第41卷 第12期 西 安 交 通 大 学 学 报 Vol.41 №12
2007年12月 JOURNAL OF XI′AN JIAOTONG UNIVERSITY Dec. 2007

一种基于边界识别的聚类算法

张选平，祝兴昌，马琮
（西安交通大学计算机科学与技术系，710049，西安）

摘要：针对基于密度的聚类算法由高密度区到低密度区的处理顺序所带来的不能识别低密度对象类别的缺陷，通过对聚类过程中可能存在的边界识别进行讨论，提出了一种基于边界识别的聚类算法.该算法的思想是：同簇优先权高于密度优先权，即在选择下一个对象进行聚类时，在已聚类的对象中优先选择同一簇的对象，当对象沿某一方向扩展到达簇边界时停止扩展，转而向其他方向扩展，这种处理顺序能使得类别最大化.通过分析簇边界的密度变化特征，建立了边界识别准则，并根据该准则对数据进行聚类.通过在合成数据和美国加州大学提供的知识挖掘数据库数据集上的实验结果表明，所提算法能有效地处理低密度区域的数据，与识别聚类结构的对象排序算法相比，聚类效果可提高4%左右，而时间性能相当.

关键词：聚类算法；密度；边界识别
中图分类号：TP311.13 **文献标识码**：A **文章编号**：0253-987X(2007)12-1387-04

Clustering Algorithm Based on Boundary Identification

Zhang Xuanping, Zhu Xingchang, Ma Cong
(Department of Computer Science and Technology, Xi′an Jiaotong University, Xi′an 710049, China)

Abstract: Focusing on the default that in the clustering algorithm based on density the objects are processed from high-density area to low-density area, thereby, the objects with low density can not be identified, a novel clustering algorithm based on boundary identification is proposed through discussing the boundary identification existed in clustering process. The main idea of the algorithm is that the objects belonging to an accumulated cluster have higher priority than the density priority, i.e. objects belonging to the same accumulated cluster will be clustered first before next clustering is processed. When the object extension reaches the boundary of cluster, the extension is stopped and turns to other direction. This method can maximize each cluster. After analyzing the density features of the cluster boundary, a boundary identification rule is created and data is clustered according to it. Experiments with synthetic data set and UCI KDD data sets demonstrate that the proposed algorithm is specially suited for processing objects with low-density, and the new algorithm can improve the performance by 4% while keeping same time complexity compared to other algorithms.

Keywords: clustering algorithm; density; boundary identification

数据挖掘就是从大量的数据中发现隐含、先前未知、对决策有潜在价值规则的过程[1]，其中聚类分析是广为研究的问题之一.聚类算法通常包括层次聚类、分割聚类、基于密度的聚类、基于网格的聚类等.DBSCAN[2]、CURE[3]、CURD[4]等都是基于密度的聚类算法，其优点是能够有效地屏蔽异常数据(噪声)对算法的影响，但由于它们对数据处理的顺序有所不同，所以仍存在噪声误判的情况，OPTICS(Ordering Points to Identify the Clustering Structure)算法[5]也存在这类问题.

收稿日期：2007-03-02. 作者简介：张选平(1963～)，男，副教授. 基金项目：国家自然科学基金资助项目(60673087).

本文提出了一种基于边界识别的聚类算法，即在密度定义的基础上，通过考察簇边界区域的密度变化情况，建立了簇边界识别准则，并根据该准则进行聚类.

1 OPTICS算法及相关概念

OPTICS算法是一种基于密度的聚类算法，它克服了DBSCAN算法过度依赖输入参数的弱点，但二者相同的是它们均通过区域查询的方法实现对象处理的. 在对象处理过程中，算法将计算出对象的核心距离及可达距离[5]这2个参数以供聚类时使用.

OPTICS算法选取的下一个对象总是与已聚类的对象的可达距离最小，这会导致对象的处理顺序为高密度区域在先低密度区域在后，其优点是能够快速地识别出核心区域对象的类别，而不受对象形状、密度和大小的影响. 但是，低密度区域的对象总是延迟处理，而对象的处理顺序又是影响聚类的关键因素，所以OPTICS算法该算法不能很好地识别低密度区域的对象类别.

图1为OPTICS算法产生的数据点与类顺序之间的映射情况，从中看到有A、B、C 3个聚类. 由于这3个聚类之间有高密度“通道”，所以OPTICS算法会优先通过这些“通道”处理3个圆形区域内的数据，而圆形区域周围的对象将在最后被划分成D类并与C类归并在一起. 如果最后处理的对象距离C类中的对象过远，那么这些对象都将作为噪声[4]，其结果显然不能令人满意. 文献[6]曾针对该问题进行了讨论，并提出了先采用OPTICS算法处理所有对象，然后再用基于距离的K-平均算法重新划分低密度区域对象的方法，这样做虽然能够将低密度对象划分到正确的聚类中，但需要进行2次处理，计算量也很大.

2 边界识别算法

2.1 算法思想

机器聚类的最终目标就是达到手工聚类的效果，而手工聚类是根据簇边界来划分对象的，据此本文提出了基于簇边界进行对象聚类的思想，这种思想也可以表述为同簇优先权高于密度优先权，即在选择下一个对象进行聚类时，在已聚类的对象中优先选择同一簇的对象. 假设p_1和p_2属于同一簇，而p_3属于另外一簇，OPTICS算法对这3个对象的处理顺序为$p_1\rightarrow p_3\rightarrow p_2$，本文算法的处理顺序为$p_1\rightarrow p_2\rightarrow p_3$.

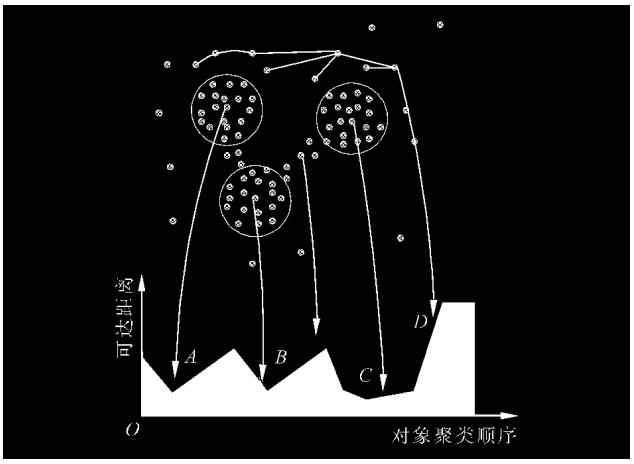


图1 OPTICS算法在低密度区域的缺陷

按照上述思想，当对象沿某一方向扩展到达簇边界时，应停止扩展，并转而向其他方向扩展，直到簇内的所有对象都扩展完毕，这种处理顺序能够使类别最大化. 若将对象间的距离信息看作局部信息，将簇信息看作全局信息，那么本文算法亦可结合局部和全局信息来选取下一个对象，避免了OPTICS算法只考虑局部信息的弊端.

2.2 边界识别准则

根据密度分布特征，可将簇分成3个部分：核心部分、中间部分和边界部分. 根据对象扩展时所经区域，可以将扩展分为4类，如表1所示.

表1 不同区域的密度变化特点

扩展类型	所经区域类型	密度变化特征
情况1	从簇边界向簇核心	不断升高
情况2	穿越簇核心	先升高后降低
情况3	从簇核心向簇边界	不断降低
情况4	穿越簇边界	先降低后升高

从表1可以看出，4种扩展类型都有各自的特征. 当对象扩展穿越簇边界时，对象密度的变化特点为先降低后升高. 如果在OPTICS算法的基础上考察3块连续区域对象的密度变化情况，就可以判断出当前对象是否在簇边界区域.

定义1 为探测边界区域而选中的进行邻域密度测试的对象为采样点.

定义2 采样点的ε邻域称为采样区域，即在边界区域探测时做密度测试的区域.

定义3 如果采样点o_i和o_j的ε邻域有重叠部分，即o_i和o_j之间的距离小于2ε，那么称o_i和o_j为彼此相邻采样点.

定义 4　假设采样点 o_i 和 o_j 是相邻采样点，如果先处理 o_i，且 o_i 的邻域密度大于 o_j 的邻域密度，则称 o_i 是 o_j 的前驱.

在进行对象聚类时，可以每隔 n 个对象进行一次采样，其中 n 是一个经验值，用来保证位于边界的采样区域彼此不会过度重合. 本文取 $n=P$（P 为一个自然数，是算法的一个输入参数），这样的设置可以基本保证采样点之间的距离大于 ε，这是在效率和效果之间的一个折中. 相邻采样点的定义能够保证区域密度之间的比较是有意义的.

假设有 3 个采样点 o_1、o_2 及 o_3，o_1 与 o_2 相邻，o_2 与 o_3 相邻，o_1 与 o_3 没有位置关系. 另假设 o_1、o_2 及 o_3 的 ε 邻域的密度变化情况如表 1 中的 4 种情况，那么表 2 给出了这 4 种情况下采样点的前驱情况.

表 2　不同情况下采样点的前驱情况

扩展类型	o_1	o_2	o_3
情况 1	/	/	/
情况 2	/	/	o_2
情况 3	/	o_1	o_2
情况 4	/	o_1	/

注：/表示前驱情况不明确.

通过表 2 可以看出，4 种情况的前驱都不相同. 情况 4，即扩展跨越边界时，采样点的前驱与另外 3 种情况有所不同，o_3 没有前驱，而它的前一个采样点 o_2 有前驱. 通过上面的分析，可以给出边界识别准则.

边界识别准则　如果当前采样点 o_3 没有前驱，而它至少有一个邻接采样点 o_2 有前驱 o_1，则 o_2 及其邻域可能位于边界区域，而 o_1 和 o_3 可能位于 2 个不同的聚类之中.

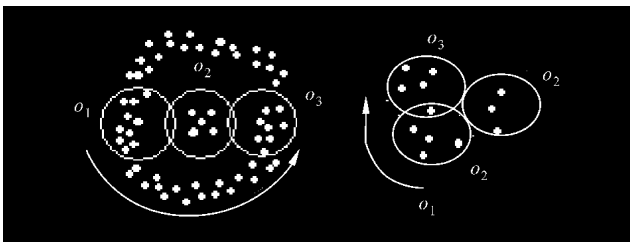
准则中提到 o_2 的邻域可能位于边界区域，这是因为存在着边界误判，下面将详细讨论这种情况及解决方法. 利用边界识别准则可以判断聚类的边界，进而可进行对象聚类.

2.3　边界区域的对象处理

当对象扩展到簇边界时，如果在其他方向上没有遇到边界，那么根据同簇优先权高于密度优先权的思想应当尽量向未遇到边界的方向扩展，并将遇到的边界区域的对象挂起，暂不处理.

挂起的对象在下面 2 种情况下可恢复为普通对象：①一个聚类区域被处理完毕后，所有的挂起对象恢复为普通对象；②对象被临时挂起，当它再次成为待处理对象时，经过检查可恢复为普通对象.

情况②针对的是聚类过程中可能出现的边界误判情况. 图 2 给出了 2 种边界误判情况，其中 o_2 本身不是边界，但 o_1、o_2 和 o_3 满足边界识别准则，于是 o_2 被误判为边界区域，o_3 邻域中的对象被错误地挂起. 出现这种情况的原因是在边界识别准则中没有规定 o_1 和 o_3 的关系，这样做是考虑到算法的执行效率. 尽管 o_3 周围的对象很可能被挂起，但由于扩展还可以向其他方向进行（如图 2 中箭头所示的方向），所以算法在扩展到挂起的对象时，就将其恢复为普通对象，从而解决了边界误判的情况，同时也不会将正确挂起的对象解挂，这是因为正确挂起的对象都与当前对象不在同一个聚类中. 若算法无法扩展，表明当前聚类已经完成，此时将所有挂起的对象全部恢复为普通对象，这正是情况①.



(a)情况①　　(b)情况②

图 2　对象临时挂起的示意图

2.4　算法描述

上面分析了算法的主要思想和对象的处理方法，下面给出算法的步骤描述.

步骤 1：按 OPTICS 算法规则处理当前对象.

步骤 2：判断当前处理对象 o_1 是否为采样点，如果不是下转步骤 7.

步骤 3：判断 o_1 的所有已经处理的相邻采样点的密度是否都小于 o_1，如果不是下转步骤 6.

步骤 4：判断 o_1 的所有相邻采样点的密度是否小于它们的所有相邻采样点的密度，如果不是下转步骤 6.

步骤 5：若 o_1 测试邻域中的对象全部挂起，下转步骤 7.

步骤 6：将 o_1 邻域内挂起的对象恢复为普通对象.

步骤 7：如果待处理集合只有挂起对象，将所有挂起的对象恢复为普通对象；如果待处理集合为空，处理结束.

步骤 8：按 OPTICS 算法规则从待处理集合中取下一个未挂起的处理对象，上转步骤 1.

步骤 1 是按照 OPTICS 算法处理当前对象，主

要包括以当前对象为核心进行区域查询，计算对象的核心距离和可达距离等；步骤3、步骤4是应用边界识别准则对 o_1 进行判断；步骤6是处理2.3节中提到的情况②；步骤7是处理2.3节中所提到的情况①.

3 实验结果与分析

3.1 时间性能分析

OPTICS算法的时间复杂度为 $O(n^2)$[5]，如果数据集中存在基于树的空间索引，那么算法的时间复杂度可以降为 $O(n\log n)$.

设边界识别算法的运行时间为 T_{CBI}，OPTICS算法的运行时间为 T_{OPTICS}. 由于边界识别算法是在OPTICS算法的基础上加入了簇边界的判断，所以运行时间应为OPTICS算法运行时间与边界判断时间之和，即 $T_{CBI}=T_{OPTICS}+T_B$，其中 T_B 为采样点的邻域测试时间. 由于边界识别算法每隔 P 个对象取一个采样点，并对这些采样点进行与OPTICS算法相同的区域查询操作，所以其边界判断时间通常为OPTICS算法的几分之一或几十分之一(需要根据 P 值来判断，实验中 $P=4$). 因此，边界识别算法与OPTICS算法的时间性能相似，也为 $O(n\log n)$.

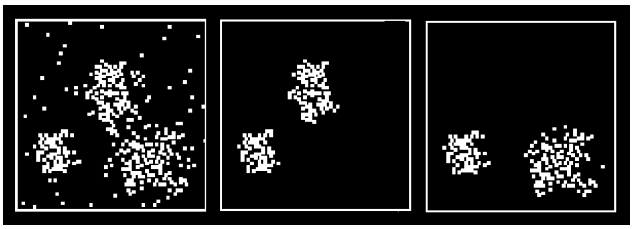
表3给出了实验结果，实验环境为PM1.6 GHz，1 GB内存，80 GB硬盘，操作系统为Windows2000. 程序采用C#语言编写，实验数据为人工合成的二维数据. 从表3可以看出，边界识别算法与OPTICS算法的运行时间相差无几，与理论分析的结果相吻合.

表3 算法的运行时间

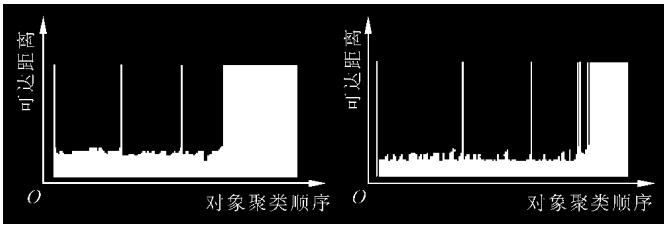
数据点数	运行时间/s	
	OPTICS算法	边界识别算法
2 000	7.4	7.4
3 000	18.0	18.3
4 000	34.0	34.3
5 000	61.0	61.5
6 000	92.3	98.5

3.2 聚类效果分析

图3a给出了一个由330个点组成的数据集，图3b和图3c分别是OPTICS算法和边界识别算法的聚类效果. 从图3b、图3c可以看出，边界识别算法和OPTICS算法均对数据形状不敏感. 另外，边界识别算法可以明显地降低噪声，且符合人的直觉. 图3d和图3e给出了2种算法的可达距离，从图中也可以看到边界识别算法的数据聚类效果更好.



(a)合成数据集　(b)OPTICS算法的聚类效果　(c)边界识别算法的聚类效果



(d)OPTICS算法的可达距离图　(e)边界识别算法的可达距离图

图3 合成数据集与聚类效果

采用美国加州大学知识挖掘数据库(UCI KDD)提供的zoo和ecoli数据集进行实验，并以F-Measure[7]值作为聚类效果的评价指标，实验结果如表4所示.

表4 2种算法的实际聚类结果

数据集	OPTICS算法			边界识别算法		
	查准率	查全率	F-Measure值	查准率	查全率	F-Measure值
zoo	0.90	0.88	0.89	0.92	0.96	0.94
ecoli	0.90	0.74	0.82	0.89	0.79	0.84

由表4可以看出，边界识别算法较OPTICS算法的F-Measure值提高了4%，主要原因是边界识别算法的查全率明显提高，查准率与OPTICS算法接近.

在边界识别算法的聚类结果中可能包含一些离群点，这是不可避免的，因为本文算法的思想是提高查全率而牺牲一些查准率. 另外，通过适当调整算法参数，边界识别算法可退化为OPTICS算法.

4 结束语

以OPTICS算法为基础，通过分析以簇边界进行聚类划分的手工聚类方法，提出了一种基于边界识别的聚类算法. 经考察簇边界的密度变化特征，分析相邻3个采样点邻域的前驱情况，提出了边界识

别准则. 理论分析和实验结果表明，本文算法较 OPTICS 算法具有相似的时间性能，查全率却明显提高，而聚类效果可提高 4%，有效避免了低密度对象被错误聚类的情况.

参考文献：

[1] Fayyad M, Piatetsky-Shapiro G, Smyth P. From data mining to knowledge discovery: an overview [C]// Advances in Knowledge Discovery and Data Mining. Menlo Park, USA: AAAI Press, 1996:1-34.

[2] Ester M, Kriegel H P, Sander J, et al. A density based algorithm for discovering clusters in large spatial databases with noise [C]// Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining. Oregon Portland: AAAI Press, 1996: 226-231.

[3] Guha S, Rastogi R, Shim K. CURE: an efficient clustering algorithm for large databases [C]// Proceedings of ACM SIGMOD International Conference on Management of Data. New York: ACM Press, 1998:73-84.

[4] 马帅，王腾蛟，唐世渭，等. 一种基于参考点和密度的快速聚类算法 [J]. 软件学报，2003，14(6):1089-1095.
Ma Shuai, Wang Tengjiao, Tang Shiwei, et al. A fast clustering algorithm based on reference and density [J]. Journal of Software, 2003, 14 (6):1089-1095.

[5] Ankerst M, Breunig M, Kriegel H P, et al. OPTICS: ordering points to identify the clustering structure [C]// Proceedings of ACM SIGMOD International Conference on Management of Data. New York: ACM Press, 1999:49-60.

[6] 孙学刚，陈群芳，马亮. 基于主题的 Web 文档聚类研究 [J]. 中文信息学报，2003，17(3):21-26.
Sun Xuegang, Chen Qunfang, Ma Liang. Study on topic-based Web clustering [J]. Journal of Chinese Information Processing, 2003, 17(3):21-26.

[7] Ayad H, Kamel M. Topic discovery from text using aggregation of different clustering methods [C]// Proceedings of the 15th Conference of the Canadian Society for Computational Studies of Intelligence on Advances in Artificial Intelligence. Heidelberg, Germany: Springer-Verlag, 2002:161-175.

（编辑　苗凌）