

一种改进的特征加权 K-means 聚类算法

王 慧¹, 申石磊^{1,2}

(1 河南大学 计算机与信息工程学院, 河南 开封 457004; 2 河南大学 计算中心, 河南 开封 457004)

摘 要: 提出了一种改进的特征加权 K-means 聚类算法. 该算法首先基于数据样本分布选取初始聚类中心, 然后设计特征加权的 K-means 聚类算法. 实验结果证明, 该算法能产生质量较高的聚类结果, 并且能处理数值、符号两类数据.

关键词: 聚类; K-means 算法; 聚类中心; 特征加权

中图分类号: TP182

文献标识码: A

文章编号: 1000-7180(2010)07-0161-03

An Improved Feature Weighted K-means Clustering Algorithm

WANG Hui¹, SHEN Shi-lei^{1,2}

(1 School of Computer & Information Engineering, Henan University, Kaifeng 475004, China;

2 Computing Center, Henan University, Kaifeng 475004, China)

Abstract: An improved weighted K-means clustering algorithm is proposed. In the presented algorithm, firstly the original clustering center is obtained according to the data sample distribution. Then the improved K-means algorithm using the feature weights is designed. Experiment results have shown that the proposed clustering algorithm can produce high quality clustering steadily and deal with the symbolic data as well as the numeric data.

Key words: clustering; K-means algorithm; clustering center; feature weighting

1 引言

聚类分析是一种重要的数据分析技术, 它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. 目前常用的聚类算法可分为以下几种^[1-4]: 分裂法, 如 K-means、K-medoid 算法等; 层次法, 代表性算法如 CURE 算法等; 基于密度的方法, 典型算法有 DBSCAN 算法、OPTICS 算法等; 基于网格的方法, 如 STING 算法、CLIQUE 算法等; 基于模型的方法, 如 SOM 算法等. 其中, K-means 算法由于其方法简单, 已成为当前最流行的聚类算法之一. K-means 是一种基于划分的聚类算法, 目的是通过在完备数据空间的不完全搜索, 使得目标函数取得最大值. 由于局部极值点的存在以及启发算法的贪心性, 传统的 K-means 算法对初始聚类中心敏感, 从不同的初始聚类中心出发, 得到的

聚类结果也不一样, 并且一般不会得到全局最优解. 并且, K-means 算法只适用于数值型数据. 如何克服对初始聚类中心敏感性, 从而获得稳定的聚类效果, 并进一步拓展 K-means 算法的适用范围, 对 K-means 算法具有重要意义.

在分类领域, 通过特征分析, 发现有些特征是与分类强相关的, 有些特征与分类弱相关, 还有许多特征与分类不相关, 即不同的特征对于分类的贡献可能很不相同. 因此, 在分类研究中很早就提出了许多特征选择及特征加权方法, 以提高分类精度. 近年来, 为了提高聚类质量, 也借鉴和引入了分类领域的上述思想, 提出了一些基于特征加权的聚类算法. 如 Chan 等人提出了一种基于特征加权的类 K 均值聚类算法^[5]; Wang 等给出了一种基于特征加权的模糊 C 均值算法^[6]. 由于这些算法随机选取初始聚类中心, 导致最终的聚类结果并不理想.

收稿日期: 2009-06-22; 修回日期: 2009-09-24

基金项目: 国家“八六三”计划项目(2007AA01Z478)

文中基于数据样本分布选取初始聚类中心,并给出一种特征加权的改进 K-means 聚类算法.这种改进的加权聚类算法,有效克服了传统 K-means 算法对初始聚类中心的敏感性,可产生质量较高的聚类结果.该算法不仅可以处理数值型数据,而且可以处理符号型数据.数值实验验证了算法的有效性和实用性.

2 确定初始聚类中心算法

假设 $X = (x_1, x_2, \dots, x_n)$ 是具有 n 个数据对象的集合, $x_i = (x_{i1}, x_{i2}, \dots, x_{im})$, $y_j = (y_{j1}, y_{j2}, \dots, y_{jm})$ ($i, j = 1, 2, \dots, n$) 分别是具有 m 维变量的数据对象.定义数据对象之间的欧氏距离为:

$$d(x_i, y_j) = \sqrt{(x_i - y_j)^T (x_i - y_j)} \quad (1)$$

数据对象 x_i 与数据集合 S 之间的欧氏距离定义为:

$$d(x_i, S) = \inf\{d(x_i, y_j), y_j \in S\} \quad (2)$$

两个数据对象集合 S_1 与 S_2 之间的欧氏距离定义为:

$$d(S_1, S_2) = \inf\{d(x_i, y_j), x_i \in S_1, y_j \in S_2\} \quad (3)$$

如果用欧氏距离表示数据对象之间的相似度程度,相似的数据对象之间的距离要比不相似的对象之间的距离要小.如果能够找到 k 个初始中心,它们分别代表了相似程度较大的数据集合,那么就找到了与数据在空间分布上相一致的初始聚类中心.基于此,选取初始聚类中心的改进 K-means 算法基本思想为:首先计算各个数据对象之间的欧氏距离,找出距离最近的两个数据对象,形成一个数据对象集合 X_1 ,并将它们从原来的数据集合 X 中去掉,计算对象集合 X_1 中每个数据对象与集合 X 中每个数据对象的欧氏距离,找出具有最小距离的数据对象 x_i ,将其并入 X_1 ,并从集合 X 中删除该对象,如此直到集合 X_1 中的对象个数达到某个阈值 θ ,从而构成最终的数据对象集合 X_1 ,原来的集合 X 变为 $X - X_1$;然后从 $X - X_1$ 集合中找到对象之间距离最近的两个数据对象构成集合 X_2 ,重复上述过程,直到形成 k 个对象集合;接着对 k 个对象集合分别进行算术平均,形成 k 个聚类中心.

在实际操作中,无法预知每个子集合 X_i ($i = 1, 2, \dots, k$) 中的对象个数,很难找到一个最佳值.一般期望,一个聚类中平均的对象个数为 n/k 个,显然阈值 $\theta \leq n/k$.这里阈值 θ 选取 $0.8 * n/k$.

算法描述:

输入:待处理数据集 X ,聚类个数 k, θ ;

输出: k 个聚类中心.

算法步骤:

① $t = 1$;

② 若 $t > k$,则转到④;否则, $t = t + 1$,计算 X 中各个数据对象之间的欧氏距离,找出距离最近的两个数据对象 x_i, x_s ($1 \leq i, s \leq n$),构成数据对象集合 $X_t = \{x_i, x_s\}$,并令 $X = X - \{x_i, x_s\}$;

③ 计算 X 与集合 $\{x_i, x_j\}$ 的距离,从 X 中找到一个距离 $\{x_i, x_s\}$ 最近的数据对象 x_k ($1 \leq k \leq n; k \neq i, j$),令 $X_t = X_t \cup \{x_k\}$, $X = X - \{x_k\}$;

④ 若 X_t 中数据对象个数 $\text{Card}(X_t) < \theta$,则转到③;否则,转到②;

⑤ 对每个集合 X_t ($1 \leq t \leq k$),分别计算其算术平均值,获得 k 个聚类中心;

⑥ 输出 k 个聚类中心点,算法结束.

3 基于改进的特征加权 K-means 算法

3.1 加权策略

在含有 n 个数据对象的数据集合中,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权值.文中采用 Domeniconi 等人提出的权重设置方法^[7],即对类内分布一致性好的特征赋予较大的权重,而且在不同的类内相同的特征赋予不同的权重,类内分布的一致性主要通过类内该特征的方差大小度量.

假设 X 代表整个数据集, X_i 代表第 i 类数据集, x 代表数据对象, E_{ir} 代表特征 r 在第 i 类内的分布方差, w_{ir} 代表特征 r 在第 i 类内的权重, c_k 代表第 k 类的类中心向量,则有以下公式:

$$X_i = \{x \mid i = \arg \min_k d_w(c_k, x)\} \quad (4)$$

$$d_w(c_k, x) = [\sum_{j=1}^d w_{kj} (c_{kj} - x_j)^2]^{1/2} \quad (5)$$

$$E_{ir} = [\sum_{x \in X_i} (c_{ir} - x_r)^2]^{1/2} / \text{Card}(X_i) \quad (6)$$

其中, c_{ir} 表示第 i 类中心的第 r 特征, x_j 表示数据对象 x 的第 r 特征, $\text{Card}(X_i)$ 表示第 i 个数据集 X_i 的数据对象个数.第 i 类的第 r 特征的权重 w_{ir} 定义如下:

$$w_{ir} = \exp(-h * E_{ir}) / (\sum_{k=1}^d \exp(-2h * E_{ik}))^{1/2} \quad (7)$$

这里, h 为正的常数,通常取 $h = 12$.

为了避免由于 E_{ir} 过大而导致 $\exp(-h * E_{ir})$,由于计算精度问题而趋于零值,需要预先对数据对

象进行标准化处理.文中采用的方法为:

$$x_s = x_s / \bar{\omega}_s$$

(8)

其中, $\bar{\omega}_s$ 为数据样本 x_s 的平均值.

通过上述计算求得特征权重后,基于调整后的特征权重,重新计算各数据对象与各聚类中心的距离,重新分配给各数据对象,并通过式(9)来计算新的聚类中心 c_i :

$$c_i = \sum_{x \in X_i} x / \text{Card}(X_i)$$

(9)

3.2 算法描述

结合第 2 节的确定初始聚类中心算法及 3.1 节的加权策略,文中提出了一种基于特征加权的改进 K-means 聚类算法,具体描述如下:

- 输入:待处理数据集 X ,聚类个数 k, θ ;
- 输出: k 个聚类.
- 算法步骤:
- ① 首先调用第 2 节的确定初始聚类中心算法,获得 k 个聚类中心;
- ② 设置初始权重 $w_{ir} = 1/d$;
- ③ Repeat;
- ④ 根据式(4)、(5)将数据对象划分到相应的数据对象集合 X_i 中;
- ⑤ 根据式(6)、(7)计算新的权重系数 w_{ir} ;
- ⑥ 根据式(4)、(5),重新计算各数据对象与当前各聚类中心 c_i 的距离,将数据对象划分到相应的数据对象集合 X_i 中;
- ⑦ 根据式(9),重新计算各聚类中心 c_i ;
- ⑧ Until 算法收敛或达到指定的迭代次数.

4 实验结果

为了验证所提出算法的有效性和可行性,文中选用 UCI 提供的机器学习数据库中的 6 个数据集进行实验,测试文中提出的新算法的聚类性能,表 1 给出了这些数据集的信息.其中,数值属性数据选取 Iris、Wine 和 Abalone 三组数据集;符号属性数据选取 Zoo、Trains 和 Soybean 三组数据.

表 1 选取的 UCI 数据集

Data Set	Number of Instances	Number of Attributes	Number of Classes
Iris	150	4	3
Wine	178	13	3
Abalone	4177	8	29
Zoo	101	17	7
Trains	10	32	2
Soybean	307	35	19

为便于比较,测试随机初始化的传统 K-means 算法及文中提出的改进的特征加权 K-means 算法,测试结果见表 2.传统的 K-means 算法仅能处理数值型数据;而文中提出的改进的特征加权 K-means 算法不仅能处理数值型数据,还能处理符号属性数据.从表 2 中不难看出,随机选取初始聚类中心的传统 K-means 算法得到的聚类结果准确率低且不稳定,而使用改进聚类算法所得到的聚类结果不仅准确率高,而且相对稳定,同时还能较好地聚类符号属性数据,因而更适用于对实际数据的聚类.

表 2 两类算法在 UCI 数据集上的实验结果

Data Set	传统 K-Means 算法准确率	文中算法准确率
Iris	76.38%	90.67%
Wine	74.63%	85.42%
Abalone	62.15%	87.26%
Zoo	0	67.54%
Trains	0	71.33%
Soybean	0	60.23%

5 结束语

针对传统 K-means 算法由于随机选取初始聚类中心而造成聚类结果不稳定、准确率低的问题,提出了一种改进的特征加权 K-means 算法.该算法首先基于数据样本分布进行初始聚类中心点选择,排除了传统选点算法中随机性对结果的影响;并且在数据的预处理方面也作了一些改进,使权值的计算更为稳定可靠.实验结果表明,文中改进的特征加权 K-means 算法所得到的聚类结果不仅准确率高,而且相对稳定,同时还能较好地聚类符号属性数据.

参考文献:

[1] Thodoridis S, Koutroumbas K. Pattern recognition[M]. San Diego: Academic Press, 1999.

[2] Han J, Kamber M. Data mining: concepts and techniques [M]. San Francisco: Morgan Kaufmann Publishers, 2000.

[3] 冯洁,陶宏才.快速挖掘最大频繁项集[J].微电子学与计算机,2007,24(3):69-73.

[4] 毛国君,段立娟,王实,等.数据挖掘原理与算法[M].北京:清华大学出版社,2005.

[5] Chan E Y, Ching W K, Ng M K, et al. An optimization algorithm for clustering using weighted dissimilarity measures[J]. Pattern Recognition, 2004, 37(5):943-952.

Synopsys Hspice 软件进行仿真. 将仿真结果导入 MATLAB 运算后, 得到 ADC 的静态性能如图 7 所示.

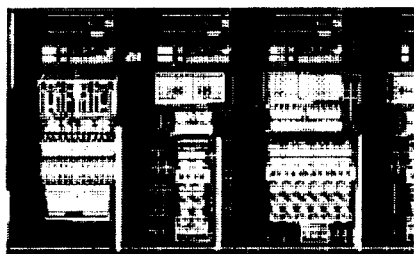


图 6 流水线级电路局部版图

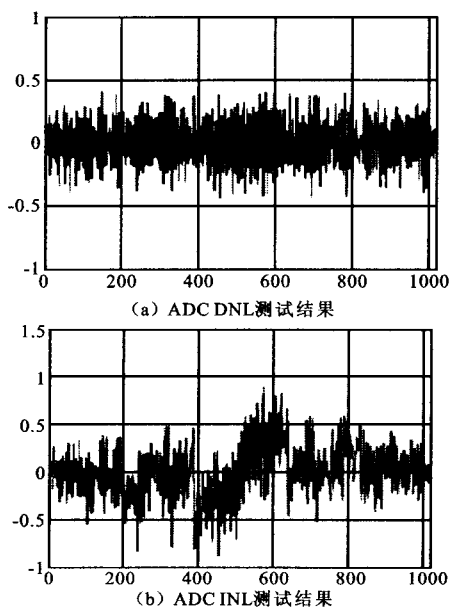


图 7 ADC 静态特性测试结果

从图 7 可以看到: 系统的 DNL 小于 0.5LSB, 满足 CCD 图像采集系统对于无缺码的要求, 同时 INL 值也在 0.8LSB 左右, 达到了良好的动态性能. Hspice 的输出文件显示: 10MHz 正弦输入信号下, 设计的流水线 ADC 功耗为 35.4mW.

5 结束语

文中设计了一款适用于 CCD 图像采集系统的高速、高精度、低功耗流水线 ADC. 通过使用了无静态功耗的动态比较器以及省略输入级采样/保持电路, 有效减小了 ADC 整体功耗和版图面积. 根据仿真结果证明, 整个 ADC 符合 CCD 图像采集系统的性能要求. 在未来的工作中, 考虑后级对于运放的要求逐步降低, 将进一步研究以等比例缩小的运放来减小系统的整体功耗^[6].

参考文献:

- [1] 佟首峰, 阮锦, 郝志航. CCD 图像传感器降噪技术的研究[J]. 光学计算机工程, 2000, 8(2): 140-145.
- [2] Miyazaki D, Furuta M, Kawahito S. A 75 mW 10 bit 120 MSample/s parallel pipeline ADC[C]// Proc. ESSCIRC. Italy, 2003: 719.
- [3] 殷湛, 郭立, 杨吉庆. 一种用于流水线 ADC 的高速电压比较器[J]. 微电子学与计算机, 2006, 23(2): 182-185.
- [4] Bogner P. A 28 mW 10-bit 80-MS/s pipelined ADC in 0.13 μm CMOS[C]// Proc. ISCAS2004. Atlanta, 2004 (1): 17.
- [5] 张凯, 周贵贤, 刘烨. 12 位 100MS/s ADC 中采样/保持电路的分析与设计[J]. 微电子学与计算机, 2007, 24 (11): 8-14.
- [6] Arias J, Bisbal D, San Pablo J, et al. Low power pipelined ADC design for wireless LANs[J]. IEEE Journal of Solid State Circuits, 2004, 39(8): 1338.

作者简介:

徐美华 女, (1957-), 博士. 研究方向为包括高清晰度平板显示与应用集成技术、汽车电子和 IC 设计.

樊裕乐 男, (1984-), 硕士. 研究方向为数模混合信号 IC 设计.

(上接第 163 页)

- [6] Wang X, Wang Y, Wang L. Improving fuzzyc-means clustering based on feature-weight learning[J]. Pattern Recognition Letters, 2004, 25(10): 1123-1132.
- [7] Domeniconi C, Papadopoulos D, Gunopulos D, et al. Subspace clustering of high dimensional data[C]// Proc. of the Fourth SIAM Int. Conf. on Data Mining. California, 2004: 517-521.

作者简介:

王 慧 女, (1981-), 硕士研究生. 研究方向为数据挖掘.

申石磊 男, (1953-), 教授, 硕士生导师. 研究方向为计算机网络、机器学习.