第33卷 第13期 计 算 机 工 程 2007年7月
Vol.33 No.13 Computer Engineering July 2007
·人工智能及识别技术· 文章编号：1000—3428(2007)13—0200—02 文献标识码：A 中图分类号：TP183

改进的k-平均聚类算法研究

孙士保[1,2]，秦克云[1]
(1. 西南交通大学智能控制开发中心，成都 610031；2. 河南科技大学电子信息工程学院，洛阳 471003)

摘 要：聚类算法的好坏直接影响聚类的效果。该文讨论了经典的k-平均聚类算法，说明了它存在不能很好地处理符号数据和对噪声与孤立点数据敏感等不足，提出了一种基于加权改进的k-平均聚类算法，克服了k-平均聚类算法的缺点，并从理论上分析了该算法的复杂度。实验证明，用该方法实现的数据聚类与传统的基于平均值的方法相比较，能有效提高数据聚类效果。

关键词：聚类算法；k-平均；权；聚类数据挖掘

Research on Modified k-means Data Cluster Algorithm

SUN Shibao[1,2], QIN Keyun[1]
(1. Intelligent Control Development Center, Southwest Jiaotong University, Chengdu 610031;
2. Electronic Information Engineering College, Henan University of Science and Technology, Luoyang 471003)

【Abstract】 The method of data clustering will influence the effect of clustering directly. The algorithm of k-means is discussed, the shortages of this algorithm such as it can not deal with symbolic data and it is sensitive for data of isolation point and noise are demonstrated. A modified k-means clustering algorithm based on weights is put forward, it changes the shortcomings of k-means. Its complexity is analyzed from theoretical. The experiments show that, compared with traditional method based on means, the modified data clustering algorithm can improve the efficiency of data clustering.

【Key words】 cluster algorithm; k-means; weights; cluster data mining

聚类是将物理或抽象对象的集合分组成为由类似的对象组成的多个类的过程。它的目的是使得属于同一类别的个体之间的相似度尽可能大，而不同类别的个体之间的相似度尽可能小。在机器学习领域，聚类是无指导学习的一个例子。聚类分析是知识发现的重要方法，在图像识别、信息检索、数据挖掘、统计学、机器学习、空间数据库、生物学以及市场营销等领域有着广泛的应用[1~4]。

目前常用的聚类算法包括：以k-平均算法(k-Means)[5]和k-中心点算法(k-Medoid)[6]为代表的划分法；以AGNES[6]和DIANA[6]为代表的层次聚类算法；以DBSCAN[7]和OPTICS[8]为代表的基于密度的方法；以STING[9]为代表的基于网格的方法；以COBWEB[2]和SOM[10]为代表的基于模型的方法；以顺序地比较一个集合中的对象[11]和OLAP数据立方体[12]为代表的基于孤立点的分析方法。这些方法中的大部分聚类算法都是面向数值属性，而针对符号属性的比较少[1,2]。现有许多改进的算法，如基于数据场改进的PAM聚类算法[13]，基于Rough集的层次聚类算法[14]。对于文献[13]中基于数据场改进的PAM聚类算法是一种较好的划分聚类算法，但在处理数据场势函数时的计算开销很大，很难应用到大型数据集中去；另外，它对数值属性效果较好，对符号属性基本上不能实现。本文给出一种基于加权改进的k-平均聚类算法，这种新的划分聚类算法，不仅可以处理数值属性，而且可以处理符号属性，另外，当被挖掘的数据中存在孤立点数据和“噪声”时，这种算法的处理效果也非常好。

1 k-平均算法的分析

k-平均(k-Means)算法是一种基于划分方法的聚类算法，它是最早提出的较为经典的聚类算法之一。

1.1 k-平均算法

k-平均算法的主要思想是试图对 n 个对象给出 k 个划分($k \le n$)，其中每个划分代表一个簇。首先，随机地选择 k 个对象，每个对象初始地代表一个簇的平均值或中心。对剩余的每个对象，根据其与各个簇中心的距离，将它赋给最近的簇。然后重新计算每个簇的平均值，对数据库中的每个对象与每个簇的平均值相比较，把对象赋给最相似的某个簇。这个过程不断重复，直到簇中的对象都是“相似的”，而不同簇中的对象都是“相异的”，即准则函数收敛使平方误差函数值最小。

1.2 k-平均算法的优缺点

用k-平均算法来聚类时，当结果簇是密集的，而簇与簇之间区别明显时，它的效果较好。对处理大数据集，该算法是相对可伸缩的和高效率的，因为它的复杂度是 $O(nkt)$，其中，n 是所有对象的数目，k 是簇的数目，t 是迭代的次数。通常 $k \ll n$ 且 $t \ll n$。这个算法经常以局部最优结束。但是，k-平均方法只有在簇的平均值被定义的情况下才能使用。这对于处理符号属性的数据不适用，它还要求用户必须事先给出 k(要生成的簇的数目)值。另外，对于“噪声”和孤立点数据是敏感的，少量的该类数据能够对平均值产生极大的影响。

1.3 k-平均算法的过程

算法：k-平均。划分的 k-平均算法基于簇中对象的平均值。

输入：簇的数目 k 和包含 n 个对象的数据库。

基金项目：国家自然科学基金资助项目(60474022)
作者简介：孙士保(1970－)，男，讲师、博士研究生，主研方向：智能信息处理；秦克云，博士、教授、博士生导师
收稿日期：2006-07-10 **E-mail：**sunshibao@126.com

输出：k 个簇，使平方误差准则最小。

方法：

(1)任意选择 k 个对象作为初始的簇中心；

(2)repeat；

(3)根据簇中对象的平均值，将每个对象(重新)赋给最类似的簇；

(4)更新簇的平均值，即计算每个簇中对象的平均值；

(5)until 不再发生变化。

2 基于加权改进的 k-平均算法

2.1 权的产生

在含有n个数据对象的数据库中，每个数据对象对于知识发现来说作用是不同的，为了区分这些相异之处，给每个数据对象赋予一个定量的值 w_j 即权。$w_j = \frac{w'_j}{\sum_{j=1}^{n} w'_j}$，其中 $w'_j = \frac{1}{n}\sum_{i=1}^{n} d(x_i, x_j)$，$d(x_i, x_j)$为 x_i 与 x_j 之间的相异度，通常它是一个非负的数值，当 x_i 与 x_j 之间越相似或接近，其值越接近 0，反之就越大，$d(x_i, x_i) = 0$。相异度有多种计算方法[2]，不同的方法会有不同的聚类效果，这里采用常用的距离作为度量方式。权重越小，说明越相似或越接近；权重越大，说明差异性越大或越远。对于比较密集的数据点，它们距中心点的距离相近，权重是比较接近的，很容易聚类在一簇。而对于一个稳定的系统来说“噪声”和孤立点的数目不会太多，如果太多这个系统就没法使用，在本算法中“噪声”和孤立点的权重稍大，为了消除“噪声”和孤立点数据的影响，采用加权平均的方式来解决。当与别的数据点一起经加权平均后对整体影响远小于直接采用平均的方法。因此，这种权重的计算方法还是比较合理的。

2.2 基于加权改进的 k-平均算法(k-WMeans)

该算法的基本思想是对簇中每个对象计算加权平均值，将数据库中的每个对象(重新)赋给最类似的簇，反复进行这种操作，直到准则函数收敛即使平方误差的总和达到满意的程度。这显然是针对数值属性数据，而符号属性数据直接对簇中对象求权平均值，然后再对数据库中的每个对象重新调整。每一簇的加权平均值的计算方法是：$AWM_j = \frac{1}{t}\sum_{i=1}^{t} w_i p_i$，其中 $AWM_j (1 \le j \le k)$ 表示簇 C_j 的加权平均值(或权平均值)；t 是簇 C_j 中对象的个数，不同的簇 t 值不同；p_i 是空间中的点，表示给定的簇 C_j 中 t 个数据对象之一；w_i 是簇 C_j 中数据对象的权重。准则函数(平方误差总和) $E = \sum_{j=1}^{k}\sum_{p_i \in C_j} | p_i - AWM_j |^2$，其中 p_i 取簇 C_j 中的每一个数据。

k-WMeans 算法：基于簇中对象的加权平均值或权平均值。

输入：簇的数目 k 和包含 n 个对象的数据库。

输出：k 个簇，使平方误差总和 E 最小。

方法：

(1)任意选择 k 个对象作为初始的簇中心；

(2)repeat；

(3)根据簇中对象的加权平均值(或权平均值)，将每个对象(重新)赋给最类似的簇；

(4)更新簇的加权平均值(或权平均值)，即计算每个簇中对象的加权平均值(或权平均值)；

(5)until 不再发生变化。

3 与经典的 k-平均算法的比较

本算法与经典的k-平均聚类算法[5]相比，就是把经典算法中的平均值变成了这里的加权平均值或权平均值，在计算加权平均值或权平均值时会增加一些时间开销，但它的处理能力却大大增强。它不仅能处理数值属性数据，还可以处理符号属性数据，对“噪声”和孤立点数据不怎么敏感，少量的该类数据不会对加权平均值(或权平均值)产生大的影响。该算法的复杂度和经典算法是一致的，也是O(nkt)，其中，n是所有对象的数目，k是簇的数目，t是迭代的次数。因此当k«n且t«n时对处理大数据集是可伸缩的和高效率的。另外，该算法和经典算法一样都需要事先估计簇的个数k，如想得到最优解时必须试探不同的k值。

4 实验结果

本文采用UCI[15]提供的机器学习数据库中的部分数据对k-WMeans算法和k-Means算法进行了测试。对于数值属性数据采用iris, thyroid-disease和glass 3 组数据集；符号属性数据采用 balloon, soybean和zoo 3 组数据集，按k-WMeans算法和k-Means算法分别对它们进行聚类。在表 1 中给出了聚类的结果。其中，第 1 列是数据集的名称；第 2、3、4 列分别给出了数据集的样品个数，决策值的个数，以及条件属性的个数；第 5 列和第 6 列是k-WMeans算法和k-Means算法在数据集上对数据进行分类的精确度。

表 1 k-WMeans 算法和 k-Means 算法在 UCI 数据集上的比较

DataSet	#Instance	#Concept	#Attribute	k-WMeans Accuracy/%	k-Means Accuracy/%
Iris	150	3	4	93.1	89.6
Thyroid-disease	215	3	5	95.2	94.3
Glass	214	7	9	88.7	82.6
Balloon	20	2	4	88.0	0
Soybean	47	4	35	97.3	0
Zoo	101	7	16	95.7	0

由表 1 可以看出 k-WMeans 对数据的聚类结果总体上优于 k-Means，同时它又能较好地聚类符号属性数据。

5 结束语

引进加权方法对现有的 k-Means 算法进行了尝试性的改进，使其减小了孤立点和“噪声”的影响，实验证明了这种基于加权改进的 k-平均聚类算法的有效性。而且这种基于加权的 k-平均算法能够处理符号属性数据，是传统的 k-平均算法所不能达到的。它可以运用到较大的数据库中去，但它能否运用到特大型复杂的数据库中进行聚类数据挖掘还有待进一步的研究。

参考文献

1 史忠植. 知识发现[M]. 北京: 清华大学出版社, 2002.

2 Han Jiawei, Kamber M. Data Mining: Concepts and Techniques[M]. San Francisco: Morgan Kaufmann Publishers, 2000.

3 Grabmeier J, Rudolph A. Techniques of Cluster Algorithms in Data Mining[J]. Data Mining and Knowledge Discovery, 2002, 6(4): 303.

4 Jain A K, Murty M N, Flynn P J. Data Clustering: A Review[J]. ACM Computing Surveys, 1999, 31(3): 264-323.

5 MacQueen J. Some Methods for Classification and Analysis of Multivariate Observations[C]//Proc. of the 5th Berkeley Symp. on Math. Statist. 1967: 281-297.

6 Kaufman J, Rousseeuw P J. Finding Groups in Data: An Introduction to Cluster Analysis[M]. New York: John Wiley & Sons, 1990.

7 Ester M, Kriegel H P, Sander J, et al. A Density-based Algorithm for Discovering Clusters in Large Spatial Databases[C]//Proc. of 1996 Intl. Conf. on Knowledge Discovery and Data Mining, Portland, OR. 1996-08: 226-231.

（下转第 209 页）