基于相对密度的聚类算法*)

刘青宝　邓　苏　张维明
(国防科学技术大学信息系统与管理学院　长沙 410073)

摘　要　基于密度的聚类算法因其抗噪声能力强和能发现任意形状的簇等优点，在聚类分析中被广泛采用，本文提出的基于相对密度的聚类算法，在继承上述优点的基础上，有效地解决了基于密度的聚类结果对参数值过于敏感、参数值难以设置以及高密度簇完全被相连的低密度簇所包含等问题。

关键词　聚类，K 近邻，聚类参数，相对密度

Relative Density-based Clustering Algorithm

LIU Qing-Bao　DENG Su　ZHANG Wei-Ming
(College of Information System and Management, National University of Defense Technology, Changsha 410073)

Abstract　With strong ability of discovery arbitrary shape clusters and handling noise, density based clustering is one of primary methods for data mining. This paper provides a clustering algorithm based on relative density, which efficiently resolves these problem of being very sensitive to the user-defined parameters and too difficult for users to determine the parameters.

Keywords　Clustering, K-nearest neighbors, Clustering parameter, Relative density

聚类分析是一种重要的人类行为，已经广泛地应用在许多领域，包括模式识别、数据分析、图像处理，以及市场研究。目前在文献中存在大量聚类分析算法，它们多数侧重于如何提高算法效率，而往往忽视了算法的有效性问题。聚类算法的有效性问题主要表现在三个方面：其一，聚类算法大多要求用户输入一定的参数，例如希望产生的簇的数目，而这些参数通常难以确定，特别是针对高维空间中稀疏分布的实际应用数据集，用户几乎无法给出合适的算法参数，因此非专业用户需要与数据分析专家密切配合才能保证获得理想的聚类结果，导致算法的使用极为不便；其二，聚类结果对于输入的参数值过于敏感，往往参数值的一些轻微变化却产生聚类结果的很大差异；其三，对于高维的实际应用数据集其数据分布往往是稀疏的、杂乱的，很难为算法选择全局的参数进行准确的聚类分析，使得聚类的质量难以保证。本文提出了基于相对密度的聚类算法，在继承了基于密度的聚类算法具有抗噪声能力强、能发现任意形状的簇等优点的基础上，有效地解决了基于密度的聚类结果对参数值过于敏感、参数值难以设置等问题，而且该算法支持密度分辨率，在聚类结果中可以区分密度等级不同的簇。

本文第 1 节介绍基于密度的数据聚类相关研究，并简要介绍本文提出的新算法思路及其与相关研究的异同；第 2 节给出基于相对密度的聚类算法所使用到的基本概念；第 3 节给出一个完整的基于相对密度聚类算法，详细解析算法的思想和执行过程；第 4 节介绍算法性能分析对比；最后总结本文的主要工作和贡献，并指出需要进一步研究和改进的工作。

1　相关研究

一个典型的基于密度聚类算法是文[1]中提出的 DBSCAN 算法，它将具有高密度的区域划分为类。DBSCAN 算法依赖两个参数实现聚类：对象的邻域半径 ε 和 ε 邻域内的最少对象数 MinPts。DBSCAN 通过检查数据集中每个点的 ε 邻域来寻找聚类：如果一个点 p 的 ε 邻域包含多于 MinPts 个点，则创建一个以 p 为核心对象的新类。然后，DBSCAN 反复寻找从这些核心对象直接密度可达的对象，当没有新点可被添加到任意类时，聚类过程结束，那些不属于任何类的点被标志为噪声。DBSCAN 可以在有噪声的情况下中发现任意形状的类，但是它留给用户决定的参数难以确定，而且它对参数值非常敏感，设置的细微不同即可能导致差别较大的聚类结果[2]。如图 1 所示，当所选的 MinPts 较大，ε 较小的时候，就会把 A、B 两个类全部判别成孤立点。

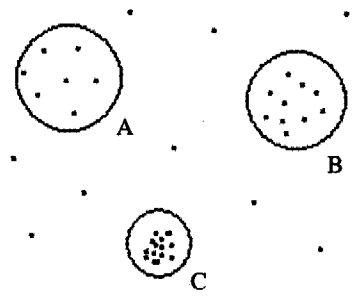

图 1　不同参数值的聚类结果

同时，考察 DBSCAN 可以发现，对于全局恒定的邻域半径 ε 和最少对象数 MinPts 值，高密度的聚类结果被完全包含在相连的低密度的聚类结果中。为解决这一问题，Mihael Ankerst 等人在文[3]中提出了 OPTICS(Ordering Points to Identify the Clustering Structure)聚类分析方法。它能为聚

*)国家自然科学基金(60172012)。刘青宝　博士生，副教授，主要研究方向为数据仓库技术和数据挖掘；邓　苏　教授，主要研究方向为指挥自动化、信息综合处理与辅助决策；张维明　博士生导师，教授，主要研究方向为军事信息系统、信息综合处理与辅助决策。

类分析计算一个簇次序，用以代表数据的基于密度的聚类结构，但它没有显式地为用户形成一个明确的数据聚类结果。分析上述基于密度的聚类分析算法，我们发现它们最大的缺点在于其采用绝对密度作为算法参数，而聚类的本质是使得同一个簇中的对象之间具有较高的相似度，而不同簇中的对象差别较大，即簇与簇之间的界线是相比较而划分的，簇与簇之间的密度差异是相对的。为此，本文提出一种基于相对密度的聚类算法 RDBClustering(Relative Density Based Clustering)，它能很好地发现任意形状、不同密度的类，并且有效地解决了高密度簇完全被相连的低密度簇所包含等问题。

2 相对密度有关概念

这里我们首先引入一些有关概念。

定义 1 对象 p 的 k 近邻距离[4]

对于任何正整数 k 和集合 D，对象 p 的 k 近邻距离定义为 p 到它的第 k 个最近邻居 o 的距离；其中 $p,o\in D$。对象 p 的 k 近邻距离用 $k\text{-}distance(p)$ 表示。

定义 2 对象 p 的 k 近邻邻居[4]

D 为给定数据集，对象 p 的 k 近邻邻居，记为 $N_{k\text{-}distance(p)}(p)$，定义为下面的集合：

$N_{k\text{-}distance(p)}(p)=\{q\in D\backslash\{p\}\mid d(p,q)<=k-distance(p)\}$；也称为 p 的 k-最近邻集合。

定义 3 对象 p 关于对象 o 的近邻距离[4]

设 $d(p,o)$ 为 d 和 o 之间的欧几里得距离。对象 p 关于对象 o 的近邻距离记为 $dist_{k\text{-}distance(o)}(p,o)$，定义如下：

$dist_{k\text{-}distance(o)}(p,o)=\max\{k\text{-}distance(o),d(p,o)\}$。

定义 4 对象 p 的近邻密度

给定集合 D，$p\in D$，$o\in N_{k\text{-}distance(p)}(p)$；$p$ 的近邻密度(Near Neighbors Density)记为 $nnd_{k\text{-}distance(p)}(p)$，定义如下：

$$nnd_{k\text{-}distance(p)}(p)=1/\left(\frac{\sum_{O\in N_{k\text{-}distance(p)}(p)} dist_{k\text{-}distance(o)}(p,o)}{|N_{k\text{-}distance(p)}(p)|}\right)$$

直观地，对象 p 的近邻密度是关于 p 的 k 近邻邻居 $N_{k\text{-}distance(p)}(p)$ 的平均近邻距离的倒数。

定义 5 对象 p 关于其 k 近邻邻居 $N_{k\text{-}distance(p)}(p)$ 的相对密度

给定集合 D，$p\in D$，p 关于其邻居 $N_{k\text{-}distance(p)}(p)$ 的相对密度(Relative Density)记为 $rd_{k\text{-}distance(p)}(p)$，定义如下：

$$rd_{k\text{-}distance(p)}(p)=\frac{\sum_{O\in N_{k\text{-}distance(p)}(p)}(nnd_{k\text{-}distance(o)}(o)/nnd_{k\text{-}distance(p)}(p))}{|N_{k\text{-}distance(p)}(p)|}$$

$rd_{k\text{-}distance(p)}(p)$ 反映了对象 p 的近邻密度与其邻居的近邻密度之间的差别，当 $rd_{k\text{-}distance(p)}(p)$ 接近 1 时，说明对象 p 与其邻居能很好地融为一体，在数据分布上密度十分接近。

定义 6 核心对象

给定阈值 $\delta>0$ 和集合 D，$p\in D$，若 $|rd_{k\text{-}distance(p)}(p)-1|<\delta$，则称该对象 p 为核心对象。

定义 7 核心集合

核心对象 p 的 k 近邻邻居中，由所有核心对象加 p 本身构成的子集称为核心对象 p 的核心集合，记为 $Core_{k\text{-}distance(p)}(p)$。若 p 非核心对象，则其核心集合无定义。

定义 8 对象 p 关于核心对象 q 密度可达

$p,q\in D$，对象 q 为核心对象，$pN_{k\text{-}distance(q)}(q)$，则称对象 p 关于核心对象 q 密度可达。

定义 9 核心对象 p 初始类

设 D 是数据集，$p\in D$，且 p 为核心对象，p 的初始类 C 是满足下列条件的数据集 D 的一个非空子集：

$\forall q\in D$，若 $\exists o\in Core_{k\text{-}distance(p)}(p)$，使得 q 关于核心对象 o 密度可达，那么 $q\in C$。

显然，核心对象 p 的初始类 C 非空，且有 $N_{k\text{-}distance(p)}(p)\subseteq C$。

定义 10 对象 p 关于类 C 的相对密度

给定集合 D，$C\subseteq D$，$p\in D$，p 关于类 C 的相对密度(*Relative Density*)记为 $rd_c(p)$，定义如下：

$$rd_c(p)=\frac{\sum_{O\in C}(nnd_{k\text{-}distance(o)}(o)/nnd_{k-distance(p)}(p))}{|C|}$$

$rd_c(p)$ 反映了对象 p 的近邻密度与类 C 中成员对象的平均近邻密度之间的差别。

3 基于相对密度的聚类算法

基于相对密度的聚类算法首先在数据集 D 中找到任意一个核心对象 p，求出 p 的核心集合，得到初始类 C；然后由初始类 C 开始进行类的扩展，直至没有任何对象可以归入该类；重新在 D 中寻找任意一个未归类的核心对象 q，重复上述过程，直至没有任何对象可以归入任何类，算法结束。

由初始类 C 的扩展过程分两步进行：首先，对 p 的核心集合进行扩展，得到类 C 的扩展核心集合；然后，根据关于扩展核心集合中核心对象的密度可达这一条件，对类 C 进行扩展，详细扩展方法见过程 ExpandCluster 中的伪码描述。

基于相对密度的聚类算法 RDBClustering 伪码描述如下：

```
RDBClustering(Set Setofpoint, int k, real δ)
//Setofpoint 是数据集，k 为最少的最近邻数，
//δ 为大于零的阈值
BEGIN
  REPEAT
    point=GetCorePoint(Setofpoint, k, δ);
//在未标识对象中找一个核心对象，
//若无核心对象，则 point 为 NULL
    IF point()NULL THEN
      Coreset=GetCoreSet(Setofpoint, point, k, δ);
//在未标识对象中得到 point 的核心集合
     clusterId=GetClusterId();
//得到新类标识号
    C=GetInitCluster(Setofpoint, point, CoreSet, k, clusterID);
//从未标识对象中得到 point 的初始类 C,
//并标识为 clusterID
     ExpandCluster(Setofpoint, C,CoreSet, k, δ);
//根据密度可达性，对类 C 进行扩展
    END IF
  UNTIL 没有任何类可以进行扩展;
END RDBKNN.
```

其中，ExpandCluster 过程伪码具体描述如下：

```
ExpandCluster (Set Setofpoint, Cluster C, Set CoreSet, int k, real δ)
BEGIN
  WHILE NOT CoreSet. empty() DO
    point=GetOutPoint(CoreSet);
//从 CoreSet 中取出一核心对象 point
   NewCoreset=GetCoreSet(Setofpoint, point, k, δ);
//在未标识的对象中得到 point 的核心集合
    FOR i FROM 1 TO NewCoreset. size DO
     object=NewCoreset. get(i);
     IF |rd_CoreSet(object)-1|<δ THEN
//对密度渐变所产生累加效应进行阈值检查
      CoreSet=CoreSet∪{object};
//object 加入类 C 的核心集合
    END IF;
   END FOR;
   C=C∪N_k-distance(point)(point);
//根据密度可达性，把核心对象 point 的 k 近//邻邻居 N_k-distance(p)(p)
扩展进簇 C
  END WHILE;
END ExpandCluster.
```

过程 ExpandCluster 中条件“$|rd_{CoreSet}(object)-1|<\delta$”用以对密度渐变所产生累加效应进行阈值检查，使得数据密度

连续渐变情况下，也能区分不同密度等级的类。

4 算法性能分析

这里从处理时间、参数选择和聚类质量3个指标分析基于相对密度的聚类算法RDBClustering的性能。

4.1 处理时间

算法运行过程中需多次使用核心对象的近邻密度nnd，算法实现时采用中间数据表M队核心对象的近邻密度nnd结果进行保存，做到一次计算多次查询。整个算法的时间主要由k近邻查询的时间和对中间数据表M的扫描时间两部分组成。k近邻查询的时间复杂度为$O(n \log n)$[5]，对中间数据表M的扫描时间复杂度为$O(n)$，因此与DBSCAN算法的时间复杂度($O(n \log n)$[2])相比属于同阶的，没有明显的时间差异。

4.2 参数选择

定理 设C是一个对象集，用dist-min表示C中对象的最小k近邻距离，即dist-min=min$\{dist_{k\text{-}distance(o)}(p,o) \mid p,o\in C,o\in N_{k\text{-}distance(p)}(p)\}$。相似地，设dist-max表示$C$中对象的最大$k$近邻距离，即dist-max=max$\{dist_{k\text{-}distance(o)}(p,o) \mid p,o\in C, o\in N_{k\text{-}distance(p)}(p)\}$。令$\varepsilon$=(dist-max/dist-min-1)，对于C中的所有对象p，若$N_{k\text{-}distance(p)}(p)\subseteq C$，而且$\forall q\in N_{k\text{-}distance(p)}(p)$，也都有$N_{k\text{-}distance(q)}(q)\subseteq C$，则$1/(1+\varepsilon)<=rd_{k\text{-}distance(p)}(p)<=(1+\varepsilon)$。

简单证明：对所有$o\in N_{k\text{-}distance(p)}(p)$，$dist_{k\text{-}distance(o)}(p,o)>=$dist-min。那么依定义4可知对象$p$的近邻密度$nnd_{k\text{-}distance(p)}(p)$小于等于1/dist-min。另一方面，$dist_{k\text{-}distance(o)}(p,o)<=$dist-max，因此，$nnd_{k\text{-}distance(p)}(p)>=1/$dist-max。设$q\in N_{k\text{-}distance(p)}(p)$。按照上面的讨论，$q$的近邻密度$nnd_{k\text{-}distance(q)}(q)$也在1/dist-max和1/dist-min之间。因此，按照定义5可知对象p相对密度$rd_{k\text{-}distance(p)}(p)$满足：dist-min/dist-max$<=rd_{k\text{-}distance(p)}(p)<=$dist-max/dist-min。即$1/(1+\varepsilon)<=rd_{k\text{-}distance(p)}(p)<=(1+\varepsilon)$。

定理的直观解释如下：C对应于一个类，我们考虑对象p，p在类内深处，即p任一k近邻邻居q都在C中，并且，q所有的k近邻邻居也在C中，对这样的类C深处对象p，p的$rd_{k\text{-}distance(p)}(p)$值是有一定范围的。应用定理能够得出结论：类$C$的核心对象$p$的$rd_{k\text{-}distance(p)}(p)$值接近于1。定理为算法参数$\delta$的选择限定了取值范围，从而不像DBSCAN算法中参数Eps那样取值具有太多的盲目性和试探性。

4.3 聚类质量

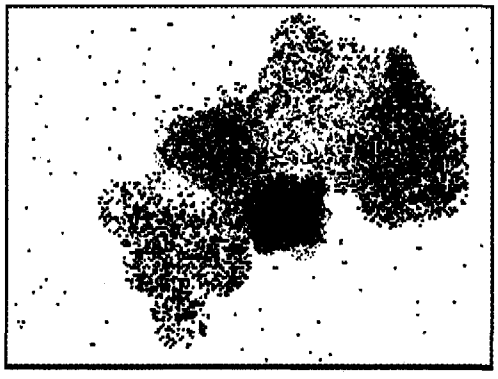

图2 实验数据集

基于相对密度的聚类算法RDBClustering像算法DBSCAN一样可以在有噪声的数据集中发现任意形状的簇，并且有效地解决了算法DBSCAN所存在的问题：高密度的聚类结果被完全包含在相连的低密度的聚类结果中。对于如图2所示的数据集，算法RDBClustering可以聚类出如图3所示的结果，如实地反映了数据集中数据的分布情况。

图3 算法RDBClustering的聚类结果

而算法DBSCAN在设置MinPts较大，ε较小的时候会出现如图4所示的结果，无法聚类出密度相对较低的簇。但若ε作适当的放大，却区分不出密度等级不同的簇，得到如图5所示的结果。

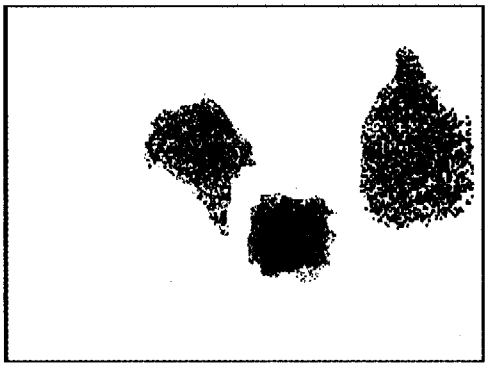

图4 算法DBSCAN的聚类结果(偏小)

图5 算法DBSCAN的聚类结果(合适)

小结 基于相对密度的聚类算法RDBClustering有效地解决了参数的设置和算法对于参数过于敏感的问题，同时也成功地解决了传统的基于密度聚类算法所存在的高密度聚类结果被包含在相连的低密度聚类结果中的问题。聚类算法RDBClustering具有以下特点：能发现任意形状的聚类；处理噪声数据的能力非常强；输入参数较易确定；能区分不同密度等级的簇。

算法的不足之处：必须用簇中的所有点来表示聚类形成

的任意形状，并要求获得整个数据集的全局信息，这在内存有限情况下对动态数据集进行聚类是难以适用的。如何在有限内存情况下对动态数据集进行增量式聚类是我们下一步研究工作的方向。

参 考 文 献

1 Ester M, Kriegel H -P, Sander J, et al. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In: Proc. 2nd Int Conf on Knowledge Discovery and Data Mining, Portland, OR, 1996. 226～231

2 Han Jiawei, Kamber M, Fan Ming, et al. 数据挖掘：概念与技术. 北京：机械工业出版社，2001

3 Ankerst M, Breunig M, Kriegel H-P, et al. OPTICS: Ordering Points To Identify the Clustering Structure. In: Proc. ACM SIGMOD'99, Int Conf. on Management of Data, Philadelphia, PA, 1999

4 Breunig M M, Kriegel H-P, Ng R T, et al. LOF: identifying density-based local outliers. In: Proc. ACM SIGMOD 2000 Int Conf on Management of Data, Dalles, TX, 2000

5 Tang Jian, Chen Zhixiang, Ada Wai-chee Fu, et al. A Robust Outlier Detection Scheme for Large Data Sets. In: http://www.cs.panam.edu/～chen/papers.html

6 Zhou Yong-Feng, Liu Qing-Bao, Deng Su, et al. An Incremental Outlier Factor Based Clustering Algorithm. In: the First International Conference on Machine Learning and Cybernetics, Nov2002, CHINA

7 Jin Wen, Tung A K H, Han Jiawei. Mining Top-n Local Outliers in Large Databases. In: Proc. ACM KDD 2001, San Francisco, California USA

（上接第 156 页）

21 Sion R. Query Execution Assurance for Outsourced Databases. In: Proceedings of the 31st VLDB Conference, Trondheim, Norway, 2005

22 Khanna S, Zane F. Watermarking maps: hiding information in structured data. SODA, 2000. 596～605

23 Thruaisingham B. Recursion Theoretic Properties of the Inference Problem in Database Security [R]. In: MTP291. MITRE Corp, Bedford, Mass, 1990. 21～33

24 Itoh T. On Lower Bounds for the Communication Complexity of Private Information Retrieval. IEICE Transactions, 2001, E84- A (1)

25 Su T, Ozsoyoglu G. Controlling FD and MVD Inferences in Multilevel Relational Database System [J]. IEEE Transactions on Knowledge and Data Engineering, 1991, 3(4): 474～485

26 Ng W, Lau Ho-Lam. Effective Approaches for Watermarking XML Data. In: Proc. of DASFAA, LNCS, 2005, 3453: 68～80

27 WMDB System Architecture. http://www.cs.stonybrook.edu/～sion/projects/wmdb, 2004

28 王正飞，王曼，汪卫，等. 数据库中加密字符数据的存储与查询 [J]. 计算机研究与发展，2004, 41(Suppl. 10): 66～71

29 王晓峰，王尚平. 秘密同态技术在数据库安全中的应用 [J]. 计算机工程与应用，2003, 14: 194～196

30 Li Yingjiu, Guo Huiping, Jajodia S. Tamper Detection and Localization for Categorical Data Using Fragile Watermarks. In: Proceeding of the DRM'04, October 25, 2004, Washington, DC, USA

31 Li Yingjiu, Swarup V. Fingerprinting Relational Databases - Schemes and Specialties. IEEE Transaction on Dependable and Secure Computing, 2005, 2(1): 34～45

32 张敏，徐震，冯登国. 数据库安全 [M]. 北京：科学出版社，2005. 163～169

33 朱虹，史凌云，张勇. 多级安全数据库系统推理问题研究 [J]. 计算机工程与应用，2004, 13: 179～181

（上接第 165 页）

人名的组字相对来说比较稳定，地名和机构名也有比较好的后缀特征可以利用。如果把概念词抽取中把专名识别独立地作一个模块，利用其中独有的一些规律，对专名进行推测和验证，系统的准确率会更高。

（2）概念词内部语义关系的获取：分析复合概念词与构成它的基本概念词之间的语义关系，从而为领域 Ontology 的辅助构建、概念词辅助定义、概念词的辅助翻译提供支持。

（3）概念词关系的获取：概念之间存在着上下位关系、部分关系、同指关系、同义关系、反义关系等关系；利用概念词内部的语义信息和语料中的分布特点，可以进一步获取概念词间的关系。

参 考 文 献

1 Bourigault D. Surface Gramatical Analysis for the Extraction of Terminological Noun Phrases. In: Proceedings of COLING 92. 977～981

2 Frantzi T K. Incorporating Context Information for the Extraction of Terms. In: Preceedings of ACLEACL' 97

3 Wu shih-hung, Hsu wen -Lian. A semi-automatic domain ontology acquisition tool from Chinese Corpus [C]. In: Proc. of the 19th International Conference on Computational Linguistics (COLING)2002, Taipei, Taiwan, 2002. 1313～1317

4 Dunning T. Accurate Methods for the Statistical of Surprise and Coincidence. Association for Computational Linguistics, 1993, 19 (1): 61～76

5 Pantel P, Lin Dekang. A Statistical Corpus-based Term Extractor. In: Canadian Conference on AI, 2001. 36～46

6 Enguehard C, Pantera L. Automatic Natural Acquistion of a Terminology. Journal of Quantitative Linguistics, 1994, 2(1): 27～32

7 Luo S F, Sun M S. Two-Character Chinese Word Extraction Based on Hybrid of Internal and Contextual Measures. In: Proceeding of ACL2003, Sapporo, Japan, 2003

8 Riloff E. Automatically constructing a dictionary for information extraction tasks. In: Proceedings of the Eleventh National Conference on Artificial Intelligence, 1993. 811～816

9 郑家恒，杜永萍，宋礼鹏. 农业病虫害词汇获取方法初探. 见：孙茂松，陈群秀. 语言计算与基于内容的文本处理. 北京：清华大学出版社，2003. 61～66

10 张春霞. 领域文本知识获取方法研究及其在考古领域中的应用：[博士论文]. 北京：中科院计算所，2005

11 罗贝，吴洁，曹存根，等. 从文本中获取植物知识方法的研究. 计算机科学，2005, 32(10): 6～13

12 刘磊，曹存根. 一种基于"是一个"模式的下位概念获取方法. 计算机科学(已录用). 2006

13 http://icl.pku.edu.cn/icl%5Fres/segtag98/