文章编号: 1001-0920 (2011) 10-1542-03

一种有效的分类型数据聚类方法

罗 可, 洪亮亮, 童小娇

(长沙理工大学 计算机与通信工程学院，长沙 410114)

摘 要: 鉴于传统的 K-means 聚类算法只限于处理数值型数据, 将 K-means 算法扩展到分类型数据域, 提出一种分类型数据聚类方法. 根据与每个分类属性的每个值相关的数据分布信息, 同时结合数据的纵向与横向分布来评价数据对象与类之间的差异性, 定义了一种新的距离度量. 该方法能发现同一属性不同值间的内在关系, 并能有效地度量对象间的差异性. 用 UCI 中的数据集对所提算法进行验证, 实验结果表明了该算法具有较好的聚类效果.

关键词: 聚类分析；分类型数据；差异性；域值；共生

中图分类号: TP18 **文献标识码:** A

Efficient categorical data clustering method

LUO Ke, *HONG Liang-liang*, *TONG Xiao-jiao*

(Institute of Computer and Communication Engineering，Changsha University of Sciences and Technology，Changsha 410114，China．Correspondent：HONG Liang-liang，E-mail：hongliangliang2008@163.com)

Abstract: The traditional K-means clustering algorithm is only for numerical data. Therefore, a categorical data clustering method is proposed through extending the K-means algorithm to categorical data domain. In accordance with the information of data distribution correlated to each value of each categorical attribute, and at the same time combined with the vertical and horizontal distribution of the data to measure the difference between data object and the class, a new distance metric is defined. This method can find the intrinsic relationship between the different values of the same attribute, and can measure the difference between objects effectively. The performance of the proposed algorithm is verified on the dataset of the UCI. Experimental results show that the algorithm has better clustering results.

Key words: cluster analysis；categorical data；dissimilarity；domain value；co-occurrence

1 引 言

聚类是数据挖掘的基本操作之一. 聚类过程就是将相似的对象划为一组, 使得类内相似性最大而类间相似性最小. 聚类分析现已成为数据挖掘研究领域中一个非常活跃的研究课题[1]. 现实世界中的数据类型多种多样, 大部分的聚类研究主要面向数值属性, 数值型数据间的相似性容易通过数据的几何位置进行定义[2], 而分类型数据由于是多值属性, 传统的距离定义不再适用. 近年来也出现了一些针对分类型数据的聚类方法, 例如, 文献 [3] 采用简单匹配度量对象间差异; [4] 采用相对频率度量; [5] 采用启发式的优化算法记录对象间链接数定义的准则函数; [6] 在聚类中使用基于熵的度量; [7] 按照次序读取每个记录, 根据该记录与已有类之间的相似性来决定是否归入已有类或新建一个类; [8] 基于属性域的基数提出一种新的距离度量等.

上述文献中有些在确定类中心时以单个点作为中心点, 这便导致了算法的不稳定, 降低了聚类准确性: 有些在评价属性值之间的距离时只考虑该属性值的频度或者采用二分法 (简单匹配和 Jaccard 系数), 而没有考虑数据分布. 此外, 对于多值属性还应考虑属性值之间的尺度. 因此, 本文根据与每个分类型属性的各个值相关的数据分布信息, 提出一种新的方法, 用于度量同一属性的不同值间的距离. 同时以 K-means 算法为基础, 结合数据的纵向 (类中该属性值的频率) 与横向 (该属性值与其他属性的各个不同值共同出现的次数) 分布来度量数据对象与类之间的差异性, 并进行了仿真验证.

收稿日期: 2010-06-20；**修回日期:** 2010-09-02.

基金项目: 国家自然科学基金项目(10926189, 10871031)；湖南省自然科学衡阳联合基金项目(10JJ8008).

作者简介: 罗可(1961−), 男, 教授, 博士, 从事数据挖掘、计算机应用等研究；洪亮亮(1987−), 女, 硕士生, 从事数据库技术、数据挖掘的研究.

2 分类型数据聚类

2.1 相关定义

设属性集 $A=\{A_1,A_2,\cdots,A_m\}$, 属性值域集合为 $D=\{D_1,D_2,\cdots,D_m\}$, 则属性 A_i 的值域为 $D_i=\{v_{i,1},v_{i,2},\cdots,v_{i,p_i}\}$, $i=\{1,2,\cdots,m\}$. 其中: $v_{i,r}$ 为属性 A_i 的第 r 个值, 即属性 A_i 的一个域值, $r=\{1,2,\cdots,p_i\}$, p_i 为属性 A_i 的不同取值数; D_i 中取值的次序只是为了计算方便, 没有其他含义.

设数据集 $X=\{x_1,x_2,\cdots,x_n\}$, 其中第 i 个样本的 m 个属性值表示为 $x_i=[x_{i1},x_{i2},\cdots,x_{im}]^{\mathrm{T}}$, $x_{ij}\in D_j$, $j=\{1,2,\cdots,m\}$.

定义 1　设一对象 x, 若其在属性 A_i 取 $v_{i,r}$ 的情况下, 属性 A_j 取 $v_{j,\theta}$($v_{j,\theta}$ 表示属性 A_j 的第 θ 个取值), 则称域值 $v_{i,r}$ 和 $v_{j,\theta}$ 在对象 x 中共生, 此时将形如 $(v_{i,r},v_{j,\theta})$ 的形式称为值对.

定义 2　设值对 $(v_{i,r},v_{j,\theta})$ 在数据集中出现的次数为 $\mathrm{tm}_{v_{i,r}}^{v_{j,\theta}}$, $\theta\in\{1,2,\cdots,nA_j\}$, 简记为 tm.

设 A_i 的两个取值为 $v_{i,rx},v_{i,ry}$, 且有 $v_{i,rx}\neq v_{i,ry}$, $rx,ry\in\{1,2,\cdots,p_i\}$, 统计出 $v_{i,rx}$ 和 $v_{i,ry}$ 与 D_j 中各元素构成的值对出现的次数. 设 $v_{i,rx}$ 和 $v_{i,ry}$ 与 D_j 中各元素均能构成值对, 且各 tm 均一致. 假设现在增加了 A_k, 分别统计出 $v_{i,rx}$ 和 $v_{i,ry}$ 与 D_j 中各元素构成的值对的 tm. 一方面, 若 $v_{i,rx}$ 和 $v_{i,ry}$ 均能与 $v_{k,\alpha}(\alpha=\{1,2,\cdots,p_k\})$ 构成值对且两者的 tm 相差较小, 则 $v_{i,rx}$ 和 $v_{i,ry}$ (在 A_i,A_j 和 A_k 条件下) 的差异性较小, 反之较大; 另一方面, 若 $v_{i,rx}$ 与 D_k 中的部分值构成值对, 则 $v_{i,ry}$ 与 D_k 中剩余的元素构成值对, 分析结果同上.

定义 3　假如属性 A_i 的第 r 个值为 $v_{i,r}$, 计算其与剩余 $m-1$ 个属性的各个值共同出现的次数, 以此构造一个行向量为

$$\begin{aligned}V_{v_{i,r}}^{A_j\in A,j\neq i}=&(\mathrm{tm}_{i,r}^{v_{1,1}},\mathrm{tm}_{i,r}^{v_{1,2}},\cdots,\mathrm{tm}_{i,r}^{v_{1,p1}},\\&\cdots,\mathrm{tm}_{i,r}^{v_{i-1,1}},\cdots,\mathrm{tm}_{i,r}^{v_{i-1,p(i-1)}},\\&\cdots,\mathrm{tm}_{i,r}^{v_{i+1,1}},\cdots,\mathrm{tm}_{i,r}^{v_{i+1,p(i+1)}},\\&\cdots,\mathrm{tm}_{i,r}^{v_{m,1}},\cdots,\mathrm{tm}_{i,r}^{v_{m,pm}}),\end{aligned}\tag{1}$$

将该行向量简记为 VIJ.

定义 4　将同一个属性的两个不同值之间的距离简记为值距离.

本文采用余弦相似性来度量两个行向量, 即属性 A_i 的第 p 个值与第 q 个值之间的值距离为

$$\sigma_{A_i}(v_{i,p},v_{i,q})=1-\frac{V_{v_{i,p}}^{A_j\in A,j\neq i}V_{v_{i,q}}^{A_j\in A,j\neq i}}{\|V_{v_{i,p}}^{A_j\in A,j\neq i}\|\|V_{v_{i,q}}^{A_j\in A,j\neq i}\|},\tag{2}$$

其中 $\|V_{v_{i,r}}^{A_j\in A,j\neq i}\|=\sqrt{\sum_{j=1,j\neq i}^{m}\sum_{\theta=1}^{pj}(\mathrm{tm}_{v_{i,r}}^{v_{j,\theta}})^2}$.

若对象 x 和 y 在属性 $A_i(i=1,2,\cdots,m)$ 下的取值分别为 $v_{i,p}$ 和 $v_{i,q}$, 则 x 和 y 的距离为

$$d(x,y)=\sum_{i=1}^{m}\sigma_{A_i}(v_{i,p},v_{i,q}).\tag{3}$$

定义 5　对于每个类, 以该类中所拥有的各个属性的各个属性值在该类中所占的比例来代表该类[9], 记第 i 类的类中心为

$$\begin{aligned}c_i=&\frac{1}{|C_i|}((n_{1,1,c},n_{1,2,c},\cdots,n_{1,p1,c}),(n_{2,1,c},n_{2,2,c},\\&\cdots,n_{2,p2,c}),\cdots,(n_{m,1,c},n_{m,2,c},\cdots,n_{m,pm,c})),\end{aligned}\tag{4}$$

其中 $n_{u,v,c}$ 表示在类 C_i 中第 u 个属性的第 v 个取值所占的个数. 那么对象 x 与类中心 c_i 的距离为

$$\begin{aligned}&d(x,c_i)=\\&\frac{1}{|C_i|}(n_{1,1,c}\sigma_{A_1}(v_{1,1},x_j)+n_{1,2,c}\sigma_{A_1}(v_{1,2},x_j)+\\&\cdots+n_{1,p1,c}\sigma_{A_1}(v_{1,p1},x_j)+\cdots+\\&n_{i-1,1,c}\sigma_{A_{i-1}}(v_{i-1,1},x_j)+\cdots+\\&n_{i-1,p(i-1),c}\sigma_{A_{i-1}}(v_{i-1,p(i-1)},x_j)+\cdots+\\&n_{i+1,1,c}\sigma_{A_{i+1}}(v_{i+1,1},x_j)+\cdots+\\&n_{i+1,p(i+1),c}\sigma_{A_{i+1}}(v_{i+1,p(i+1)},x_j)+\cdots+\\&(n_{m,1,c}\sigma_{A_m}(v_{1,1},x_j)+\cdots+\\&(n_{m,p,mc}\sigma_{A_m}(v_{m,pm},x_j)).\end{aligned}\tag{5}$$

2.2 算法描述

下面给出有效的分类型数据聚类 (ECDC) 算法的主要步骤:

输入: 数据集 $X=\{x_1,x_2,\cdots,x_m\}$, 聚类数 k;

输出: 已聚好的类 $C_1,C_2,\cdots,C_k$.

Step 1: 扫描数据集, 指定聚类数 k;

Step 2: 计算每个属性值对应的 VIJ;

Step 3: 根据式 (2) 计算属性值间的距离;

Step 4: 为对象随机分配 $1,2,\cdots,k$ 类标号;

Step 5: 按照定义 5 计算类中心;

Step 6: 按照式 (5) 重新计算每个对象到各个类中心距离, 将其分配到离其最近的类, 待分配完后重新计算类中心;

Step 7: 若此时有大于或等于一个类的对象成员为空, 则返回 Step 4; 否则, 重复 Step 6, 直到类成员不再变化或者达到指定的迭代次数为止, 算法结束.

3 算例分析

文献 [9] 已表明, 对于如表 1 所示的数据集, 将其聚为两个类最佳; 而从表 1 可以看到, 对于类 c_1, 属性 Genre 和 Actor 都只有 1 个值, 只有属性 Director 有 2 个值, 类 c_2 中每个属性都有 2 个值, 这便使得 c_2 的紧

凑性看上去不如 c_1. 这个例子说明, 属性值间的距离不能严格被认为是二值的, 因为在一个较优的类中应该是各值对之间更加相似, 而不是属性值与分离的类中占主导地位的属性值更加相似.

表 1 movie database的一个实例

	Director	Actor	Genre	c
t_1	Scorsese	De Niro	Crime	c_1
t_2	Coppola	De Niro	Crime	c_1
t_3	Hitchcock	Stewart	Thriller	c_2
t_4	Hitchcock	Grant	Thriller	c_2
t_5	Koster	Grant	Comedy	c_2
t_6	Koster	Stewart	Comedy	c_2

ECDC 算法的算例分析过程如下:

Step 1: 构造每个属性值的VIJ向量, 如构造Scorsese和Coppola的VIJ. 由定义3可得

$$V_{\text{Scorsese}}^{\text{Actor,Genre}} = (1,0,0,1,0,0),$$
$$V_{\text{Coppola}}^{\text{Actor,Genre}} = (1,0,0,1,0,0).$$

同理可得其他各个属性值的VIJ.

Step 2: 计算每个属性各个属性值两两之间的值距离, 如计算属性 A_i 的值Scorsese和Coppola之间的值距离. 由式(1)可得

$$\sigma_{\text{Director}}^{\text{Scorsese,Coppola}} = 1 - \frac{2}{\sqrt{2}\times\sqrt{2}} = 0.$$

同理可得该属性的其他各属性值两两之间的值距离以及其他属性各属性值两两之间的值距离.

Step 3: 计算各对象与各类中心的距离, 如计算对象 t_1 到类 c_1 的距离. 类 c_1 的中心表示为 ((Scorsese, Coppola), 2DeNiro, 2Crime)/2. 那么对象 t_1 到类 c_1 的距离为

$$d(t_1,c_1) =$$
$$(\sigma_{A_1}(\text{Scorsese},\text{Coppola}) + \sigma_{A_1}(\text{Scorsese},\text{Scorsese}) +$$
$$2\sigma_{A_2}(\text{DeNiro},\text{DeNiro}) + 2\sigma_{A_3}(\text{Crime},\text{Crime}))/2 = 0.$$

同理, 可得其他各对象与各中心点的距离.

4 实验结果及分析

4.1 算法仿真及分析

为了分析ECDC算法的性能, 分别在UCI[10]中的Soybean, Zoo, Breast cancer和Vote数据集上对算法进行测试, 各数据集描述见表2. 同时, 以文献[8]的 K-modes算法和文献[9]所示的算法作对比. 采用Huang提出的聚类精度来度量聚类的准确性, 其定义为

$$r = \frac{1}{n}\sum_{l=1}^{k} a_l, \tag{6}$$

其中 a_l 为同时出现在类 C_l 和其相应的标记类中的对象数. 对于有缺失值的数据集, 如Vote, 对每个有缺失值的属性采用该属性中出现频率最高的属性值来填补该属性的缺失值. 实验结果如表3所示, 其中Zoo数据集的聚类精度为多次运行算法得到的平均值. 由于处理字符型分类数据时算法需要时间较长, 只对小数据集进行了测试, 由实验结果可以看出, 本算法聚类效果更好.

表 2 数据集描述

数据集	属性数	样本数	分类数
Soybean	22	47	4
Zoo	17	101	7
Breast cancer	10	699	2
Vote	16	435	2

表 3 数据集的聚类精度

数据集	文献[8]	文献[9]	本文算法
Soybean	0.37	1	1
Zoo	0.42	0.81	0.85
Breast cancer	0.65	0.93	0.94
Vote	0.61	0.87	0.87

4.2 ECDC的时间复杂度

若样本数为 n, 属性数为 m, 聚类数为 K, 第 i 个属性有 $v_i(i\in 1,2,\cdots,m)$ 种取值, m 个属性的取值种数之和为 R, 即 $R = v_1+v_2+\cdots+v_m$. 记 $T = v_1^2+v_2^2+\cdots+v_m^2$, 那么计算 m 个属性的所有值距离的时间复杂度为 $O(TR+nR^2)$. 对于每一次迭代, 算法ECDC的时间度为 $O(nKms)$ (S 为 m 个属性的取值种数的平均值). 如果算法ECDC的迭代次数为 p, 则算法ECDC总的时间复杂度为 $(RT+nR^2+pnkms)$, 可以看出算法总的时间复杂度与对象数成线性关系.

5 结 论

本文提出了一种分类型数据聚类方法. 在度量对象与类中心的距离时, 不仅考虑了类中数据的分布, 同时考虑了每个属性的各个属性值分别与数据集中其余所有属性的各个属性值之间的联系. 该方法能有效地度量同一属性不同值之间以及对象与中心点的差异性. 实验结果表明了该算法有更好的聚类精度.

参考文献(References)

[1] HAN J W, KAMBER M. Data mining: Concepts and techniques[M]. New York: Morgan Kaufmann Publishers, 2001: 335-388.

[2] Parmar D, Wu T, Jennifer B. MMR: An algorithm for clustering categorical data using rough set theory[J]. Data and Knowledge Engineering, 2007, 63(3): 879-893.

[3] Huang Z X, Michael K N. A fuzzy k-modes algorithm for clustering categorical data[J]. IEEE Trans on Fuzzy Systems, 1999, 7(4): 446-452.

[4] San O M, Huynh V N, Nakamori Y. An alternative extension of the K-means algorithm for clustering categorical data[J]. Applied Mathmatic Computer Science, 2004, 14(2): 241-247.

(下转第1548页)