基于相对密度和熵的混合属性聚类融合算法①

余 泽

(浙江工业大学 计算机科学与技术学院, 杭州 310023)

摘 要: 混合属性聚类是近年来的研究热点, 对于混合属性数据的聚类算法要求处理好数值属性以及分类属性, 而现存许多算法没有很好得平衡两种属性, 以至于得不到令人满意的聚类结果. 针对混合属性, 在此提出一种基于交集的聚类融合算法, 算法单独用基于相对密度的算法处理数值属性, 基于信息熵的算法处理分类属性, 然后通过基于交集的融合算法融合两个聚类成员, 最终得到聚类结果. 算法在 UCI 数据集 Zoo 上进行验证, 与现存 k-prototypes 与 EM 算法进行了比较, 在聚类的正确率上都优于 k-prototypes 与 EM 算法, 还讨论了融合算法中交集元素比的取值对算法结果的影响.

关键词: 聚类融合; 混合属性; 信息熵; 相对密度

Clustering Ensemble Algorithm for Mixed Attributes Data Based on Relative Density and Entropy

YU Ze

(College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou 310023, China)

Abstract: Mixed attributes data clustering is a research hotspot in recent years. For mixed attributes data clustering algorithm, it requires handling numeric attributes and categorical attributes simultaneously. However many algorithms have not very good balance with numeric and categorical attributes, and the cluster results are not satisfied. For mixed attributes data set, a new clustering ensemble algorithm based on intersection is proposed. It processes the numeric attributes with a new relative density clustering algorithm, and processes the categorical attributes with a clustering algorithm based on information entropy. Then it fuses these two cluster members with a cluster fusion algorithm based on intersection. Finally, it gets the clustering results. It is validated by taking an experiment on UCI data set Zoo, and compared with the existing k-prototypes algorithm and EM algorithm. The experiment result shows that the new algorithm has higher flexibility and accuracy. The influence of the intersection element ratioand to the result is also discussed.

Key words: clustering ensemble; mixed attributes; entropy; relative density

俗话说, 物以类聚, 人以群分, 聚类是数据挖掘的重要功能, 它能将物理或抽象对象的集合分为由类似对象组成的多个类或簇(Cluster)的过程[1]. 聚类以后, 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. 与分类不同, 聚类属于无监督无指导的学习方法, 完全是数据驱动的[2]. 聚类分析在各个广泛领域扮演着重要角色, 包括生物学、统计学、模式识别、信息检索、机器学习等等.

传统的聚类算法如 k-means 等只能处理纯数值属性数据, 但是现实世界中数据类型多种多样, 比如天气信息, 就包含温度, 湿度这样的数值属性数据, 也包括“阴”“晴”“雨”, “有风”“无风”这样的分类属性数据. 所以像 k-means 这类传统算法显然不能处理同时包含数值属性和分类属性的复杂数据. Huang 在[6]提出的 k-modes 算法处理分类属性数据, 进而提出结合 k-means 和 k-modes 算法的 k-prototypes 算法使之可以聚类分类属性和混合属性的数据集. 但是这些算法存在结果随机性大、不稳定、准确度不高、产生空簇等

① 收稿时间:2014-03-27;收到修改稿时间:2014-05-04

缺点.

聚类融合(cluster ensemble)也是进 10 年来一个研究比较多的课题. 聚类融合有两个基本步骤: 1.用不同的聚类算法对原始数据进行聚类, 得到不同的聚类划分; 2.采用融合算法将得到的聚类划分进行组合计算, 最终获得有效聚类结果. 文献[3]给出了聚类融合的基本算法框架图, 见图 1. X 为原始数据, $\Phi^{(i)}$ (i=1,2,3....r)为分别处理原始数据的聚类算法, $\lambda^{(i)}$ (i=1,2,3....r)为对应的各个不同的聚类划分, Γ为融合算法, λ 即为最终的聚类结果.

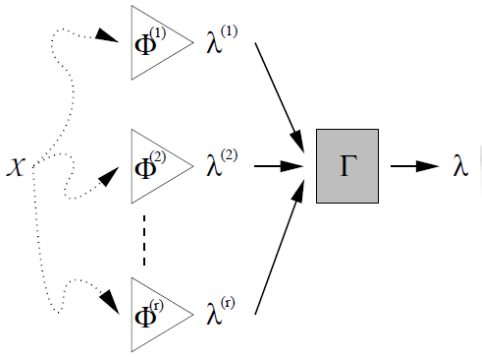

图 1　传统聚类融合框架图

由于聚类融合算法使用不同算法进行聚类, 再将各个聚类结果进行合并, 所以它比使用单一算法得到的结果更为稳定, 平均性能更加出色, 对噪点、孤立点等不敏感, 并且能对数据集子空间进行并行聚类[4].

本文首先介绍混合属性数据聚类问题产生的背景以及现存的问题; 接着介绍了混合属性聚类融合算法的相关研究以及发展, 提出改进的算法框架; 然后再新的算法框架下, 提出一种新的混合属性聚类融合算法; 之后详细介绍本文提出的具体融合算法和实现过程; 最后进行仿真实验验证.

1　相关研究

针对引言中提到的 k-prototypes 的缺点, 清华大学赵宇、李兵等提出 CEMC(cluster ensemble-based mixed attribute cluster)将聚类融合方法推广到了混合属性数据聚类问题的求解中. 但是 CEMC 虽然解决了 k-prototypes 算法的随机性, 稳定性等问题, 但其对数值属性直接采用 k-means 算法准确性不高, 所以存在一定不足. 通过研究发现, 对于具有 d 维属性的数据集, 若对每个维度都采用独立的聚类算法进行聚类, 那么将产生 d 个聚类划分成员, 若 d 过大, 则会大大降低融合算法的效率, 因此本文将图 1 的框架进行改进如图 2 所示.

本文在图 2 框架的基础上, 优化了聚类融合算法的第一步, 即对数值属性采用改进的相对密度算法, 而对分类属性采用基于熵的聚类算法, 提出了新的一种混合属性聚类融合算法, 推广了原来的混合属性聚类融合, 建立了算法框架, 提出有效的目标函数的和具体算法, 最后仿真验证算法的有效性.

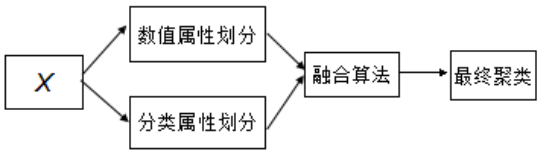

图 2　改进的聚类融合框架图

2　各相关定义及基础算法设计

2.1 FastRDBClustering 算法

针对 DBSCAN[5]算法对半径 Eps 和领域内数据点阈值 MinPts 这两个参数的极度依赖, 刘青宝等在文献[6]中提出相对密度算法 RDBClustering 解决了对全局参数的依赖. RDBClustering 算法不需要像 DBSCAN 那样设置 Eps 和 MinPts. 取而代之引入 k 近邻概念计算每个核心数据对象的密度, 以及相对密度差阈值 δ, 以此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能区分不同密度等级的. RDBClustering 以相对密度作为聚类计算准则, 将相对密度低于阈值 δ 的所有对象作为核心对象,并将相对密度可达的核心对象及其近邻对象构成的集合作为一个簇.

对于给定包含 d 维数值属性的数据集 $D=(x_1,x_2,\ldots,x_i,\ldots,x_n)$, $i=1,2,\ldots,n$, $p\in D$, k 为正整数, 相对密度差阈值 $\delta>0$.

RDBClustering 算法描述:

输入: D, 近邻个数 k, 相对密度阈值 δ

输出: 聚类结果

1) BEGIN

2) REPEAT

从 D 中未标识的对象中找核心对象 p;

3)以 p 为起始点生成核心对象集 *CoreSet*, 与其全部近邻共同形成一个簇;

4)UNTIL D 中所有核心对象都被处理;

5)根据密度可达将数据集中的所有未处理过的数据分配到其所在类;

6)END.

RDBClustering 的算法时间复杂度为与 DBSCAN 同阶, 为 $O(n\log n)$(n 为数据集中包含的对象数目), 与 DBSCAN 无明显差异, 聚类的大部分时间是用在区域查询操作上的, 所以如果能减少算法的区域查询执行次数, 就可以提高聚类速度[7].

而无论是 DBSCAN 还是 RDBClustring 要对核心对象领域内的所有对象进行查询操作来首先扩展核心集类簇, 之后再将边界点归入它所属的簇. 若 q 是核心对象 p 邻域中的一个核心对象,如果它的邻域被 p 邻域中的另一核心对象 r 的邻域所覆盖, 则对 q 的的领域的所有数据对象的查询都是不必要的. 因为 q 的邻域中所包含的数据对象可以通过 r 的领域内的对象查询得到, 所以 q 就没有必要作为核心对象用于类扩展.

因此我们采用文献[7]中采用的策略来对 RDBClustering 进行扩展, 提出 FastRDBClustring 不对核心对象领域内的所有数据对象查询, 而是只选取领域内的核心代表点来用户类扩展, 具体选取核心代表点的个数为属性个数的两倍, 对于 n 维数据, 则对于每个选中的核心对象选取其领域内 $2n$ 个核心代表对象进行扩展. 这样大大降低了数值属性部分的聚类时间复杂度. FRDBClustering 对 RDBClustering 算法第(3)步进行改进, 首先选出一个与核心对象最远的对象作为第 1 个代表对象; 随后则选出离所有已被选出的代表对象最远的对象作为下一个代表对象,直到选出所需的全部代表对象为止, 形成核心代表集, 然后再进行类扩展.

FastRDBClustering 算法描述

输入: D, 近邻个数 k, 相对密度阈值 δ

输出: 聚类结果

1)BEGIN

2)REPEAT

3)从 D 中未标识对象中找一个核心对象 p;

4)以 p 为起始点生成核心代表对象集 *CoreRepSet*, 与其全部近邻共同形成一个簇;

//由于数据是 m 维, 所以核心对象 p 有 $2m$ 个代表对象

5)UNTIL D 中所有核心代表对象都被处理;

6)根据密度可达将数据集中的所有未处理过的数据分配到其所在类;

7)END.

2.2 基于 Distance-熵的分类属性 ECCD 聚类算法

对于分类属性不能用聚类来进行数据对象相似性的计算, 而信息论中的熵可以用来度量信息量的大小, 所以用熵来处理分类属性比较合适, 在文献[9]中, 王述云等将 EFC 算发直接将混合属性与数值属性统一采用基于熵的方法聚类, 本文将文献[10]中对于数值属性的 EFC 改进为 ECCD(Entropy-based Clustering for Categorical Data)算法用于分类属性聚类.

对于给定包含 d 维数值属性的数据对象 $X=(x_1,x_2,\ldots,x_i,\ldots,x_d)$, $i=1,2,\ldots,d$, x_i 的取值 $v\in V_i$, $p=(x_i=v)$表示 $x_i=v$ 的概率, 则 X 的熵定义为[8]:

$$E(X)=-\sum_{i=1}^{d}\sum_{v\in V_i}p(x_i=v)\log_2(p(x_i=v)) \quad (1)$$

对于的数据集 $D=(x_1,x_2,\ldots,x_i,\ldots,x_n)$,

$$E(D)=-\sum_{i=1}^{d}\sum_{v\in V_i}p(x_i=v|D)\log_2(p(x_i=v|D)) \quad (2)$$

若数据集 $C=(C_1,C_2,\ldots,C_i,\ldots,C_k)$为 D 的一个划分, C_i 是其中的一个类, 则

$$E(C)=\sum_{i=1}^{k}\left(\frac{n_i}{n_D}(E(C_i))\right) \quad (3)$$

公式(3)中 $E(C)$称为划分 C 的期望熵, 其中 n_i 为 C_i 的数据对象个数, n_d 为 D 中数据对象总数. 根据熵的极大值原理,当 $E(C)$的值为最小时的划分为最佳, 即得到最佳聚类.

开始聚类是首先需要选择初始聚类点, 文献[10]中定义了一种用与数值属性的 Distance-熵, 因为熵的本质是客观表示系统的信息量, 所以本文直接将其运用于分类属性的初始点选择. Distance-熵的值域为[0,1],两点之间距离越近或者越远, 则熵值趋近于 0; 两点之间的距离越接近整个数据集的平均距离, 则熵值越趋近于 1. 而聚类的本质是数据对象与类内点距离越近越好, 与其他类的距离越远越好, 所以数据点的 Distance-熵越小, 则其越可能作为聚类中心点, 也适合于作为初始点.

设 $D=(x_1,x_2,\ldots,x_i,\ldots,x_n)$为包含 d 维数值属性的数据集, $i=1,2,\ldots,n$. Distance-熵计算公式如下:

$$E_i = -\sum_{j=1}^{n}\left(S_{ij}\log_2 S_{ij} + (1-S_{ij})\log_2(1-S_{ij})\right)(i \neq j) \quad (4)$$

S_{ij} 为 x_i 与 x_j 之间的相似度计算公式如下[10]:

$$S_{ij} = e^{-\alpha D_{ij}} \quad (5)$$

$$\delta(x_{it}, x_{jt}) = \begin{cases} 0 & (x_{it} = x_{jt}) \\ 1 & (x_{it} \neq x_{jt}) \end{cases} \quad (6)$$

D_{ij} 为 x_i 与 x_j 之间的距离, 对于分类属性数据:

$$D_{ij} = \sum_{t=1}^{d}\delta(x_{it}, x_{jt}) \quad (7)$$

经过文献[10]论证, 公式(5)中 α 取 ln 0.5/D_{avg} 适宜, D_{avg} 为数据集中所有点的平均距离.

聚类思想过程为, 首先选取 Distance-熵最小的数剧对象作为聚类中心, 设置相似度阈值 β (实验设定), 将所有与聚类中心相似度大于 β 的数据加入到此聚类中, 且将处理过的数据标记. 然后对未标记的数据重复此过程, 直到所有数据都被处理过.

设 D=(x_1,x_2,…,x_i,…,x_n)为包含 d 维数值属性的数据集, i=1,2,…,n, $p \in D$.

ECCD 算法描述

输入: D

输出: 聚类结果

1)BEGIN

2)计算 D 中每个数据对象的 Distance-熵;

3)从 D 中未标识的数据对象中选 Distance-熵最小的值作为聚类中心 p, 标记聚类 ID;

4)以 p 为聚类中心, 将 D 中未标识的数据对象中所有与 p 相似度大于 β 的数据加入到此聚类中, 并标记所有处理过的数据对象;

5)跳转 3), 直到所有点都处理;

6)END.

3 ICEMD混合数据聚类融合算法

本文在文献[11]的融合算法基础上进行改进,提出在基于交集的数据融合最终算法 ICEMD(Intersection-based Clustering Ensemble for Mixed Data), 假设对于混合属性数据集 D, 已经采用上述 Fast RDB Clustering 算法和 ECCD 算法, 分别对数值属性与分类属性进行聚类, 得到数值属性聚类划分 C_N 和分类属性聚类结果 C_C, 则 ICEMD 算法即需要将 C_N 和 C_C 进行融合, 最后产生最终聚类.

设对于某混合属性数据集, 已经得到其数值属性聚类成员划分:

$$C_N = \{C_{n1}, C_{n2}, \cdots C_{ni} \cdots C_{nk}\},\ i = 1,2,\cdots k,$$

分类属性聚类成员划分:

$$C_C = \{C_{c1}, C_{c2}, \cdots C_{cj} \cdots C_{cl}\},\ i = 1,2,\cdots l.$$

从 C_N 和 C_C 中任取一簇,他们之间的交集 $\omega_t = C_{ni} \cap C_{cj}$; 他们之间的并集 $\pi_t = C_{ni} \cup C_{cj}$;

设 $\pi(i,j) = \pi_i \cap \pi_j$;

设 $\theta = \dfrac{|\pi(i,j)|}{\max\{|\pi_i|, |\pi_j|\}}$ 作为合并簇的阈值.

ICEMD 算法描述

输入: 数值属性聚类划分 C_N,分类属性聚类结果 C_C, 空集 Π, 合并阈值 ε.

输出: 聚类结果

1)BEGIN

2)从 C_N 和 C_C 中分别任取一个簇 C_n 和 C_c, 计算交集为 ω_i, 并集 π_i;

3)遍历所有交集 ω_t 含有大于两个以上的元素的组合, 将这两个簇的并集 π_i 加入 Π;

4)计算 Π 中两两集合之间的交集元素比 θ, 并从大到小排列.

5)若 $\theta > \varepsilon$ (ε 需要在具体实验中调整),

则合并 π_i 和 π_j, 并且更新 Π, 跳转(4);

否则

若存在 $\pi(i,j) \neq \varnothing$ 那么

若 $|\pi_i| > |\pi_j|$ 则 $\pi_i = \pi_i - \pi(i,j)$;

否则 $\pi_j = \pi_j - \pi(i,j)$;

更新 Π, 跳转(4);

否则, 若存在数据不属于 Π 中的任何一个集合, 为数据独立新建一个集合.

6)END.

举例如表 1 为人工生成的测试数据集, 含有 1 维数值属性和 2 维分类属性.

对表 1 数据集的 1 维数值属性采用 FastRDBClustring 算法得到聚类成员 C_N, 对剩下的分类属性采用 ECCD 算法得到聚类成员 C_C:

C_N={{1,2,3,15},{5,7,9,20},{4,6,8,10},
{11,12,16,19},{10,14},{13,17,18}};

C_C={{7,9,13,14,17},{13,18},{5,20},
{3,6,12,16,19},{1,2,4,8,10,15},{11}}.

最后采用 ICEMD 算法将 C_N 与 C_C 融合，最终得到聚类结果为{{1,2,4,8,10,13,15},{5,7,9, 13,14,17,18,20},{3,6,11,12,16,19}}，与数据集的实际情况完全符合.

表 1 测试数据集

数据序号	温度	天气	颜色	数据序号	温度	天气	颜色
1	35	晴	蓝	11	12	雨	红
2	30	晴	蓝	12	6	雨	红
3	8	雨	红	13	22	阴	绿
4	28	晴	蓝	14	21	阴	黄
5	18	阴	绿	15	36	晴	蓝
6	7	雨	红	16	4	雨	红
7	21	阴	绿	17	20	阴	黄
8	29	晴	蓝	18	22	阴	绿
9	20	阴	绿	19	5	雨	黄
10	35	晴	蓝	20	19	阴	绿

3 算法性能分析以及仿真实验

3.1 性能分析

设实验混合属性数据集有 n 个数据对象，d 维分类属性，对于数值属性采用 FastRDBclustering 算法，时间复杂度与 DBSCAN 同阶，为 $O(n\log n)$，但是改进了核心集的计算；对于分类属性采用了 ECCA 算法，计算初始化 Distance-熵的复杂度为 $O(n^2d)$，接着从未标记的数据对象中重新寻找聚类中心的复杂度为 $O(\log d)$；最后 ICEA 算法遍历两个聚类成员，设两个聚类成员的类个数分别为 k_1 和 k_2，则 ICEA 的时间复杂度为 $O(k_1k_2)$. 若三个算法顺序执行则总时间复杂度应该是 $O(n\log n)+O(n^2d)+O(k_1k_2)$，不过因为前两个算法处理不同的数据，没有先后顺序之分，因此可采用分布式计算，大大降低时间消耗.

3.2 实验结果分析

实验使用操作系统为 Windows 7 下的 weka 工具上来进行仿真验证. 采用 k-prototypes 算法以及 EM 算法实验一选取了 UCI 上的 Zoo 数据集，它包含 101 条动物的形态和习性信息,每种动物由 1 个数值型属性和 16 个类别型属性来描述,将其中 8 个冗余的分类属性删除，数据集的最后一个类别属性给出了这些动物的 7 个分类信息.

图 3 为 Zoo 的正确分类显示图，图 4 和图 5 分别为 EM 算法和本文算法的聚类结果，图 4 阴影多边形中的数据对象分类完全错误，而本文聚类结果基本正确. 聚类结果的误差率分析如表 2 所示.

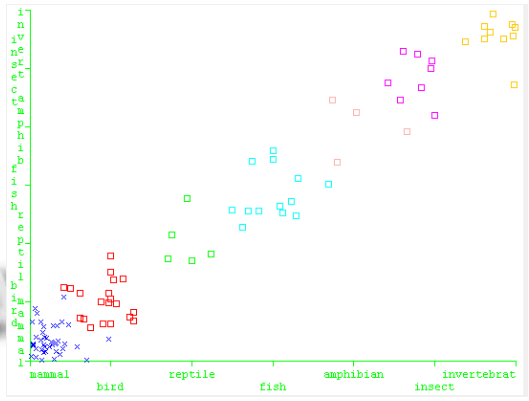

图 3 Zoo 的正确分类

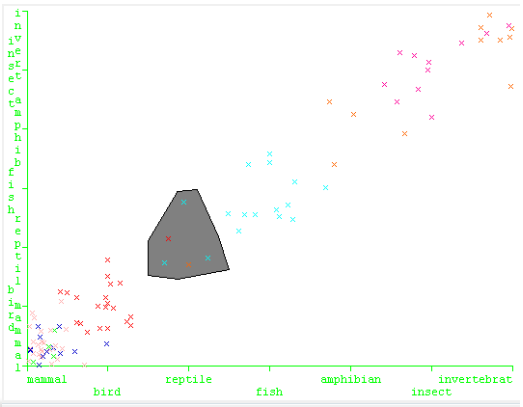

图 4 EM 算法聚类结果

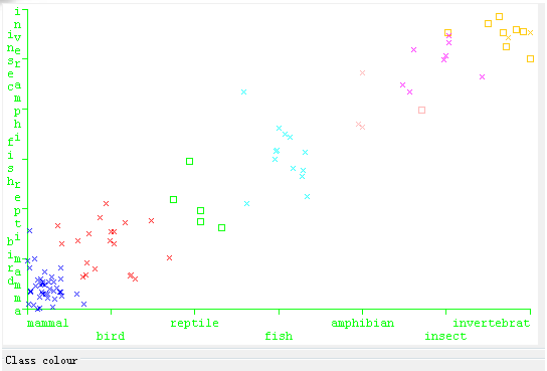

图 5 本文算法聚类结果

采用正确率(Accuracy)和聚类纯度(Purity)来评估算法的聚类质量:

$$Accuracy = \frac{\sum_{i=1}^{k} a_i}{n} \tag{8}$$

$$Purity = \frac{\sum_{i=1}^{k} \frac{a_i}{a_i + b_i}}{k} \tag{9}$$

其中, a_i表示应该分到i类, 且正确分到i类的数据点数目, b_i表示不应该分到 i 类但是被分到了 i 类的数据点数目, k 为聚类数目[12].

表 2　聚类结果的误差率分析

算法	正确分类个数	正确率	聚类纯度
k-prototypes	80	79.2%	79.4%
EM	73	72.3%	70.2%
本文算法	95	86.2%	93.4%

可以看出本文算法的正确率和聚类纯度相比 k-prototypes 和 EM 算法都要高.

本文融合算法中涉及到的合并交集元素比 ε 的取值不同, 会对实验结果产生很大的影响, 因此实验中分别取了不同的 λ 值, 由实验结果得出对于 Zoo 数据集 ε 最佳取值, 如图 6 所示, 横坐标为 ε 的不同取值, 纵坐标为, 分类的正确率. 当 ε=0.65 时, 聚类正确率最高.

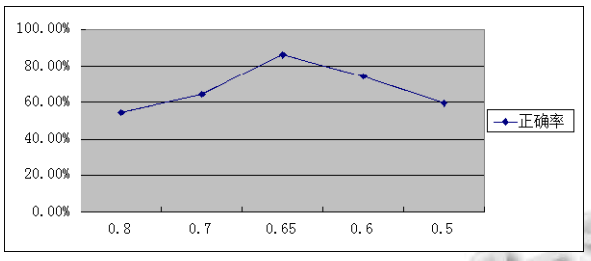

图 6　ε 不同取值与正确率的关系

4　总结与展望

混合属性数据的聚类问题与聚类融合算法的研究都是近年来的热门课题. 对混合属性数据的处理需要兼顾数值属性又要考虑分类属性, 在聚类过程中要同时考虑两种数据对聚类结果的影响. 本文提出了一种基于交集运算的聚类融合算法来处理对混合属性数据的聚类, 算法分为两步, 首先改进了 RDBClustering 算法, 提高了数值属性聚类的时间效率, 同时引入了信息熵来处理分类属性聚类提出 ECCD 算法, 较好的解决了分类属性的相似性度量问题, 最后 ICEMD 算法融合算法, 采用交并集运算特性将数值属性聚类划分与分类属性聚类划分进行融合, 最终得到聚类结果.

需要注意的是文本 ICEMD 算法中需要对交集元素比 ε 进行讨论并且, ε 的取值不同会对使最终的聚类结果不同, 所以下一步对 ε 的取值原则, 需要进一步研究.

本文算法针对的是静态的混合属性数据集的聚类问题, 下一步可以针对混合属性数据流的聚类融合算法做进一步研究.

参考文献

1 毛国君,段立娟,王石,石云.数据挖掘原理与算法.北京:清华大学出版社,2007.

2 Huang ZX. Extensions to the K-means algorithm for clustering large data sets with categorical values. Data Mining and Knowledge Discovery, 1998, 2(3): 283–304.

3 Strehl A, Ghosh J. Cluster ensembles: A knowledge reuse framework for combining multiple partitions. Journal of Machine Leaming Research, 2003, 3(3): 583–617.

4 Topchy A, Jain AK, Punch W. A mixture model for clustering ensembles. Proc. of the 4th SIAM International Conference on Data Mining. USA: Society of Industrial and Applied Mathematics Press. 2004. 379–390.

5 Ester M, Kriegel H, Sander J, et a1. A density-based algorithm for discovering clusters in large spatial databases with noise. Proc. 2nd Int Conf on Knowledge Discovery and Data Mining. Portland, OR. 1996. 226–231.

6 刘青宝,邓苏,张维明.基于相对密度的聚类算法.计算机科学,2007,34(2):192–195.

7 周水庚,周傲英,曹晶,胡运发.一种基于密度的快速聚类算法.计算机研究与发展,2000,37(11):1287–1292.

8 Gokcay E, Principe JC. Information theoretic clustering. IEEE Trans. on Pattern Analysis and Intelligence, 2002, 24(2): 158–171.

9 王述云,胡运发,范颖捷,等.基于距离与熵的混合属性数据流聚类算法.小型微型计算机系统,2010,31(12):2365–2372.

10 Yao J, Dash M, Tan ST, et al. Entropy-based fuzzy clustering and fuzzy modeling. Fuzzy Sets and Systems, 2000, 13: 381–388.

11 李桃迎,陈燕,张金松,等.基于聚类融合的混合属性数据增量聚类算法.控制与决策,2010,27(4):603–609.

12 赵恒,杨万海.模糊 K-Modes 聚类精确度分析.计算机工程,2003,29(12):27–28.