易出现停滞、过早收敛于局部最优解的现象,但运行时间稍差。

1　K-means 聚类算法

K -means 算法是一种基于划分的聚类算法,初始化时通过给出要聚类簇的数目和聚类中心,不断迭代更新聚类中心,以达到最优解。解的评价常使用目标函数 F :

$$F = \sum_{j=1}^{k} \sum_{i=1}^{m} \mathrm{dist}(p_i, c_j) \tag{1}$$

式中, m 为属性数目; p_i 为属于聚类中心 c_j 的数据对象; dist 为欧氏距离,定义如下:

$$\mathrm{dist}_{ij} = \left(\sum_{k=1}^{p} | x_{ik} - x_{jk} |^2 \right)^{\frac{1}{2}} \tag{2}$$

式中, p 表示数据的属性个数。

K -means 算法具体描述如下:

(1)随机选择 k 个点作为聚成簇的数目;

(2)在数据样本点中随机选取 k 个数据对象作为初始聚类中心;

(3)根据距离度量(文中采用欧氏距离),将数据对象分配给距离最小的聚类中心;

(4)重新计算聚类中心,把簇内对象属性的平均值作为新的聚类中心;

(5)采用公式(1)计算目标函数 F 的值;

(6)重复步骤(3),(4),(5)直到达到迭代次数或者簇趋于稳定。

在使用 K -means 聚类算法时必须在聚类开始时给出 k 个簇和初始聚类中心点,这个 k 值选定非常难以估计。而且初始的聚类中心确定一个初始的划分,其后的步骤是对初始的划分进行优化。初始聚类中心的选择对聚类结果的影响较大,一旦初始值选择不好,可能无法得到有效的聚类结果。另外, K -means 算法对孤立数据点非常敏感,极少的孤立数据就可能对聚类结果产生极大影响,从而影响算法性能。但 K -means 算法也有其一定的优势。首先,其算法思想简单,实现容易;另外算法运算时间复杂度低,收敛速度较快。因此,可以使用该算法对数据进行预处理,得到聚类结果的雏形。

2　蚁群聚类算法

基于觅食原理的蚁群算法,蚂蚁在觅食过程中释放一种信息素,该信息素能够被其他的蚂蚁所感知,起到指引作用。并且随着时间推移,信息素不断挥发,而蚂蚁选择信息素较多的路径,由于选择的蚂蚁不断释放信息素使该路径上的信息素不断增加,最终所有蚂蚁都选择该路径。蚂蚁的这种行为称之为正反馈机制,成功地运用于 TSP 等优化问题。

基于蚁群算法觅食原理的聚类算法思路如下:随机选择一个数据样本 i 作为蚂蚁遍历的起始位置,该蚂蚁根据公式(3)或者随机产生的一个随机值小于预先设置的 q_0 分配到聚类中心 $c_j (j = 1, 2, \cdots, k)$ 处,那么蚂蚁就在数据对象 i 到聚类中心 c_j 的路径上留下信息素 τ_{ij}。蚂蚁 i 选择聚类中心 j 的概率定义如下:

$$p_{ij} = \frac{\tau_{ij} \times \eta_{ij}}{\sum_{j=1}^{k} \tau_{ij} \times \eta_{ij}} \tag{3}$$

式中, η_{ij} 表示启发函数, $\eta_{ij} = \dfrac{1}{d_{ij}}$。

随着蚂蚁的移动,路径上信息素不断挥发,公式定义如下:

$$\tau_{ij}^{\mathrm{new}} = \rho \tau_{ij} + \frac{Q}{d_{ij}} \tag{4}$$

式中, ρ 为信息素的挥发系数; Q 为一个正常数; d_{ij} 表示数据对象 i 到聚类中心 c_j 的距离。

在应用蚁群聚类算法时,算法收敛的速度比较缓慢,特别在迭代初期,由于信息素的更新较慢所以很难把各个路径上的信息素明显区分开。但在迭代后期,某些路径上的信息素不断堆积,使以后操作的蚂蚁选择路径的可能性越来越趋向于信息素较多的路径,但不能保证其解是全局最优解,从而出现早熟现象。因此介绍 K -means 蚁群聚类算法,并给出改进算法。

3 *K*–means 蚁群聚类算法

利用 *K* –means 算法快速收敛的特性,将数据集进行预处理,得到粗略的聚类中心。用 *K* –means 算法预处理之后,根据数据对象 i 所属的聚类中心 c_j,在路径 d_{ij}(数据对象到聚类中心 c_j)分配较多的信息素 τ_{ij}。接下来,蚁群算法通过开始时到各个聚类中心的路径上不同的信息素进行聚类操作。*K* –means 蚁群聚类的详细过程描述如下:

(1)从数据样本中随机选取 k 个数据作为初始聚类中心:$c_1, c_2, \cdots, c_k$;

(2)通过上述的 *K* –means 聚类过程得到预处理的聚类中心和簇;

(3)给每个数据对象 i 到相对应的聚类中心 c_j 分配初始不同的信息素 τ_{ij},信息素根据步骤(2)的预处理过程,各路径上的信息素有所不同,其中属于某类的数据样本到其聚类中心的信息素相对较高。初始 Q、ρ(信息素挥发系数)、n(蚂蚁数)、q_0(分配阈值);

(4)随机产生 $p \in (0,1)$,如果 p 值小于给定的 q_0,则按路径上信息素大小原则把数据样本分配到某类,如果大于 q_0,则根据公式(3)计算蚂蚁转移概率,选择聚到某类;

(5)一只蚂蚁经过所有数据样本点,根据公式(1)计算 F 值,即新的聚类中心;

(6)所有蚂蚁完成一次遍历得到较好的聚类方案,给最好聚类方案按公式(4)进行信息素更新;

(7)达到预定的迭代次数则不在前进,输出最优解;否则转到步骤(2)继续运行。

K –means 蚁群聚类算法能够有效地改善蚁群算法初始时收敛速度较慢的问题,加快收敛速度,但基本蚁群聚类算法的一些缺点仍没有有效解决。随着路径上信息素的增加,选择某个路径的概率越来越大,这样的结果可能导致早熟现象,不能达到全局最优解。

4 改进 *K*–means 蚁群算法

现有的 *K* –means 蚁群聚类算法在迭代的后期,路径上信息素逐渐积累,导致后面迭代差异性不大,可能导致得到的解不一定是全局最优解,易出现收敛于局部最优的情况。在此引入变异操作的概念。在一次迭代结束时根据数据对象的分类情况,随机选择分到同一类的数据对象中信息素最小的一个数据对象进行变异操作,若变异操作后通过计算目标函数 F 值,和之前该类的 F 值进行比较,如果小于原先的 F 值则进行变异操作,反之不进行。在变异之后更新变异点到聚类中心的信息素值。假设有 8 个数据对象的数据集,聚类结果分成 3 类。分类情况假设为 1 1 2 2 3 3 2,随机选择三类中的一类,然后获得该类中各个数据元素到聚类中心的信息素,选择信息素最小的一个数据进行变换。如果开始该数据分给了第 3 类,则改为分到第 1 和第 2 类,分别计算 F 值,如果获得的 F 值比原先值小,则进行变异,同时更新信息素值;否则,不进行。

改进 *K* –means 蚁群聚类算法详细描述如下:

(1)初始化。随机选取 k 个初始聚类中心 $c_1, c_2, \cdots, c_k$;

(2)通过 *K* –means 算法预处理得到聚类中心和簇;

(3)初始化 ρ、Q、q_0 的值,蚂蚁的个数 n,最大迭代数 N_c,根据步骤(2)得到聚类中心和簇初始化信息素 $\tau_{ij}(i = 1,2,\cdots,n; j = 1,2,\cdots,k)$;

(4)计算各数据对象到各聚类中心的距离 d_{ij} 及启发函数 η_{ij};

(5)随机产生概率 p,如果 p 值小于给定的 q_0,则按路径上信息素大小原则把数据样本分配到某类;如果大于 q_0,则计算蚂蚁转移概率,选择聚到某类;

(6)比较各个蚂蚁的目标函数值,保存最好的解;

(7)对每只蚂蚁得到的解进行变异操作,如果 F 值小于原来值则保存,否则不保存;

(8)按照公式(4)更新信息素;

(9)满足结束条件或迭代次数,输出最优解,否则转到步骤(2)继续运行。

改进的 *K* –means 蚁群聚类算法增加了算法迭代后期的随机性,扩大了聚类结果的取值范围,一定程度上改善了后期出现的停滞和早熟现象。

5 仿真实验

实验数据来源 UCI 数据库中的数据集 Iris, Wine, Glass。实验比较目标函数 F 的值,F 为各簇中所有点到其聚类中心的欧氏距离的和。F 值能够直接反映聚类结果的好坏,其值越小,则表达聚类结果越紧凑、越独立[14]。

实验采用 *K* –means 算法、标准的蚁群聚类算法(Ant Colony Clustering Algorithm, ACCA)、原有的 *K* –means 蚁群聚类算法(*K* –Means Ant Colony Clustering Algorithm, KMACCA)和文中的改进 *K* –means 蚁群聚类算法(Changed K–Means Ant Colony Clustering Algorithm, CKMACCA)进行上述数据集的比较。记录最优、最差和平均值 100 次。实验运行环境为 Inter Core2, 2.20 GHz CPU, C++编程。实验参数 $\rho = 0.99$, $Q = 100$, $q_0 = 0.98$,迭代次数 $N_c = 200$,蚂蚁数量 $n = 50$。结果如表 1 所示。

从表 1 的结果可以看出改进的算法比原有的 *K* –means 算法稍有进步,尤其在平均值方面。主要原因是改进算法增加了变异操作改善了早熟现象,并增加

了随机的搜索,所有能够在有限的迭代次数内产生全局最优解。但在表 2 中运行时间的比较时发现,由于我们增加了变异的操作使算法的运行时间相比较原有的 K -means 蚁群聚类算法运行时间较高。但相比较基本的蚁群聚类算法,运行时间还是较低一些,保留了原有 K -means 蚁群聚类算法加快收敛速度的优点。

表 1　4 种算法的 F 值对比

算法	数据	最优值	平均值	最差值
K -means	Iris	97.22	102.78	123.97
	Wine	16 530.53	16 913.26	18 437.82
	Glass	213.22	225.10	258.83
ACCA	Iris	97.22	101.69	125.40
	Wine	165 30.53	16 932.10	19 208.37
	Glass	213.22	233.93	261.59
KMACCA	Iris	97.22	99.84	123.85
	Wine	165 30.53	16 887.46	16 523.53
	Glass	213.22	220.12	241.36
CKMACCA	Iris	97.19	98.76	121.16
	Wine	165 30.53	16 533.29	16 592.31
	Glass	213.22	217.52	237.83

表 2　算法运行时间对比

算法	运行时间		
	Iris	Wine	Glass
KMACCA	1.000 0	1.000 0	1.000 0
CKMACCA	1.607 9	1.834 2	1.528 9
K -means	0.772 7	0.945 1	0.965 1
ACCA	1.742 9	1.862 4	1.540 1

注:运行时间以 KMACCA 算法为基准,表中结果为其他算法运行时间除以 1.000 的结果。

6　结束语

文中对现有的 K -means 蚁群聚类算法进行改进,增加聚类结果的变异操作,避免了迭代后期停滞不前,收敛局部最优情况,但同时也出现运行时间较高的问题。进一步需要解决的问题是:如何优化算法使时间复杂度降低,从而使运行效率更好;利用 K -means 算法聚类结果来初始化路径信息素怎样设置更合理、更有效。

参考文献:

[1] 贾　冀. 基于聚类的图像分割与配准研究[D]. 西安:西安电子科技大学,2013.

[2] 赖桃桃,冯少荣. 聚类算法中的相似性度量方法研究[J]. 心智与计算,2008,2(2):176-181.

[3] 贺　玲,吴玲达,蔡益朝. 数据挖掘中的聚类算法综述[J]. 计算机应用研究,2007,24(1):10-13.

[4] 冯　波,郝文宁,陈　刚,等. K-means 算法初始聚类中心选择的优化[J]. 计算机工程与应用,2013,49(14):182-185.

[5] Colorini A,Dorigo M,Maniezzo V,et al. Distributed optimization by ant colonies[C]//Proceedings of 1st European conf on artificial life. Paris:Elsevier Publishing,1991:134-142.

[6] Shelokar P S,Jayaraman V K,Kulkarni B D. An ant colony approach for clustering[J]. Analytica Chimica Acta,2004,509(2):187-195.

[7] Deneubourg J L,Goss S,Franks N,et al. The dynamics of collective sorting:robot-like ant and ant-like robot[C]//Proceedings of the first conference on simulation of adaptive behavior:from animals to animals. Cambridge,MA:MIT Press,1991:356-365.

[8] Goss S,Aron S,Deneuborug J L,et al. Self-organized shortcuts in the argentine ant[J]. Naturwissenschaften,1989,76:579-581.

[9] 朱　峰,陈　莉. 一种改进的蚁群聚类算法[J]. 计算机工程与应用,2010,46(6):133-135.

[10] 杨　燕,靳　蕃,Mohamed Kamel. 一种基于蚁群算法的聚类组合方法[J]. 铁道学报,2004,26(4):64-69.

[11] 张　蕾,曹其新,李　杰. 一种新型的自适应蚁群聚类算法[J]. 上海交通大学学报,2009,43(6):906-909.

[12] 贾瑞玉,邢　猛,徐庆鹏,等. 一种动态调整的蚁群聚类算法[J]. 计算机技术与发展,2009,19(2):145-147.

[13] 邢洁清,朱庆生,郭　平. 蚁群聚类方法组合方法的研究[J]. 计算机工程与应用,2009,45(18):146-148.

[14] 贾瑞玉,王会颖. 基于改进蚁群算法的聚类分析[J]. 计算机应用与软件,2010,27(12):97-100.

一种改进的K-means蚁群聚类算法

作者： 李振， 贾瑞玉， LI Zhen， JIA Rui-yu
作者单位： 安徽大学 计算机科学与技术学院,安徽 合肥,230601
刊名： 计算机技术与发展 ISTIC
英文刊名： Computer Technology and Development
年，卷(期)： 2015,25(12)