基于 KD 树子样的聚类初始化算法①

潘章明

(广东金融学院 计算机科学与技术系，广州 510521)

摘 要: 在处理大数据集聚类初始化问题时，随机子样法是一种重要的数据约简操作。对随机取样的过程、特征及缺陷进行了分析，提出一种基于 KD 树子样的聚类初始化方法。该方法利用 KD 树将样本空间以递归方式细分成多个子空间，并分别在各子空间中随机取样形成 KD 树子样，有效避免了随机子样分布有偏的不足，使得子样中好的聚类初始点也能很好的表达整个数据集的聚类结构。仿真结果表明，该方法选择的聚类初始点更加接近期望的聚类中心，能获得更高的聚类精度。

关键词: 聚类初始化；KD 树；子样；K 均值算法

Initialization Algorithm of Clustering Using Subsample for KD-Tree

PAN Zhang-Ming

(Department of Computer Science and Technology, Guangdong University of Finance, Guangzhou 510521, China)

Abstract: In the field of initialization of clustering for large data set, random sampling is used as an important reduction operation. This paper focuses on the process and property of random sampling, and proposes a novel random sampling method which is based on KD-Tree samples. Sample spaces were further divided into several sub spaces using KD-Tree. KD-Tree samples were created for each sub-space. This overcomes the defect of skewness of the random samples. Thus the good initial centroids can well describe the clustering category of the whole data set. The experiment results show that the cluster initial centroids selected by the new method is more closed to the desired cluster centers, and the better clustering accuracy can be achieved.

Keywords: clustering initialization; KD-tree; subsamples; K-means algorithm

基于目标函数的聚类算法(如 K-Means、EM 等)本质上是一种局部搜索的爬山算法，具有对初始值敏感、易于陷入局部极小的缺陷[1]，因此，大数据集聚类时，选择或构造一组接近全局最优位置的聚类初始点是提高聚类性能、加速聚类算法收敛进程的重要手段。

根据聚类初始化时应用的数据范围不同，聚类初始化主要有基于数据集的初始化方法[2,3]和基于随机子样的初始化方法[4,5]。随机子样初始化将聚类初始化操作从数据集转移到规模较小的随机子样中，并以子样的聚类初始点表示整个数据集的聚类初始点，从而提高聚类初始化的效率。但是，以传统方法获取的随机子样存在如下缺陷：取样比例较小的随机子样分布是有偏的，通常不能表达数据集空间分布趋势；数据集的空间分布范围可能因抽样而明显收缩；数据集中规模小的聚类可能在抽样中丢失。因此，随机子样很难维持数据集的分布、聚类形状及密度变化的趋势，无法有效地从随机子样中获得好的聚类初始点。

本文从改进随机取样方法入手，使用 KD 树对数据集空间进行分割，在树的叶节点按比例取样形成 KD 树子样，然后从子样中获得聚类初始点。仿真结果表明，KD 树子样能有效表达数据集的分布特征，来自 KD 树子样中的聚类初始点可以获得更优的聚类性能。

① 收稿时间:2010-04-27;收到修改稿时间:2010-05-29

1 KD树简介

KD 树是由 Bentley[6,7]提出的一种多维数据结构，被广泛用于高维空间数据索引和查询，它通过超平面把一个空间递归划分为两个子空间。设 D 为样本维数，d 为分割维索引，$h \in [d_{\min}, d_{\max}]$，其中 $d_{\min}$ 和 $d_{\max}$ 分别为超矩形第 d 维的下界和上界。若 D 维超矩形被一个正交于第 d 维的超平面分割为两个子超矩形，则这个超平面可以表示为：

$$H = \left\{ x \in R^D ; x_d = h \right\} \tag{1}$$

被超平面分割为两个子超矩形 R_l 和 R_r 可分别表示为：

$$R_l = \left\{ x \in R^D ; x_d \le h \right\} \tag{2}$$

$$R_r = \left\{ x \in R^D ; x_d > h \right\} \tag{3}$$

在构建 KD 树过程中，选择分割维 d 及在分割维上确定分割超平面位置 h，具有多种策略[8]。本文旨在应用 KD 树将样本集空间分割成多个样本数量相近的子空间，然后依次在各子空间中取样，因此，d 在 D 个维中依次循环取值，h 取超矩形中第 d 维的中位数。

2 基于KD树子样的聚类初始化方法

2.1 基于 KD 树的随机抽样

KD 树子样的产生包括两个步骤：(1)利用 KD 树将数据集空间分割成多个子空间，每个子空间对应 KD 树的一个叶节点(leaf node 或 bucket)，叶节点中包含样本子集，各叶节点中的样本数量接近；(2)在叶节点中按比例抽样，获得样本子集 $P_i(i=1,2,\ldots,n)$，n 为 KD 树中叶节点的数量，则 KD 树子样 $P_{sub} = P_1 \cup P_2 \cup \cdots \cup P_n$。显然，子空间划分的细致程度影响着子样的代表性：子空间划分越粗，叶节点中的样本越多，取样越粗糙，子样的代表性就越低；子空间划分越细，叶节点中的样本越少，抽样越精细，子样的代表性就越高。同时，叶节点中的样本数量和 KD 树划分的深度也相关，叶节点中的样本越多，则 KD 树的深度越浅，KD 树分割的时间代价就越低，反之亦然。因此，KD 树空间分割何时终止，应考虑数据集规模、空间维数及样本分布等因素，同时也要寻求子样代表性和空间分割时间代价之间的平衡。设 N 为样本数，K 为聚类数，L 为细分因子，每个叶节点的最大样本数 N_{leaf} 由式(4)表示：

$$N_{leaf} = \frac{N}{L * K} \tag{4}$$

式(4)说明：样本集空间分割的粗细程度，由数据集的聚类结构确定，并且确保在平均意义上，一个聚类被分割成 L 个子块。显然，如果一个聚类被分割为 10 个子块(即 L=10 时)，再分别取样，应该是非常精细的。图 1 给出了时，不同取样比例下某二维空间数据集(包含 9 个聚类，共 650 个样本)及 KD 树子样的分布情况，其中，图 1(a)为数据集分布，图 1(b)～图 1 (f)分别是从(a)中按 0.1、0.2、0.3、0.4、0.5 的比例取样结果。可以看出，使用基于 KD 树的取样方法，即使很小的取样比例，子样也能很好的反映数据集的分布特征。

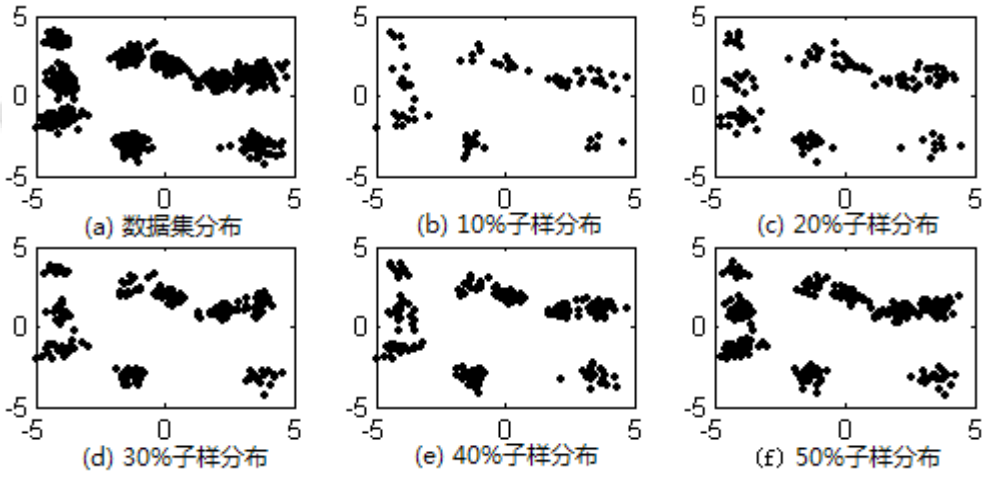

图 1 数据集及不同比例 KD 树子样的分布

2.2 从 KD 树子样到数据集的聚类初始点

本文使用在 KD 树子样中多次运行 K-Means 聚类，并从中选择最好的聚类结果作为数据集的聚类初始点。相对于数据集来说，子样规模通常较小，因此在子样中多次运行 K-Means 算法，不会明显增加整个数据集聚类的时间代价。下面给出该方法的伪码。

算法 1. 从 KD 树子样到聚类初始划分

subsamples_initial(Psub, K, J)

begin

CMS←ϕ;

for i=1 to J

SP←kd_tree_random(Psub, K);

CMi←kmeans(SP, Psub, K);

CMS←CMSCMi;

end for

FM←$\arg\min_{CM_i}\{Distortion(CM_i, P_{sub})\}$;

return FM;

end

在算法 1 中，P_{sub} 为 KD 树子样，K 为聚类数，J 表示在 KD 树子样中重复运行 K-Means 算法的次数，SP 为 K-Means 的聚类初始点，CM_i 为第 i 次聚类结果，$Distortion(CM_i, P_{sub})$ 用于在 P_{sub} 中计算 CM_i 的聚类目标函数值(即误差平方和)，FM 是 J 次聚类中目标函数最优的结果。

为了提高 KD 树子样中 K-Means 算法的聚类性能，kd_tree_random 函数充分利用了 KD-Tree 空间分割的成果，改进了传统的随机初始化方法。具体做法是：首先从构成子样 P_{sub} 的 n 个子集 $P_i(i=1,2,...,n)$ 中随机选择 K 个子集，然后从每个子集中随机选择一个样本，作为 K-Means 的 K 个聚类初始点，这样可以使 SP 中样本点在 KD 树子样中尽可能分散。

2.3 算法流程

基于 KD 树子样的聚类初始化方法主要包括：KD 树空间分割、叶节点取样、从 KD 树子样中获取聚类初始点等三个步骤。

算法 2. 基于 KD 树子样的聚类初始化

kd_tree_initial(samples, K, J, scale, L)

begin

KDTree←build_kdtree(samples, K, L);

subsamples←get_subsamples(KDTree, scale);

FM←subsamples_initial(subsamples, K, J);

return FM;

end

函数 build_kdtre 构建一个 KD 树，树中每个叶节点中样本的最大数 N_{leaf} 由式 (4) 计算；函数 get_subsamples 从 KD 树的叶节点中按比例 scale 进行取样，得到 KD 树子样 subsamples；函数 subsamples_initial 在子样 subsamples 中获取聚类初始点 FM。

3 仿真及结果

3.1 实验数据准备

3.1.1 模拟数据

以随机方法产生多元高斯混合分布的方式，生成模拟数据集，每个数据集由 K 个多元高斯分布组成。均值向量 $\mu_k(k=1,2,...,K)$ 的每一维在区间[-5,5]中以均匀分布随机产生。协方差矩阵 Σ_k 对角线上的元素在区间[$0.03\sqrt{D}, 0.15\sqrt{D}$]中以均匀分布随机产生，D 为样本维数。使用样本维数 D 表示区间，是确保在高维空间中随机产生的聚类具有一定的分离度。为了使随机产生的聚类的轴方向不必平行于坐标轴，对协方差矩阵 Σ_k 进行了旋转，得到 $\tilde{\Sigma}_k = Q_k \Sigma_k Q_k^{-1}$，其中，k 表示第 k 个聚类，$Q_k$ 为正交矩阵。Q_k 通过对随机产生的 D×D 矩阵 A_k 进行 QR 分解获得，即 $A_k = Q_k R_k$，其中 R_k 为上三角矩阵。此外，为了使模拟数据集更具有现实意义，数据式随机产生。

按照上述方法，产生了 8 个模拟数据集，每个数据集都由 10 个聚类构成（即 K=10），8 个数据集的维数 $D=\{2,5,10,20,30,40,50,100\}$。

3.1.2 真实数据

选择两个真实的数据集[9]检验本文方法的性能，第 1 个数据集是“Pen-Based Recognition of Handwritten Digits”，由 10992 个实例构成，16 个属性，10 个聚类。

第 2 个数据集是“Image Segmentation”，由 2310 个实例构成，19 个属性，7 个聚类。

3.2 实验方法

为了检验本文聚类初始化方法(简称 KDTI)的性能，将 KD 树子样中构造的聚类初始点作为数据集 K-Means 聚类的输入，然后计算 K-Means 算法收敛后的误差平方和。同时，将 KDTI 和文献[10]中的 Forgy Approach(即随机子样法，简称 FA)以及文献[4]中的 Refining Initial(简称 RI)聚类初始化方法的结果进行对比。在比较过程中，J 取值 5，取样比例 scale 取 0.1，L 取 10，每个算法对每个数据集重复运行算法 10 次，取其均值进行比较。

此外，为了检测 KDTI 和 RI 方法对取样比例的依赖特征，针对 ds_50 数据集，在取样比例依次是 30%，25%，20%，15%，10%，8%，6%，4%，2%的情况下，比较两个初始化方法的效果。

值得说明的是， KDTI 中的 J 表示对 KD 树子样重复聚类的次数，而 RI 中的 J 表示重复抽取子样并聚类的次数。

3.3 实验结果

表 1 为 FA、RI 及 KDTI 针对不同数据集初始化后，再经过 K-Means 聚类后获得的误差平方和对比。图 2 显示了数据集 ds_50 在取样比例取不同值时，RI 和 KDTI 的聚类性能变化。

由表 1 可以看出，本文方法获得聚类初始点用于数据集聚类的效果普遍好于 FA 和 RI 方法，并且在高维空间中本文方法也能表现出很好的性能。图 2 表明在不同取样比例下，本文方法均能在子样中获得较好的聚类初始点，并且在取样比例逐渐变小的情况下，聚类初始点的质量只略呈下降趋势。

表 1 不同初始化方法的性能对比

数据集	维数	误差平方和		
		FA	RI	KDTI
ds_2	2	183.9002	180.6617	161.9922
ds_5	5	5.57E+03	4.05E+03	2.43E+03
ds_10	10	2.96E+04	2.45E+04	1.93E+04
ds_20	20	2.13E+05	1.79E+05	6.97E+04
ds_30	30	3.58E+05	3.51E+05	2.57E+05
ds_40	40	6.73E+05	5.58E+05	4.91E+05
ds_50	50	1.57E+06	1.32E+06	1.13E+06
ds_100	100	7.30E+06	2.51E+06	2.48E+06
image	16	1.54E+07	1.46E+07	1.32E+07
pen	19	5.93E+07	5.38E+07	5.05E+07

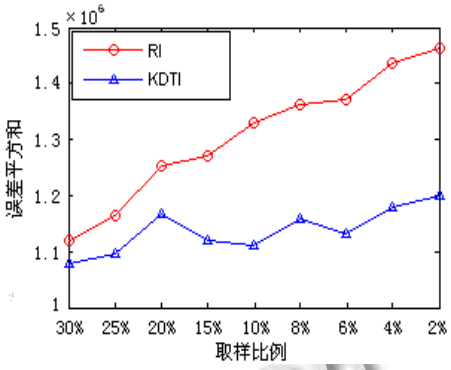

图 2 不同取样比例下两种初始化方法的性能对比

4 结束语

基于传统的随机子样很难反映原数据集分布、聚类形状及密度变化的趋势，分析了问题存在的原因，提出了一种基于 KD 树子样的聚类初始化方法。新方法利用 KD 树实现空间快速分割并分散取样，使得子样能很好的反映数据集的分布趋势，从而将聚类初始化成功的限制到规模很小的子样中，有效提高了从子样中获取聚类初始点的质量，聚类性能获得了的改善。实际上，本文给出了一种数据集约简的方法，对于大数据集和时间代价较高的聚类算法(如基于进化的自动聚类等)而言，数据集约简是一种提高运行效率的可行途径，值得进一步研究。

参考文献

1 Xu R, Donald Wunsch II. Survey of clustering algorithms. IEEE Trans. on Neural networks, 2005,16(3):645－678.

2 He J, Lan M, Tan CL, et al. Initialization of cluster refinement algorithms: a review and comparative study. Proc. of Int'l Joint Conference on Neural Networks. 2004: 297－302.

3 Arai K, Barakbah AR. Hierarchical K-means: an algorithm for centroids initialization for K-means. Reports of the Faculty of Science and Engineering, 2007,36(1):25－31.

4 Bradley PS, Fayyad UM. Refining Initial Points for K-Means Clustering. In: Shavlik J, ed. Proc. of the Fifteenth Int'l Conf. on Machine Learning (ICML). AAAI Press, 1998. 91－99.

5 Rocke DM, Dai J. Sampling and Subsampling for Cluster Analysis in Data Mining. With Applications to Sky Survey Data, Data Mining and Knowledge Discovery, 2003,7(2):215－232.

6 Bentley JL. and Friedman JH. Data structures for range searching. ACM Computing Surveys, 1979,11(4):397－409.

7 Tamminen M. Comment on quad- and octrees. Communications of the ACM, 1984,30(3):204－212.

8 仇明华,殷丽华,李斌.基于多维二进制搜索树的异常检测技术.计算机工程与应用,2007,43(22):122－125.

9 Alpaydin E, Alimoglu F. UCI Repository of Machine Learning Databases. http://archive.ics.uci.edu/ml/, 2009, 11.

10 Forgy E. Cluster analysis of multivariate data: efficiency vs. interpretability of classifications. WNAR meetings, Univ of Calif Riverside, number 768, 1965.