

## 一种基于排序划分的聚类初始化方法

殷君伟, 陈建明, 薛百里, 张 健

(苏州大学 计算机科学与技术学院, 江苏 苏州 215006)

**摘 要:**  $k$ -means 聚类算法, 是在  $d$  维空间  $R^d$  里把  $n$  个数据对象划分为  $K$  个类, 其划分原则是计算每个数据对象与  $K$  个聚类中心的距离并将其分配到最近的一个类. 传统直接  $k$ -means 算法是随机选取初始中心的, 不同的初始中心会产生不同的聚类结果, 针对这个不足, 提出了一种基于排序划分的聚类初始化方法, 该方法简单易实现, 将其应用在真实数据集和模拟数据集上, 实验表明在处理非高维数据上这是一种简单而有效的方法, 在很大程度上提高了聚类精度和效率.

**关键词:**  $k$ -means; 排序划分; 聚类分析; 初始中心

**中图分类号:** TP302

**文献标识码:** A

**文章编号:** 1000-7180(2013)06-0080-04

## An Initialization Method for Clustering Center Based on Sorting and Partition

YIN Jun-wei, CHEN Jian-ming, XUE Bai-li, ZHANG Jian

(School of Computer Science and Technology, Soochow University, Suzhou 215006, China)

**Abstract:** In  $k$ -means clustering, we are given a set of  $n$  data points in  $d$ -dimensional space  $R^d$  and an integer  $K$  the problem is to determine a set of  $K$  points in  $R^d$ , called centers, so as to minimize the mean squared distance from each data point to its nearest center. The initial centers of direct  $k$ -means algorithm are chosen randomly, different initial centers will lead to different results. In this paper, in view of the deficiency of direct  $k$ -means algorithm, we propose a novel method about initial centers based on sorting and partition and apply it to real data as well as simulated data, which show that this is a simple and efficient method to improve the clustering accuracy and efficiency.

**Key words:**  $k$ -means; sorting and partition; clustering analysis; initial centers

### 1 引言

在实际应用中,  $k$ -means 聚类算法已经被证明是令人满意的聚类结果. 然而传统  $k$ -means 算法是一种局部优化策略, 容易陷入局部最优解<sup>[1]</sup>, 而且对初始中心非常敏感<sup>[2]</sup>, 好的初始中心有助于避免陷入局部最优, 除此之外,  $k$ -means 算法还要进行大量的循环迭代, 使效率低下. 因此, 针对传统直接  $k$ -means 算法的不足, 文献[3-6]中提出了改进算法, 但这些算法, 要么改进了初始中心的选择却忽略了循环迭代的优化, 要么优化了时间复杂度却无法避免算法陷入局部最优, 最主要的是这些算法实现非

常复杂. 尤其当这些算法应用于低维数据时, 其执行效果与性能反而下降, 受文献[7]的影响, 从  $k-d$  树自身空间分割的创建过程得到启示, 在聚类初始中心选择方面提出了一种基于排序划分的方法, 这种方法能离散的选择初始中心, 使其在整体上符合数据集的分布从而达到较好的聚类效果, 并加快聚类收敛速度.

### 2 直接 $k$ -means 算法

直接  $k$ -means 聚类算法是一种硬聚类方法, 是在  $d$  维空间  $R^d$  里把  $n$  个数据对象划分为  $K$  个类, 此算法包括两阶段: 首先, 确定所要聚类的准确数目

$K$ ,即把数据集划分为  $K$  类,并随机选取  $K$  个对象作为初始聚类中心;然后对剩余的每个对象,根据其与各中心的距离将它指派给最近的类,再重新计算每个类中对象的平均值作为新的聚类中心.这个过程重复进行,直到聚类收敛为止,如算法1所描述.

#### 算法1

输入:  
 $D=\{x_1, x_2, \dots, x_n\}$  //表示有  $n$  个  $d$  维对象数据集  $D$ .  
 $K$  //聚类数目.  
 输出:  
 $K$  个聚类集合.  
 方法:  
 1. 从  $D$  中任意选择  $K$  个对象作为初始聚类中心,分别代表要分成的  $K$  个类.  
 2. repeat  
 3. 根据距离中心最近的原则,将每个对象指派到最相似的类中.  
 4. 更新类中心,即计算每个类中对象的均值.  
 5. until 满足准则函数

该算法最大的不足是随机选取初始聚类中心,因为对于不同初始中心会产生不同的结果.

### 3 基于排序的 $K$ 划分算法

直接  $k$ -means 算法中初始中心是随机选择的,其存在这样的缺陷:取样比较小的随机选取通常不能表达数据集空间的分布趋势,随机选取很难反映数据集的分布、聚类形状及数据密度变化的状态.因此本文提出了一种基于排序的  $K$  划分选择算法( $k$ -partition based sorting, KPBS),这是一种启发式选择算法,能有效避免上述缺陷问题,与目前各种复杂的初始中心选择算法相比,在非高维数据处理上,此算法简单有效.

#### 3.1 算法思想

KPBS 的基本思想是采用启发式的方法,使所选初始中心与数据集的分布相关,即所选初始中心能反映数据集的分布,如此在一定程度上便能避免陷入局部最优. KPBS 算法首先对输入数据集排序,然后把排序后的数据集划分成  $K$  个子集,  $K$  为所要聚类的数目,最后把每个子集的平均值作为初始聚类中心.为进一步清晰的阐述这种思想,图1为数据集  $D$  的分布图(符合高斯分布),从中可以判断出其分布情况及聚类形状,我们的算法思想首先是按分布的离散度对数据排序,则排序后的第一个数据和

最后一个数据的相似度最小,反应在数据点上就是这两个点最离散或距离最远,如图2中点  $X_1$  与  $X_2$ ,接下来对数据集进行  $K$  均分,此处选择分为3类,反映在图中就是把线段  $x_1x_2$  均分为三段,直线  $L_1$ ,  $L_2$  为线段  $x_1x_2$  的均分垂直分割线,把空间分成了三部分(图2中 I、II、III),可以看到数据点大体上准确的划分在了各个空间,最后计算各个空间点集的均值作为初始聚类中心.

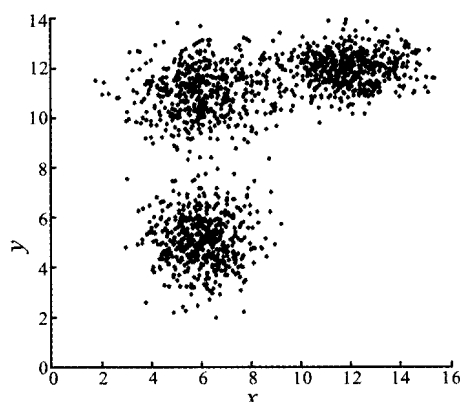


图1 含3个聚类的数据集分布

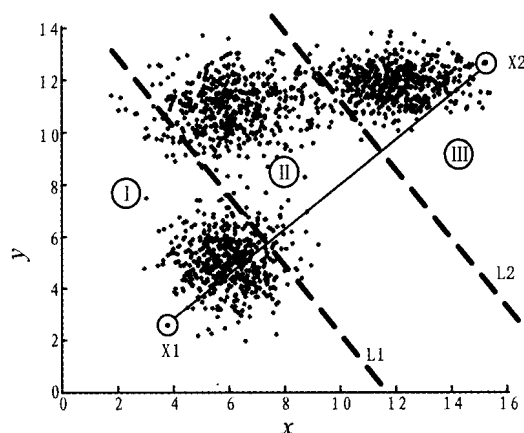


图2 数据空间划分

#### 3.2 算法设计

存在一个问题,一般来说,我们所要处理的数据都是多维数据,每一个数据对象  $X_i$  包含多个属性,例如  $X_{i1}, X_{i2}, \dots, X_{id}$ ,  $d$  表示数据对象的属性数目或维的个数.这种情况下,首先要确定排序维,即选择在哪一个属性上排序,在此我们采用最大离散度的方式确定排序维,即这个属性越离散越能反映数据集的分布特征.在整个数据集上计算每一属性维的方差,然后把方差最大的属性作为排序维,并按此排序维对数据集进行升序排序,一般来说,要处理的数据集都比较大,所以选用堆排序的方法.接着把排

序后的数据集平均划成  $K$  个子集,最后计算这  $K$  个子集的算术平均值,并将其作为初始聚类中心,具体算法设计如算法 2 描述。

#### 算法 2

输入:

$D = \{x_1, x_2, \dots, x_n\}$  // 表示有  $n$  个  $d$  维对象数据集  $D$ .

$K$  // 聚类数目.

输出:

$K$  个聚类中心.

方法:

阶段 1

1. 对整个数据集,计算每一维的方差:

$$D(d_j) = \frac{\sqrt{(X_1 - E(d_j))^2 + (X_2 - E(d_j))^2 + \dots + (X_n - E(d_j))^2}}{n}$$

$$= \frac{\sqrt{\sum_{i=1}^n (X_i - E(d_j))^2}}{n}, 1 \leq i \leq n, 1 \leq j \leq d, d \text{ 表示维数}, E$$

$(d_j)$  表示第  $j$  维的平均值.

2. 确定方差最大的属性维  $d_k (1 \leq k \leq d)$  作为排序维.

3. 对整个数据集基于排序维  $d_k$  按升序排序.

4. 把排好序的数据集平均划分为  $K$  个子集.

5. 计算步骤 4 中获得的  $K$  个子集的均值,记为:  $C_1, C_2, \dots, C_k$ , 这些均值即作为初始聚类中心.

阶段 2

6. 由阶段 1 获得初始聚类中心:  $C_1, C_2, \dots, C_k$ ,

$i (1 \leq i \leq K)$  个聚类中对象个数为  $M_i$ , 并记第  $i (1 \leq i \leq K)$  个聚类中对象组成的集合为  $T_i$ , 设  $c$  为迭代次数, 令  $i=1, c=1$ .

7. 计算其他对象  $x_i (1 \leq i \leq n)$  与每个中心  $C_i (1 \leq i \leq K)$  的欧几里德距离, 把对象  $d_i$  指派到最近的类集合  $I_j$  中.

8. 计算第  $i (1 \leq i \leq K)$  个聚类的中心:

$$C_i = \begin{cases} \frac{1}{M_i^{-1}} \sum_{x \in T_i^{-1}} x, M_i^{-1} > 0 \\ 0, \text{其他} \end{cases}$$

如果  $i=K$ , 那么转向步骤 4, 否则  $i=i+1$ , 重复此步骤.

9. 计算准则函数的值:

$$J_1 = \sum_{i=1}^K \sum_{x \in T_i^{-1}} ||x - C_i||^2 \text{ 然后转向步骤 10.}$$

10. 计算新的分配方式, 对每一对象  $x_j (1 \leq j \leq n)$ , 计算其最近聚类中心:  $\min(x_i, C_i)$ , 则把  $x_j$  分配到第  $i$  个类中.

11. 计算准则函数的值:  $J_2 = \sum_{i=1}^K \sum_{x \in T_i^{-1}} ||x - C_i||^2$ ,

然后转向步骤 12.

12. 如果  $|J_1 - J_2| < \delta$ ,  $\delta$  表示收敛精度, 那么停止计算; 否则令  $i=1, c=c+1$ , 转向步骤 8.

与随机选取聚类中心的直接  $k$ -means 算法相

比, 本文所提出的方法依据数据集的分布, 所选择的初始中心与聚类形状相关, 因此, 不仅能提高聚类结果的准确, 还能大大减少迭代次数, 从而加快聚类收敛, 由于采用了排序的方法, 这也减少了对象间比较次数, 从而降低时间复杂度.

#### 3.3 时间复杂度分析

直接  $k$ -means 算法中, 初始中心是随机抽取的, 其时间主要耗费在指派每个对象到最近的聚类集合, 即用于算法收敛前的计算, 则时间复杂度为  $O(nkl)$ , 其中  $n$  表示数据集中数据对象的个数,  $k$  表示聚类个数,  $l$  表示迭代的次数.

在本文提出的初始中心选择算法中, 首先计算每一维的方差需要  $O(n)$  的时间,  $n$  表示数据对象个数, 在第二步中求出方差最大的一维为排序维, 其需要  $O(nm)$  的时间,  $m$  表示对象为  $m$  维数据, 接下来是按排序维进行数据排序, 采用堆排序, 其时间复杂度为  $O(n \log n)$ , 把  $n$  个数据对象进行  $K$  等分并计算每一部分的均值花费的时间为  $O(n)$ . 由于  $m$  远小于  $n$ ,  $O(nm)$  与  $O(n \log n)$  相比可忽略, 因此整个初始中心选择算法的时间复杂度为  $O(n \log n)$ .

在第二阶段, 指派每个数据对象到最近聚类中心与直接  $k$ -means 算法相同, 所花费的时间为  $O(nkl)$ . 然而由于所选择的初始中心符合数据分布, 并且对数据进行了排序, 这在很大程度上减少了数据对比时间的花费, 也就是能降低迭代次数, 加快收敛速度, 即此处的  $l$  会减小. 因此, 此算法的整体时间复杂度在最坏的情况下为  $O(n \log n + nkl)$ .

#### 4 实验结果

为测试所提出算法(KPBS)的精度和性能, 本文使用 UCI 机器学习数据库中的三个基准数据集 (Breast Cancer-Wisconsin, Thyroid, E-Coli) 作为实验数据, 数据特征见表 1. 除了与直接  $k$ -means 算法对比外, 还与文献[8]中提出的算法(E-Kmeans)进行对比分析, E-Kmeans 是在循环迭代方面改进的一种算法, 它记录前一次迭代中每个对象到所属聚类中心距离, 在下一次迭代时只需对比与新的聚类中心的距离, 若上一次距离小于新距离, 则不再与其他  $K-1$  个聚类中心对比, 仍属于原先的聚类, 因此它的效率要高于直接  $k$ -means 算法.

本文采用如下实验方案评估 KPBS 算法的准确率和效率, 直接  $k$ -means 算法和 E-Kmeans 算法初始中心阶段采用随机选择法, 本文的 KPBS 算法则

在迭代阶段与直接  $k$ -means 相同. 其中准确率是通过把聚类结果与 UCI 数据集中标准聚类作对比得出的, 定义如下:

$$\text{准确率: } f = \frac{\sum_{t=1}^p \alpha_t}{n} \times 100\%$$

式中,  $n$  是所有的对象数,  $p$  是正确的分类数,  $\alpha_t$  是经过某算法聚类后分到第  $t$  类的对象数.

表 1 数据集特征

| 数据集                       | 对象数目 | 属性数 | 标准聚类数 |
|---------------------------|------|-----|-------|
| Breast - Cancer - Wiscon- | 669  | 10  | 2     |
| sin Thyroid               | 215  | 5   | 3     |
| Ecoli                     | 336  | 7   | 8     |

本实验结果是运行多次后的均值, 具体如表 2 所示, 比较直接  $k$ -means 算法和 E-Kmeans 算法, 可以看到在准确率上相差不大, 因为它们都是随机选取初始中心的, 然而在时间开销上 E-Kmeans 算法明显低于直接  $k$ -means 算法, 再对比本文提出的 KPBS 算法, 整体上准确率有较大提高, 尤其在 Thyroid 数据集上结果明显, 对于 Breast Cancer 数据集聚类准确率都很高且相差不大, 主要是因为此数据集本身就只分为良性和恶性乳癌两类, KPBS 算法在时间开销上与 E-Keans 算法相当, 说明与直接  $k$ -means 算法相比, 准确率和效率都有了很大提高.

表 2 实验结果对比

| 数据集              | 算法         |           |           |           |           |           |
|------------------|------------|-----------|-----------|-----------|-----------|-----------|
|                  | $k$ -means |           | E-Kmeans  |           | KPBS      |           |
|                  | 准确率/<br>%  | 时间/<br>ms | 准确率/<br>% | 时间/<br>ms | 准确率/<br>% | 时间/<br>ms |
| Breast<br>Cancer | 92.4       | 78        | 94.2      | 66        | 96.6      | 62        |
| Thyroid          | 77.8       | 60        | 76.4      | 52        | 85.5      | 51        |
| Ecoli            | 72.3       | 64        | 74.5      | 48        | 79.7      | 45        |

上面的实验是应用于真实的数据集, 为了进一步检验本文算法在运行时间方面的表现, 我们采用多元高斯混合分布的方式随机生成了八个数据集, 这些数据集中数据对象数目依次递增, 将上述的三种算法应用于这八个数据集, 所得聚类运行时间与数据集大小的关系如图 3 所示, 与直接  $k$ -means 算法相比, 本文算法在性能上有很大提升, 这表明聚类初始中心的选择符合数据分布, 能减少循环迭代次数, 加快聚类收敛速度.

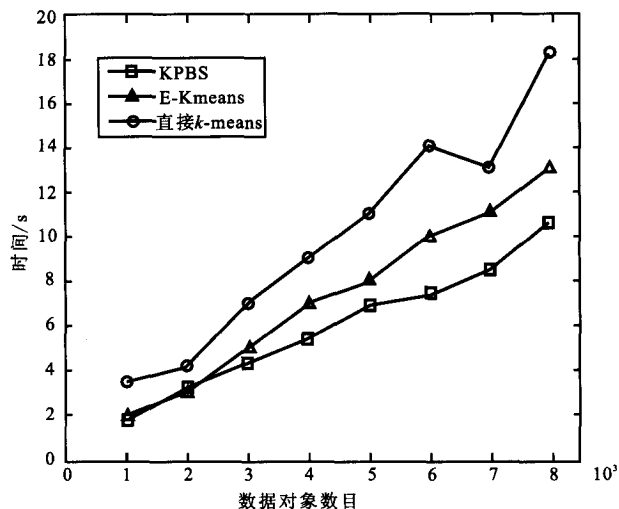


图 3 不同数据集的执行时间

## 5 结束语

本文提出了一种基于排序划分的聚类方法, 可确保产生的初始中心能代表数据集的分布, 提高聚类的精度与效率. 与直接  $k$ -means 算法相比, 实验结果表明本文所提出的算法能用更少的时间开销产生更准确的聚类结果.

本文方法的局限是不适合处理高维数据, 同时还要把聚类数  $K$  值作为一个输入条件, 由于数据集分布的不同,  $K$  值的选取与聚类精度有很大关系, 这两方面也正是下一步的主要研究内容.

## 参考文献:

- [1] Xu R, Donald Wunsch II. Survey of clustering algorithms[J]. IEEE Trans on Neural networks, 2005, 16(3):645-678.
- [2] Pena J M, Lozano J A, Larranaga P. An empirical comparison of four initialization methods for the  $k$ -means algorithm [J]. Pattern Recognition Letter, 1999, 20(10):1027-1040.
- [3] Anferberg M R. Cluster Analysis for Applications [M]. New York: Academic Press, 1973:32-38.
- [4] Khan S S, Ahmad A. Cluster center initialization algorithm for  $k$ -means clustering [J]. Pattern Recognition Letters, 2004, 25(11):1293-1302.
- [5] Birgin E G, Martinez J M, Ronconi D P. Minimization subproblems and heuristics for an applied clustering problem [J]. European Journal of Operational Research, 2003, 146(1):19-34.
- [6] Corral A, Alnendros JM. A performance comparison of distance-based query algorithms using R-trees in

(下转第 87 页)

此时系统可以快速的检测到入侵物体,同时用 Mad-play 播放录制好的音频文件进行语音报警. 智能监控实现流程如图 4 所示.

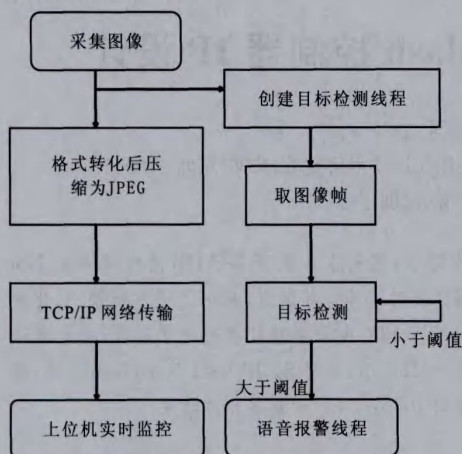


图4 智能监控实现流程

#### 4.4 监控平台测试

在 Linux 平台输入以下命令,启动监控系统. 在 PC 上位可以通过 MJPG\_Streamer 软件自带的图形界面进行显示,也可以通过网页浏览器进行页面显示. 经试验,当有物体进入场景时,能够快速的进行语音报警. 当平台测试完毕后,把 MJPG\_Streamer 设置为在 Linux 启动后自动运行. 见图 5.



图5 系统平台测试

```

mjpg_streamer -i "/lib/WebCam/input_uvc.
so" -o "/lib/WebCam/output_http.so" -w 192.
168.1.6:8080"
  
```

#### 5 结束语

本文设计的基于软件 MJPG\_Streamer 视频监控系统,具有实时性好,远程监控,上位机简易,开发周期短,易于实现的特点. 系统实现了语音报警的功能,充分利用了 ARM 平台的特性,不需要设计额外的语音模块电路. 系统的目标检测功能已完成,下一个研究方向是利用软件 MJPG\_Streamer 对云台控制的支持,实现模式识别和跟踪算法,使系统的功能更强大,应用范围更广.

#### 参考文献:

- [1] SuMung. S3C2440A User's manual 2004 [M]. Korea: SUMUNG Co. L td 2004:62-82/353-385.
- [2] 高丽萍,周斌. 嵌入式 LinuxOV9650 驱动程序的实现[J]. 软件导刊,2012,11(4): 73-75.
- [3] 徐宝玺. 基于 S3C2410 的 DM9000 网卡驱动移植[J]. 天津市经理学院学报,2011,10(5):81-84.
- [4] 霍燃,高丽萍,陈庆奎. 嵌入式 Linux 系统下基于 UDA1341 芯片的音频驱动程序设计与实现[J]. 计算机应用与软件,2012,29(4):16-19.
- [5] 徐家,陈奇. 基于 V4L2 的视频设备驱动开发[J]. 计算机工程与设计,2010,31(16):3569-3572.
- [6] 陈书益,黄永慧. 开源视频服务器软件 MJPG-streamer 的研究和应用[J]. 电子设计工程,2012,20(5):172-176.
- [7] 关向荣,任金昌. 视频监控中背景的提取与更新算法. 微电子学与计算机,2005,22(1):95-97.

#### 作者简介:

肖儿良 (1969—),博士,副教授. 研究方向为信息采集与处理和嵌入式系统.

(上接第 83 页)

- spatial databases [J]. Information Sciences, 2007, 177(11): 2207-2237.
- [7] Stephen J, Tedmod, Conor Heneghan. A method for initializing the  $k$ -means clustering algorithm using kd-trees [J]. Pattern Recognition Letters, 2007, 28(8): 965-973.
- [8] Fahim A M, Salem AM, Torkey A et al. An Efficient enhanced  $k$ -means clustering algorithm [J]. Journal of Zhejiang University, 2006,10(7):1626-1633.

#### 作者简介:

殷君伟 男,(1987—),硕士研究生. 研究方向为智能信息处理、数据挖掘.

陈建明 男,(1960—),副教授,硕士生导师. 研究方向为智能信息处理、智能交通、数据挖掘.

薛百里 男,(1991—). 研究方向为智能信息处理.

张健 男,(1988—),硕士研究生. 研究方向为智能信息处理.