

基于 PDS 和 ENNS 的快速 K-Means 聚类算法

禹贵辉,潘志斌,乔瑞萍,邹 彬,姜彦民

(西安交通大学 电子与信息工程学院,陕西 西安 710049)

摘 要: 在将部分失真搜索算法 PDS,等均值最近邻搜索算法 ENNS 集成到 K-Means 算法迭代过程中的基础上,进一步利用迭代过程中已获取的历史索引信息构造优先搜索序列来减小 K-Means 算法的计算量,降低时间开销.实验结果表明,此算法提高了聚类的速度,在利用标准测试 Lena 图生成不同尺寸码书的情况下,能够将计算时间降至传统全搜索 K-Means 的 8.6%~14.5%.

关键词: K-Means 算法;PDS;ENNS;聚类速度

中图分类号: TP311

文献标识码: A

文章编号: 1000-7180(2011)06-0016-06

Fast K-Means Algorithm Based on PDS and ENNS

YU Gui-hui, PAN Zhi-bin, QIAO Rui-ping, ZOU Bin, JIANG Yan-min

(School of Electronic and Information Engineering, Xi'an Jiaotong University, Xi'an 710049, China)

Abstract: In this paper, PDS and ENNS methods are firstly plugged in the search stage of K-Means iteration. Then by using the indexes obtained from previous iterations, we propose a priority search list(PSL) in order to achieve a smaller value of initial minimum sooner to accelerate K-Means clustering. The experimental results show that the improvements are remarkable without degrading the output performance of K-Means algorithm using 4 test data sets, and the proposed method can reduce computational time to 8.6%~14.5% in the case of generating different size codebooks using the image "Lena".

Key words: K-Means; PDS; ENNS; clustering speed

1 引言

根据采用的方法,聚类算法可分作层次法、划分法、基于密度的方法、基于网格的方法等几大类^[1], K-Means 作为一种典型的基于划分法的聚类方法在模式识别、数据挖掘等领域被广泛使用. K-Means 通过迭代的方法根据最近邻原则不断地对数据点进行重新划分,然后更新聚类中心,直至收敛得到一系列稳定的聚类中心.在划分阶段每个数据点都需要搜索所有的聚类中心以寻找其最近邻中心,因此运算量非常大,使得 K-means 在处理高维的大数据集时时间开销很大,严重影响了其实际应用.为了提高 K-Means 算法速度,许多文献提出了各种快速 K-Means 算法.文献[2]利用三角不等式排查原则,避

免了搜索过程中的不必要的距离计算,但三角不等式需要计算聚类中心间的距离,以及相应的空间来存储这些距离,增加了额外的计算代价和存储代价.文献[3-4]利用 kd-树结构来加速 K-Means 过程,能够取得与数据维数无关的计算量下降,对于高维大数据集更有效;但在构建 kd-树时,将所有数据点都组织在树结构中,需要大量的额外存储.文献[5]利用统一投影的方法加速搜索过程,并且提出了理论上严格证明有效的投影轴的选取准则,以排除不必要的搜索空间.但上述方法只是利用数据点的几何信息加快聚类过程,并没有利用迭代过程中产生的任何信息,而随着迭代的不断进行聚类中心会趋于稳定,因此迭代过程包含了许多可以利用的信息.文献[6]根据数据点与对应聚类中心间的距离在

收稿日期: 2010-10-09; **修回日期:** 2010-12-28

基金项目: 国家自然科学基金项目(60672054);陕西省科学技术攻关项目(2008K04-01);德州仪器创新基金(2009W1201)

前后两次迭代中的变化,判断该数据点是否属于这个类;如果与聚类中心的距离相对前次迭代变小,则无需计算该数据点与其他中心的距离就直接判定该数据点属于这个类;然而对于距离变大的情况依旧存在冗余计算,同时算法可能降低最终的聚类性能.文献[7]认为在一定迭代次数后,大多数数据点已经收敛到其最终的聚类中心而不再发生改变,因此算法在迭代过程中,将聚类中心分作静态组和动态组,通过中心的替代信息拒绝那些不可能成为数据点最近邻中心的中心点;算法需要为每个数据点保存对应的静态组和动态组索引,对于大数据集需要大量的存储空间.针对K-Means聚类过程中存在大量冗余计算和在迭代的过程中聚类中心不断趋于稳定的特点,本文首先将快速搜索算法PDS和ENNS算法引入到K-Means算法中以减少迭代过程中冗余的距离计算和维数计算,然后在此基础上利用迭代过程已经获得的历史信息,提出了和优先搜索序列的概念以进一步提高收敛速度.

2 K-Means 算法

K-Means是一种被广泛使用的基于划分的聚类方法^[1].给定数据集 $X = \{X_i \mid X_i \in \mathbb{R}^D\}$,其中 $i = 1, 2, \dots, N$, D 表示数据的维数.将数据划分为 K 类: $(C_1, C_2, \dots, C_K)$,每类样本的数目为 $N_k (k = 1, 2, \dots, K)$,均值为 $(m_1, m_2, \dots, m_K)$,则

$$m_k = \frac{1}{N_k} \sum_{X_i \in C_k} X_i \quad (1)$$

K-Means聚类试图找出使平方误差函数值最小的划分,其目标函数为

$$J = \sum_{k=1}^K \sum_{i=1}^{N_k} \|X_i - m_k\|^2 \quad (2)$$

算法流程如下所示.

步骤1:初始化算法选取个初始聚类中心点;

步骤2:划分(搜索)阶段:计算每个数据点与聚类中心的距离,根据最小距离原则进行搜索,对数据点进行划分;

步骤3:更新阶段:根据数据点的划分结果更新聚类中心;

步骤4:循环步骤2和3,直到达到收敛条件;

最小距离是通过计算数据点与所有聚类中心的Euclidean距离来决定的.数据点 X_i 与聚类中心 C_j 的距离定义如下:

$$D(X_i, C_j) = \sum_{d=1}^D (X_{id} - C_{jd})^2, 1 \leq i \leq N, 1 \leq j \leq K \quad (3)$$

在划分阶段,对于任意的数据点 $X_i (i = 1, 2, \dots, N)$,必须计算它和每一个聚类中心的距离,比较后找到最近邻中心,故为每个数据点划分聚类空间需要 K 次距离计算,从式(3)可以看出每次距离计算需要 D 次乘法和 $2D-1$ 次加法,因此划分阶段的复杂度为 $O(N \cdot K \cdot D)$,对于高维大数据集这将导致极大的计算量,因此研究快速K-Means算法具有重要意义.

2.1 部分失真搜索算法 PDS

部分失真搜索(Partial Distortion Search, PDS)算法^[8]在搜索过程中添加一个提前退出条件,判断当前计算数据点与聚类中心的距离计算过程中已经累加的部分失真是否已经超过目前的最小失真,如果超出则终止距离计算.算法的具体过程描述如下:

假设当前失真 $d_{\min} = D(X, C_p), 1 \leq p \leq K$,部分失真记为

$$D_{qd}(X, C_q) = \sum_{i=1}^d (X_i - C_{qi})^2, 1 \leq q \leq K, 1 \leq d \leq D \quad (4)$$

若 $D_{qd}(X, C_q) \geq d_{\min}$,则一定有 $D_q(X, C_q) = \sum_{i=1}^D (X_i - C_{qi})^2 \geq d_{\min}$,那么 C_p 就不可能是数据点 X 的最近邻的聚类中心,就不再需要进行余下部分维数的计算.这样通过额外的 d 次比较运算可以减少 $(D-d)$ 维数计算,减小距离计算过程中的冗余^[9].PDS算法易实现并且不需要额外的存储空间,因此常被用于许多搜索算法的最后一步.

2.2 等均值最近邻搜索算法 ENNS

文献[10]提出了等均值最近邻搜索(Equal-average Nearest Neighbor Search, ENNS)算法. ENNS算法将搜索范围限制在等均值的超平面内以排除冗余的距离计算,减少搜索空间,其过程如下:假设当前失真 $d_{\min} = D(X, C_p), 1 \leq p \leq K$,对于聚类中心 C_p ,如果满足条件:

$$|m_X - m_{C_q}| \geq \sqrt{d_{\min}/D} \quad (5)$$

则一定有 $D_q(X, C_q) \geq d_{\min}$,那么不用计算 $D_q(X, C_q)$ 就可以判断 C_q 不是数据点 X 的最近邻的聚类中心,从而避免不必要的距离计算,这里 m_X, m_{C_q} 分别表示数据点 X 和聚类中心 C_q 的均值.在迭代过程中,数据点并不发生改变,因此 m_X 只需要计算一次;而聚类中心在不断的更新,需要在每次搜索阶段之前重新计算聚类中心的均值.对于一次距离计算,如果通过式(5)能够成功地排除其成为最近邻中心的可能,只需要1次乘法和3次加法计算量,

而传统的搜索需要 D 次乘法和 $2D-1$ 次加法, 因此对于高维数据 ENNS 算法的计算量减少是显著的.

3 本文算法

由于 PDS 和 ENNS 算法局限在划分阶段的搜索过程中, PDS 算法引入的提前退出条件能够减少距离计算过程中的冗余维数计算; ENNS 算法通过式(5), 利用数据点和聚类中心的均值之间的标量计算就可以排除当前搜索的聚类中心是否是该数据点的最近邻中心, 以减少划分阶段的冗余距离计算. 本文首先将 PDS 和 ENNS 算法直接运用到 K-Means 的划分阶段, 其中 PDS 算法应用到 ENNS 算法的最后一步, 用以排除 ENNS 无法排除的聚类中心.

引入 PDS, ENNS 算法后, 首先从聚类中心的第一个中心点开始搜索, 初始的 $d_{\min}$ 是计算数据点与第一个聚类中心的距离得到的. 而从式(4)和(5)不难看出, 在划分过程中, $d_{\min}$ 越快速取得的值越小, 那么在余下的搜索过程中, ENNS 算法将能够排除更多距离计算, PDS 算法也将能够减少更多的冗余维数计算. 因为对于所有的数据点, $d_{\min}$ 最终等于数据点与聚类中心的最小距离 $D(X, C_p)$, 所以在最理想的情况下, 如果划分过程能够从最近邻聚类中心 C_p 开始搜索从而使得初始的 $d_{\min}$ 等于 $D(X, C_p)$, 就能够减少最多的计算量. 虽然 C_p 须在搜索结束后才能完全确定, 但在一定迭代次数后, 大多数数据点已经收敛到其最终的聚类中心而不再发生改变. 图 1 给出了用 Lena 图生成 $K=256$ 的码书过程中, 数据点在第 I 次迭代对应的索引号在过去 $I-1$ 次迭代过程中出现过的比例随着迭代次数变化的曲线, 可以看出在第 7 次迭代后, 90% 数据点的当前索引在过去的迭代中出现过, 说明聚类中心趋于稳定或只有轻微的变化, 那么 $D(X, C_{p'})$ 与 $D(X, C_p)$ 就应该非常接近 (p' 表示过去出现过的所有索引, p 表示本次迭代的索引). 于是相对于其他中心, $C_{p'}$ 与数据点更接近, 更有可能成为最近邻中心, 因此优先搜索与这些中心的距离将会更快取得较小的值, 本文将利用这些中心构成优先搜索序列, 以进一步减少计算量.

本文保留每个数据点在迭代过程中的历史索引记录, 提出了构造优先搜索序列的方案. 如图 2 所示, 每个数据点 X_i 的历史索引信息保存在对应的 L_i 中, 初始的 L_i 为空; 图中虚线框表示所有的聚类中心 $\{C_i | i=1, 2, \dots, K\}$, 搜索过程就是利用最近邻原则为每个数据点寻找其最近邻聚类中心, 虚线箭

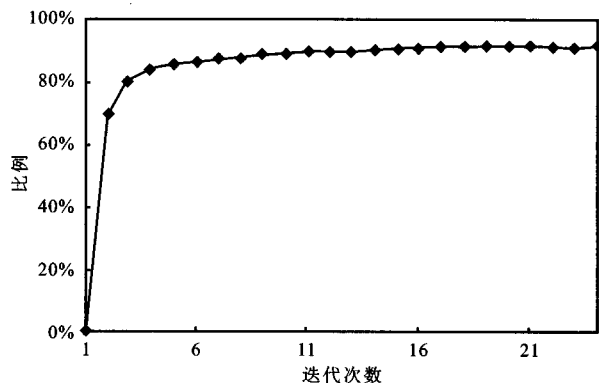


图 1 第 I 次迭代时索引在过去 $I-1$ 出现过的所占比例随迭代次数变化曲线 (Lena, $K=256$)

头表示 X_i 的上次迭代已获取的索引号 p , 即 $X_i \in C_p$, 实线对应当前迭代重新获取的索引 q . 图 2 中数据点 X_i 前后两次迭代的索引号分别为 p 和 q , 且 $p \neq q$, 即索引发生改变则将 q 加入到对应的优先搜索序列 L_i 的顶部 Top, 而数据点 X_j 前后两次迭代的索引号都是 r , 未发生改变, 故不添加 r 到 L_j 中. 下次搜索过程优先从 L_i 的 Top 元素开始以尽快得到较小的 $d_{\min}$, 然后搜索聚类中心除 L_i 以外的其余部分, 得到最终的最近邻中心. 例如, 如果 $K=6$, 在完成了 5 次迭代后某个数据点 X_i 的索引号依次是“1 2 2 2 3”, 则其保存的优先序列 L_i 为“3 2 1”, 可以看出优先序列不重复的保存着已进行的迭代过程中数据点所有索引号, 而且最新的索引号保存在 L_i 的顶部位置; 在进行第 6 次迭代的搜索过程中, 优先从 L_i 开始搜索以尽快获取较小的 $d_{\min}$ 值, 然后搜索聚类中心的余下部分, 即最终的搜索序列为“3 2 1 4 5 6”. 本文算法描述如下:

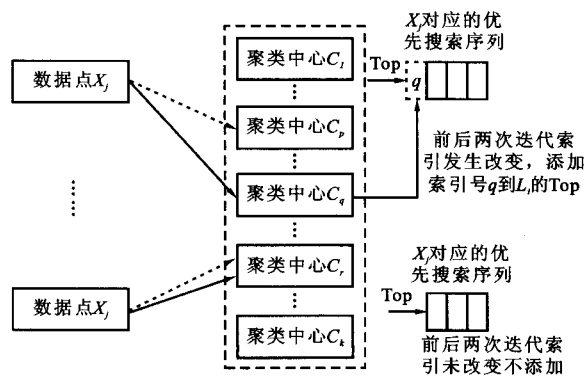


图 2 优先搜索序列构建示意图

步骤 1: 初始化算法选取 K 个初始聚类中心点, 计算数据点均值 m_X , 并为每个数据点 X_i 初始化空的优先搜索序列 L_i ;

步骤 2: 循环步骤 3 至步骤 5, 直到达到收敛条件;

步骤 3: 计算聚类中心的均值 $\overline{C_p}$;

步骤 4: 划分(搜索)阶段, 对每个数据点 X_i 执行下面步骤;

如果 L_i 不为空, 利用 ENNS 和 PDS 算法计算 X_i 与集合 $\{C_p, p \in L_i\}$ 的最小距离 $d_{\min}$, 否则 $d_{\min}$ 令为无穷大, 跳至下一步;

利用 ENNS 和 PDS 算法计算 X_i 与 $\{C_q | q \notin L_i, q = 1, 2, \dots, K\}$ 的距离得到最小的 $d_{\min}$ 以及对应的聚类中心索引号 q ;

如果前后两次迭代索引号发生变化, 将 p 加入到 L_i 顶端;

步骤 5: 更新阶段: 根据划分结果更新聚类中心.

4 实验

为验证本文算法的性能, 我们采用了 4 组不同的数据集. 第一组数据集包含约 100 000 个数据点, 来自标准灰度 Lena 图(图 3), 数据点由图像块构成. 第二组数据来自 UCI^[11], 包含 Iris、Wine 和 Letter Recognition 等, 表 1 给出了数据集的具体描述. 第三组数据包含 40 个均匀分布在 $[-1, 1]^D$ 空间的聚类中心, 数据维数从 4 到 40 的高斯分布数据集^[7]. 第四组包含了几组利用高斯-马尔可夫定理 (Gauss Markov Sequence) 生成的包含 10 000 数据点的数据集, 其中数据维数 D 分布从 8 到 40, 序列均值 $\mu=0$, 序列标准差 $\sigma=10$, 相关系数 $\alpha=0.9$.

本文通过以上数据集比较了传统的全搜索 K-Means(FS), 引入 ENNS 算法(PDS 算法在 ENNS 算法的最后一步)的 K-Means (ENNS), 以及本文提出的优先搜索序列算法 (Priority Search List,

PSL) 3 种算法的性能. 所有测试在 Intel T2130 主频 2.8 GHz, 内存 1024 MB 的个人计算机上进行. 对每种算法使用相同的 50 组随机选取的初始聚类中心, 统计每次迭代的平均计算时间, 每个数据点平均距离计算次数以及每次距离计算中的平均维数计算次数.

表 1 实验采用的 4 组来自 UCI 的真实数据的具体描述

数据集	数据维数 D	类中心数目 K	数据点总数 N
Iris	4	3	150
Wine	13	3	178
Letter Recognition	16	26	20 000



图 3 Lena 图

表 2 列出了第一组数据集的实验结果, 3 种不同算法的 PSNR 性能参数并未发生改变, 这是因为引入 ENNS 算法和改进的 ENNS 算法并不改变聚类结果. 全搜索的 K-Means 的计算量随着聚类中心数目增加呈线性递增; 引入 ENNS 算法后, 能够使计算量降至 7%~15%, 单次迭代时间降至 9.6%~21.0%. 而相对于 ENNS 算法, 本文的 PSL 算法性能有显著提升, 距离计算次数相对减少了 23.6%~43.3%, 单次迭代时间减少了 10.3%~30.6%.

表 2 第 1 组数据集实验结果 (PSNR 为峰值信噪比)

算法	K=64				K=128				K=256				K=512				K=1024			
	PSNR	时间	距离	维数	PSNR	时间	距离	维数	PSNR	时间	距离	维数	PSNR	时间	距离	维数	PSNR	时间	距离	维数
FS	29.52	146.3	64	16.0	30.60	288.4	128	16.0	31.54	572.8	256	16.00	32.42	1 138.9	512	16.00	33.32	2 255.6	1 024.0	16.0
ENNS	29.52	30.7	9.7	11.5	30.60	46.5	15.5	10.0	31.54	72.5	24.8	8.6	32.42	122.4	42.1	7.6	33.32	217.5	73.6	6.8
PSL	29.52	21.3	5.5	8.2	30.60	34.3	9.6	6.8	31.54	59.0	17.0	5.9	32.42	106.1	31.3	5.3	33.32	194.9	56.2	4.7

表 3 第 2 组 UCI 数据集实验结果

Method	Ltrrec			Wine			Iris		
	时间/ms	距离	维数	时间/ms	距离	维数	时间/ms	距离	维数
FS	79.10	26.00	16.00	0.08	3.00	13.00	0.02	3.00	4.00
ENNS	57.05	21.17	9.19	0.08	2.96	10.22	0.03	2.13	3.94
FSL	47.03	18.81	7.23	0.08	2.94	9.08	0.02	1.32	3.87

表 3 给出了第二组 UCI 数据集的实验结果,在 Ltrrec 数据集上算法的性能提升效果最为显著,对于 Wine, Iris 数据集,在维数计算、距离计算量都有提升,但是由于额外的提前退去的判断以及 ENNS 算法每次迭代需要重新计算聚类中心的均值等额外开销使得性能提升在时间方面的改善相对较小。

图 4、图 5 和图 6 给出在第三组数据集的平均迭代时间,每个数据点的平均距离计算次数和平均每次距离计算中的计算维数。从图 4 可以看出通过引入优先搜索的 PSL 原则能够显著的减少迭代所需时间。一方面,如图 5 所示,在 FS 算法中,每个数据点的平均距离计算次数等于聚类中心数目,ENNS 算法排除冗余距离计算的效果很明显,同时本文算法相对 ENNS 有一个相对恒定的距离次数减少;另一方面,如图 6 所示,优先搜索 PSL 的方法可以使 PDS 算法中的维数计算进一步改善。可见,本文算法通过引入 PSL 以便尽快得到较小的思路,能够进一步明显减少冗余距离计算次数和维数计算。

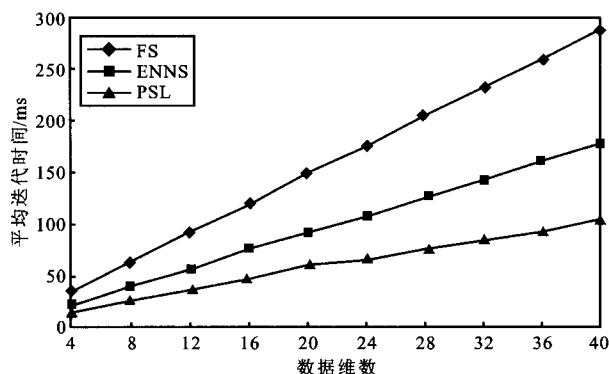


图 4 在第 3 组数据集,平均迭代时间随数据维数的变化,其中 $N=20\,000$, $K=40$

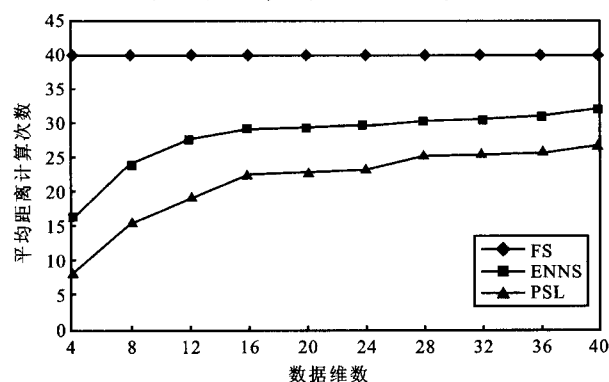


图 5 在第 3 组数据集,每个数据点的平均距离计算次数随数据维数的变化,其中 $N=20\,000$, $K=40$

图 7、图 8 和图 9 给出了 3 种算法在第四组数据集上的实验结果。从图 8 的每个数据点的平均距离计算次数可以看出在数据维数接近到 16 时,

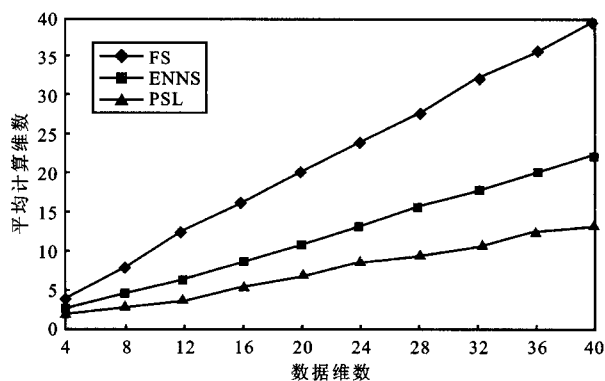


图 6 在第 3 组数据集,每次距离计算的平均计算维数随数据维数的变化,其中 $N=20\,000$, $K=40$

ENNS 的距离计算次数趋近于聚类中心数目,说明 ENNS 算法的距离计算接近于全搜索的 K -Means 算法;然而在图 9 中,ENNS 算法的每次距离计算的计算维数约降至全搜索的 60%,在此基础上本文的优先搜索 PSL 算法能够进一步降低约 15%;从图 7 的时间性能体现了冗余的计算维数的减小,ENNS 算法在迭代平均时间仍有显著改进,减少了 13%~32%;相对于 ENNS,本文算法进一步降低了 11%~17%。

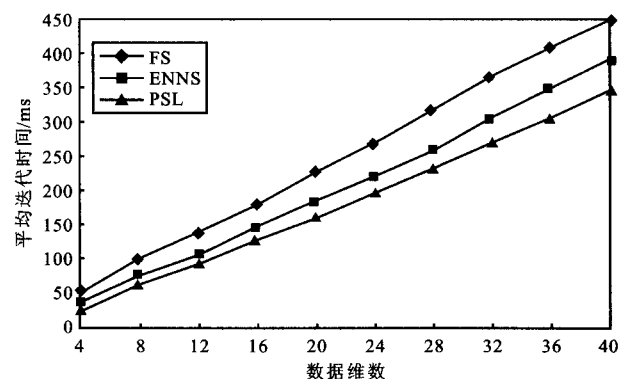


图 7 在第 4 组数据集,平均迭代时间随数据维数的变化,其中 $N=10\,000$, $K=128$

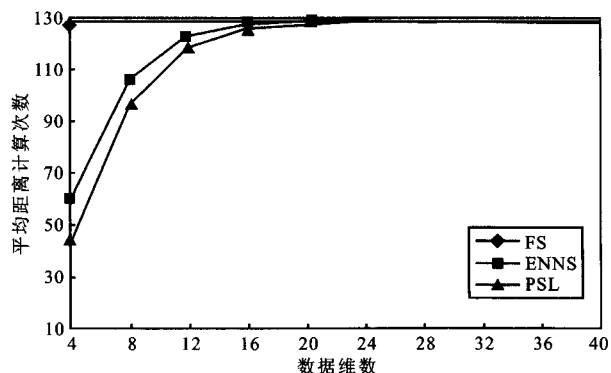


图 8 在第 4 组数据集,每个数据点的平均距离计算次数随数据维数的变化,其中 $N=10\,000$, $K=128$

5 结束语

本文首先将快速搜索算法PDS,ENNS集成到K-Means聚类的划分阶段,ENNS算法通过限制搜索空间而减少大量冗余的距离计算,而且通过在最后一步使用PDS算法能够有效减少距离计算中的冗余维数计算;在此基础上,本文通过利用迭代过程中已经获取的索引历史信息为数据点构造一个优先搜索序列,以尽快得到较小的值,从而进一步明显减少了冗余距离计算次数和维数计算,提高了聚类的时间性能。本文算法易于实现,在不改变K-Means聚类的输出结果的前提下,对4种不同类型的数据集能够在计算量和迭代时间上取得进一步明显的性能提升。

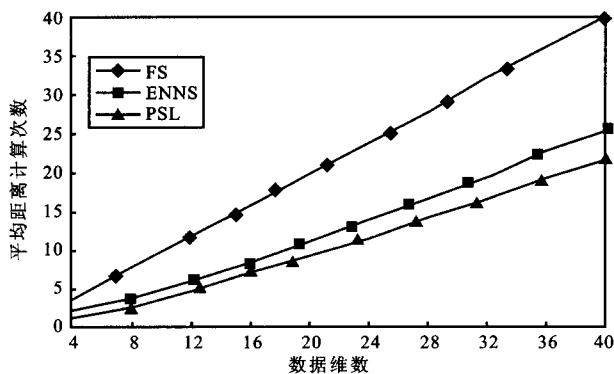


图9 在第4组数据集,每次距离计算的平均计算维数随数据维数的变化,其中 $N=10\,000$, $K=128$

参考文献:

- [1] Berkhin P. A survey of clustering data mining techniques[J]. Grouping Multidimensional Data, 2006: 25-71.
- [2] Elkan C. Using the triangle inequality to accelerate k-means[C]// Proceedings of the Twentieth International Conference on Machine Learning (ICML'03). Washington, DC: ICML, 2003: 147-153.
- [3] Kanungo T, Mount D M, Netanyahu N, et al. An efficient k-means clustering algorithm: Analysis and implementation [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2002, 24(7): 881-892.
- [4] Pelleg D, Moore A. Accelerating exact k-means algorithms with geometric reasoning[C]// Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. NY, USA: ACM, 1999: 277-281.
- [5] Pan Z, Kotani K, Ohmi T. A unified projection method for fast search of vector quantization [J]. IEEE Signal Processing Letters, 2004, 11(7): 637-640.
- [6] Pan JS, McInnes FR, Jack MA. Fast clustering algorithms for vector quantization[J]. Pattern Recognition, 1996, 29(3): 511-518.
- [7] Lai J Z C, Huang T J, Liaw Y C. A fast k-means clustering algorithm using cluster center displacement [J]. Pattern Recognition, 2009, 42(11): 2551-2556.
- [8] Bei C D, Gray R M. An improvement of the minimum distortion encoding algorithm for vector quantization [J]. IEEE Transactions on Communications, 1985, 33(10): 1132-1133.
- [9] 孙圣和, 陆哲明. 矢量量化技术及应用[M]. 北京: 科学出版社, 2002.
- [10] Ra S W, Kim J K. A fast mean-distance-ordered partial codebook search algorithm for image vector quantization [J]. IEEE Transactions on Circuits and Systems II—Analog and Digital Signal Processing, 1993, 40(9): 576-579.
- [11] Asuncion A, Newman D. UCI machine learning repository[DB/OL]. [2010-03-13]. [http://www.ics.uci.edu/~mllearn/MLRepository.html]. Irvine, CA: University of California, School of Information and Computer Science, 2007.

作者简介:

禹贵辉 男,(1985—),硕士研究生。研究方向为图像识别。
 潘志斌 男,(1963—),教授。研究方向为图像编码、图像识别。
 乔瑞萍 女,(1966—),副教授。研究方向为图像编码算法及其DSP实现。
 邹彬 男,(1986—),硕士研究生。研究方向为图像识别。
 姜彦民 男,(1976—),硕士研究生。研究方向为快速图像编码的DSP实现。