一种基于图论的加权聚类融合算法*

谢岳山[1a]，樊晓平[1a,2]，廖志芳[1b,†]，尹红练[1b]，罗 浩[1a]
（1. 中南大学 a. 信息科学与工程学院；b. 软件学院，长沙 410075；2. 湖南省财政经济学院，长沙 410205）

摘 要：现有聚类融合算法对混合属性数据进行处理的效果不佳，主要是融合后的结果仍存在一定的分散性。为解决这个问题，提出了一种基于图论的加权聚类融合算法，通过对数据集聚类得到聚类成员后，利用所设计的融合函数对各个数据对象赋予权重，同时通过设置各个数据对间边的权重来确定数据之间的关系，得到带权最近邻图，再用图论的方法进行聚类。实验表明，该算法的聚类精度和稳定性优于其他聚类融合算法。

关键词：聚类融合；融合函数；混合属性；图论；加权
中图分类号：TP301.6 **文献标志码**：A **文章编号**：1001-3695(2013)04-1015-02
doi:10.3969/j.issn.1001-3695.2013.04.014

Weighted cluster fusion algorithm based on graph

XIE Yue-shan[1a], FAN Xiao-ping[1a,2], LIAO Zhi-fang[1b,†], YIN Hong-lian[1b], LUO Hao[1a]
(1. *a. College of Information Science & Engineering, b. School of Software, Central South University, Changsha* 410075, *China*; 2. *Hunan University of Finance & Economics, Changsha* 410205, *China*)

Abstract: The results of the existing cluster fusion algorithms are usually not so good when they process the mixed attributes datas, the main reason is that the results of the algorithms are still dispersed. To solve this problem, this paper presented a new weighted cluster fusion algorithm based on graph theory. It first clustered the datasets and got cluster members, and then set weights to each data object with a proposed fusion function, and determined the relationship between the data-pair by setting weights to the edges between them, so it could get a weighted nearest neighbor graph. At last it did a last-clustering based on graph theory. Experiments show that the accuracy and stability of this cluster fusion algorithm is better than other clustering fusion algorithms.

Key words: cluster fusion; fusing function; mixed attributes; graph theory; weighted

聚类融合就是将传统的聚类算法与融合方法结合起来对数据进行聚类处理，其基本思想是通过对一组对象集采用不同的算法或使用不同的初始条件来进行多次聚类，将所得到的聚类结果用一种融合方法进行合并[1]，从而得到比单一算法更优的结果。聚类融合方法具有处理高维度、海量、多种数据类型等数据特征的数据时精度较高的特点。Topchy 等人[2]经过分析验证后指出，聚类融合方法的鲁棒性、适用性、稳定性及并行性和可扩展性等都比较优越。目前，聚类融合是聚类研究中的热点。例如，Fred[3]用数据点之间的相似度作为元素建立共生矩阵，设置一个阈值，矩阵中大于阈值的两个点即认为属于最终聚类结果中的一类。Ayad 等人[4]提出了三个基于超图的方法，即 CSPA、HGPA 和 MCLA，将共生矩阵转换为超图，使用无向图割算法 METIS 产生融合结果。然而这种基于图类的融合算法是基于无权重的共生矩阵，其劣质聚类结果会对融合结果的质量产生影响。基于以上分析，本文提出了一种基于图论的加权聚类融合方法——CFBG(cluster fusion method based-graph)。

1 CFBG 算法

本章首先建立 CFBG 算法的框架，然后依据该算法框架依次阐述算法的思路和实现方法，最后给出一个在模拟数据上的实例分析。

1.1 CFBG 算法框架

首先给定一个有 n 个数据点的数据集 $X=\{x_1,x_2,x_3,\cdots,x_n\}$，对该数据集用 r 次聚类算法进行聚类得到 r 个聚类结果，即结果集为 $R=\{\lambda^{(1)},\lambda^{(2)},\lambda^{(3)},\cdots,\lambda^{(r)}\}$，其中 $\lambda^{(i)}(i=1,2,3,\cdots,r)$ 为执行第 i 次聚类算法得到的聚类结果；然后设计一种融合函数 Γ，对这 r 个聚类成员进行合并，再用图论的方法进行聚类处理；最后得到一个最终的聚类结果 λ。算法的整体处理框架如图 1 所示。

1.2 聚类成员的产生

聚类成员的产生可以通过选择不同的初始值、选择不同的算法、选择不同的对象子集或选择不同的特征子集投影到数据子空间等多种方法[5]。考虑到本文要处理的是混合属性数据，因此采用能处理此种类型数据的 k-prototype 算法作为聚类算法，并随机选择不同的初始点和聚类个数，以此产生所需的聚类成员。

对于具有 n 个数据点的数据集 $X=\{x_1,x_2,\cdots,x_n\}$，用 C_1，$C_2,\cdots,C_m$ 分别表示数据对象 $x_i(1\leqslant i\leqslant n)$ 的 m 个属性。用 k-prototype 聚类算法将数据集划分为 k 个簇（k 为指定的正整数）。k-prototype 算法的描述如下：

收稿日期：2012-08-06；**修回日期**：2012-11-07 **基金项目**：国家科技支撑项目计划资助项目(2012BAH08B01)；湖南省自然科学基金资助项目(12JJ3074)

作者简介：谢岳山(1965-)，男，高级经济师，博士，主要研究方向为数据挖掘在审计中的应用；樊晓平(1961-)，教授，博导，主要研究方向为智能信息处理；廖志芳(1968-)，女(通信作者)，副教授，博士，主要研究方向为数据挖掘(zfliao@csu.edu.cn)；尹红练(1990-)，女，本科，主要研究方向为数据挖掘.

a)随机选择 k 个数据点作为簇的初始点,即簇原型;

b)根据对象间的相异度将每个对象分配到各自最近的簇,分配后即更新簇原型;

c)重新测试对象和簇原型的相异度,若发现该对象最近的原型不是当前的簇原型,而是其他簇原型,则将该对象重定位并更新簇原型;

d)重复 c),直到整个数据集依次测试一遍后没有对象再需要改变簇。

其中对象间相异度的计算方法可作如下描述:设对象 x、$y \in X$ 都具有 m 个属性 $C_1^r, C_2^r, \cdots, C_p^r, C_{p+1}^c, \cdots, C_m^c$,其中上标为 r 的是数值型属性,上标为 c 的是分类型属性,X 和 Y 的相异度计算公式为

$$d(x,y) = \sum_{j=1}^{p} (x_j - y_j)^2 + r \sum_{j=p+1}^{m} \delta(x_j, y_j) \tag{1}$$

其中:$\delta(x_j, y_j) = \begin{cases} 0 & x_j = y_j \\ 1 & x_j \neq y_j \end{cases}$。式(1)中的第一项为数值属性的欧几里德平方距离,第二项为类别属性的匹配相异度,参数 r 用于避免结果偏向于数值属性或类别属性,取值为 $0 \leqslant r \leqslant 1$。对于聚类成员个数的选取现在并没有明确的标准,Kuncheva 等人[6]详细讨论了聚类成员的个数对聚类融合的影响,本文不再讨论。

1.3 融合函数的设计

在对对象集 X 进行 r 次聚类后得到结果集 $R = \{\lambda^{(1)}, \lambda^{(2)}, \cdots, \lambda^{(r)}\}$,第 i 个聚类结果 $\lambda^{(i)}$ 中簇的个数用 k_i 表示。构造一种融合函数 WSnnG = (V, E),其中 $V = \{v_1, v_2, \cdots, v_n\}$ 为带权顶点集,E 为顶点之间带有权重的边的集合。用该融合函数进行融合后,再用基于图论的 METIS 算法进行聚类,得到最终的聚类结果。WSnnG 的构造主要分为边权值的确定和顶点权值的确定两部分。

边权值的分配可由对象对 i 与 j 之间的共享最近邻来决定。两个对象都在对方的最近邻列表中,共享最近邻就是它们共享的近邻个数。两个对象 i 与 j 之间边的权值可由式(2)给出。

$$\sigma_{ij} = 2 \times \frac{|A \cap B|}{|A| + |B|} \tag{2}$$

其中:A 表示对象 i 的最近邻集;B 表示对象 j 的最近邻集。

顶点权值的分配可通过平衡因子 l 与共享最近邻数 θ_i 的比率计算。其计算式为

$$w_i = \frac{l}{\theta_i} \tag{3}$$

其中:w_i 表示 i 的顶点权值;θ_i 表示对象 i 与其他对象的共享最近邻数的总和,即 $\theta_i = \sum_{j=1}^{m} \theta_{ij}$。$l$ 取 $l = n$,n 为总的样本数据集。

构造好带权最近邻图后,采用基于图论的 METIS 算法对图进行聚类划分,得到最终的聚类结果。该算法的主要原理是通过最小化边的切割,使得顶点的权值被平均分配到不同的区域中。

1.4 人工数据实例分析

给定一个数据集 $X = \{x_1, x_2, \cdots, x_{10}\}$,由 10 个数据点对象组成,它的自然簇结构为 $C_0 = \{\{x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8\}, \{x_9, x_{10}\}\}$。用三次聚类方法对其进行聚类处理,得到 $C_1 = \{\{x_1, x_2, x_3, x_4, x_5, x_6, x_7\}, \{x_8, x_9, x_{10}\}\}$、$C_2 = \{\{x_1, x_2, x_3, x_4, x_5, x_6, x_8\}, \{x_7, x_9, x_{10}\}\}$、$C_3 = \{\{x_1, x_2, x_3, x_4, x_5, x_7, x_8\}, \{x_6, x_9, x_{10}\}\}$ 这三个聚类成员,成员结果集用 $C = \{C_1, C_2, C_3\}$ 表示。对该成员结果集用 WSnnG 算法进行融合,得到带权最近邻图,如图 2 所示。

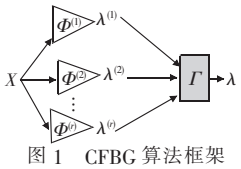

图 1 CFBG 算法框架

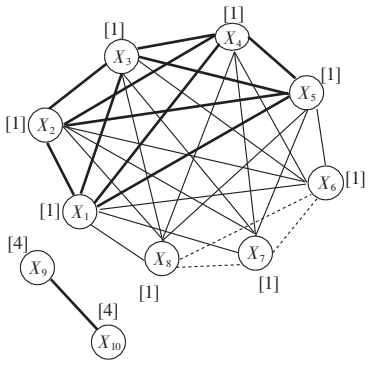

图 2 WSnnG 实例图

图 2 中边的粗细反映了数据点之间的紧密程度,顶点的权值分配有利于簇之间保持平衡性。图 2 包含了所有三个聚类结果,表示的划分与自然簇划分十分接近。

2 实验分析

为了检验 CFBG 算法的精度和稳定性,本文通过实验分别对其进行测试。首先在两个实际数据集上运行该算法以及另外两种既有的聚类算法,对它们的聚类精度进行对比;然后在同一个数据集上运行 CFBG 和另一种算法各若干次,对它们聚类结果的稳定性进行对比。

2.1 Mobile 数据集

选择 Mobile 实际数据集对 CFBG 算法进行测试。该数据来源于中国移动某分公司,记录手机充值卡的销售情况。该数据集包含 21 个属性,其中数值属性 15 个,类属性 6 个,共有 815 条记录,不包含丢失数据。为了验证算法的性能,采用本文提出的 CFBG、k-prototype 以及 CSPA 融合算法分别对数据集聚类,对这三种算法得到的聚类结果的精度进行了对比,如图 3 所示。聚类成员数目(即聚类算法执行的次数)从 2 依次增加到 8,对每个聚类成员数都进行算法的精度比较,k-prototype 算法平均精度为 0.59,CSPA 算法平均精度为 0.66,CFBG 算法平均精度为 0.73。

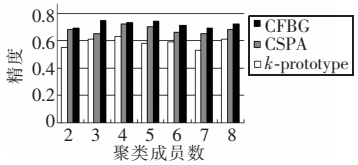

图 3 Mobile 数据集中三种算法精度的比较

2.2 BCA 数据集

BCA(bank credit approval)数据集记录了某银行不同的信用卡批准信息,包含 19 个属性,其中数值属性 10 个,类属性 9 个,共有 1 027 条记录,去掉 31 个丢失数据,最终采用 996 条记录。三种聚类算法在不同聚类成员数目下的聚类精度对比结果如图 4 所示。聚类结果成员数目从 2 到 8,分别对每个聚类成员数进行算法精度比较。k-prototype 算法平均精度为 0.76,CSPA 算法平均精度为 0.84,CFBG 算法平均精度为 0.87。

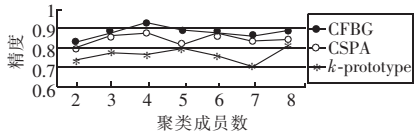

图 4 BCA 数据集中三种算法精度的比较

为了验证算法的稳定性,在 BCA 数据集上,(下转第 1034 页)

表 1 三种算法的优化结果比较

测试函数	最优值	ICSA		DE-ICSA		PSO-ICSA	
		平均值	标准差	平均值	标准差	平均值	标准差
F_1	0	8.7e-4	6.0e-5	4.0e-5	3.0e-6	1.6e-4	9.1e-6
F_2	0	6.6e-4	4.0e-5	9.0e-5	2.7e-5	2.5e-4	3.2e-5
F_3	0	3.7e-3	1.9e-4	9.5e-4	6.3e-5	1.0e-3	7.8e-4
F_4	0	5.4e-3	3.4e-4	8.2e-4	9.5e-5	9.2e-3	6.3e-4

测试 2 计算时间。计算时间是衡量算法性能的一个重要指标。仿真环境采用 MATLAB r2009a,联想 ThinkPad 2.3 GHz CPU、2 GB 内存。函数 $F_1 \sim F_4$ 的目标值设定为 10^{-4}。计算时间测试结果如表 2 所示。从表中可以看出,达到同样的目标值,DE-ICSA 的平均计算时间最少。

表 2 计算时间比较

测试函数	目标值	ICSA/s	DE-ICSA/s	PSO-ICSA/s
F_1	10^{-4}	19.88	1.79	2.31
F_2	10^{-4}	16.52	3.09	3.51
F_3	10^{-4}	26.14	4.69	5.35
F_4	10^{-4}	32.64	5.74	6.24

测试 3 成功率。成功率是指算法找到设定目标值的次数与总运行次数的比率。函数 $F_1 \sim F_4$ 的目标值设定为 10^{-4},各独立运行 40 次,成功率如图 2 所示。从图中可以看出,DE-ICSA 的成功率最高。

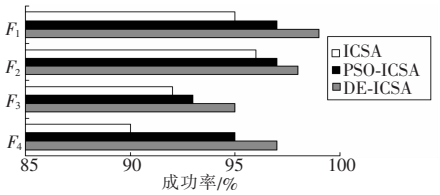

图 2 三种算法的成功率比较

以上标准测试函数的优化结果清楚地表明,DE-ICSA 具有更好的收敛性能。DE-ICSA 不仅具有更高的收敛速度,得到最优解的精度也更高。

5 结束语

本文针对ICSA的搜索能力弱、易陷入早熟收敛的缺点,提出了一种 DE-ICSA 优化方法,该方法利用 DE 提高 ICSA 的抗体亲和度,对 DE-ICSA 的收敛性进行了分析。为了测试该算法的有效性,将该算法应用于函数优化问题中,仿真结果表明,该方法具有更高的收敛速度和收敛精度。

参考文献:

[1] 罗一丹,蔡自兴,龚涛,等. 一种新的免疫进化算法在函数优化中的应用[J]. 计算机工程与科学, 2008,30(8):49-52.

[2] CHENG Li-jun, DING Yong-sheng, HAO Kuang-rong, *et al*. An ensemble kernel classifier with immune clonal selection algorithm for automatic discriminant of primary open-angle glaucoma[J]. Neurocomputing,2012,83(4):1-11.

[3] 魏圆圆,唐超礼,黄友锐. 自适应克隆选择算法及其仿真研究[J]. 模式识别与人工智能,2009,22(2):202-207.

[4] PRICE K V, STORN R M, LAMPINEN J A. Differential evolution: a practical approach to global optimization[M]. Berlin: Springer, 2005.

[5] WANG Yong, CAI Zi-xing, ZHANG Qing-fu. Differential evolution with composite trial vector generation strategies and control parameters [J]. IEEE Trans on Evolutionary Computation, 2011, 15(1): 55-66.

[6] HANSEN N, OSTERMEIER A. Completely derandomized self-adaptation in evolution strategies [J]. Evolutionary Computation, 2001, 9(2): 159-195.

[7] De CASTRO L N, Von ZUBEN F J. The clonal selection algorithm with engineering applications[C]//Proc of GECCO. 2000: 36-37.

[8] 张著洪,黄席樾. 一种新的免疫算法及其在多模态函数优化中的应用[J]. 控制理论与应用,2004,21(1):17-21.

[9] KENNEDY J, EBERHART R C. Particle swarm optimization [C]// Proc of IEEE International Conference on Neural Networks. 1995: 1942-1948.

[10] TRELEA I C. The particle swarm optimization algorithm: convergence analysis and parameter selection [J]. Information Processing Letters, 2003, 85(6): 317-325.

(上接第 1016 页)将聚类的簇数固定为三个,分别随机运行 k-prototype 算法和 CFBG 算法各 7 次,两种算法每次运行的精度对比结果如表 1 所示。

表 1 k-prototype 与 CFBG 算法的稳定性比较

算法	单次精度						
	1	2	3	4	5	6	7
k-prototype	0.74	0.83	0.68	0.82	0.79	0.75	0.80
CFBG	0.84	0.85	0.82	0.83	0.83	0.86	0.84

由表 1 可知,CFBG 聚类融合算法的精度相对较高,并且在七次实验中的聚类精度变化不大,最高精度和最低精度仅相差 0.04,比较稳定;而 k-prototype 算法的精度浮动较大,且其最高精度与最低精度相差达到 0.15。其原因是 k-prototype 算法受到初始点选择的影响较大,而 CFBG 算法因为融合了多次聚类结果,因此聚类性能相对较高且结果比较稳定。

3 结束语

要衡量一个聚类算法性能的优劣,精度和稳定性是两个重要指标,也是人们在应用中所关注的两个重要方面。本文提出的 CFBG 算法结合了聚类分析和融合技术,并加入图论的处理方法,使得算法的精度和稳定性得到了明显提高,并得到了比单一算法更为优越的结果。现在的聚类融合算法研究还不是十分成熟,有很多东西需要继续研究,如何合理地确定每个聚类成员的聚类个数、最终聚类的个数以及它们之间的关系,仍是一个值得探讨的问题。

参考文献:

[1] STREHL A, GHOSH J. Cluster ensembles: a knowledge reuse framework for combining multiple partitions [J]. Journal of Machine Learning Research, 2003, 3(3): 583-617.

[2] TOPCHY A, JAIN A K, PUNCH W. A mixture model for clustering ensembles[C]//Proc of the 4th SIAM International Conference on Data Mining. 2004: 379-390.

[3] FRED A L. Finding consistent clusters in data partitions [C]//Proc of the 2nd International Workshop on Multiple Classifier Systems. 2001: 309-318.

[4] AYAD H, KAMEL M. Finding natural clusters using multi-clusterer combiner based on shared nearest neighbors [C]//Proc of the 4th International Workshop on Multiple Classifier Systems. 2003: 166-175.

[5] 阳琳赟,周海京,卓晴,等. 基于属性重要性的加权聚类融合[J]. 计算机科学 2009,36(4):243-239.

[6] KUNCHEVA L I, HADJITODOROV S T. Using diversity in cluster ensembles[C]//Proc of IEEE International Conference on Systems, Man and Cybernetics. 2004: 1214-1219.

[7] YANG Lin-yun, ZHOU Hai-jing, ZHUO Qing, *et al*. Weighted cluster ensemble based on significance of attribute[J]. Computer Science, 2009, 36(4): 243-249.