

A linear time deterministic algorithm to find a small subset that approximates the centroid [☆]

Pratik Worah ^a, Sandeep Sen ^{b,*}

^a Department of Computer Science, University of Illinois, Urbana Champaign, IL 61801, USA

^b Department of C.S.E., I.I.T. Delhi, New Delhi 110016, India

Received 18 February 2006; received in revised form 6 June 2007; accepted 19 July 2007

Available online 2 August 2007

Communicated by S.E. Hambrusch

Abstract

Given a set of points S in any dimension, we describe a deterministic algorithm for finding a $T \subset S$, $|T| = O(1/\varepsilon)$ such that the centroid of T approximates the centroid of S within a factor $1 + \varepsilon$ for any fixed $\varepsilon > 0$. We achieve this in linear time by an efficient derandomization of the algorithm in [M. Inaba, N. Katoh, H. Imai, Applications of weighted Voronoi diagrams and randomization to variance-based k -clustering (extended abstract), in: Proceedings of the Tenth Annual Symposium on Computational Geometry, 1994, pp. 332–339].

© 2007 Elsevier B.V. All rights reserved.

Keywords: Computational geometry; Clustering; Derandomization

1. Introduction

Given a point set S in \mathcal{R}^d , let $c(S)$ denote the centroid of S that is defined as $c(S) = \sum_{s_i \in S} s_i / |S|$.

Let

$$\text{Var}(S) = \sum_{s_i \in S} \|s_i - c(S)\|^2 / |S|$$

denote the variance of the point set S where $\|\mathbf{x}\|^2 = \mathbf{x} \cdot \mathbf{x}$ for a vector \mathbf{x} .

Note that throughout the paper the multiplication and addition operations on the points in the d -dimensional

points represent vector addition and dot product, respectively. We do not use any special notation for these operations.

So, (optimum) centroid of $S = |S| \cdot \text{Var}(S)$ and the 1-center of S is $c(S)$. The main result in this paper can be stated as follows

Theorem 1. Given a set S of n points in \mathcal{R}^d and an $\varepsilon > 0$, we can find a multiset $T \subset S$ of $O(1/\varepsilon)$ points (deterministically) in time $O(n/\varepsilon)$, such that

$$\sum_{s_i \in S} \|s_i - c(T)\|^2 / |S| \leq (1 + \varepsilon) \cdot \text{Var}(S).$$

2. Preliminaries and background

Our algorithm is based on the following result that has recently found interesting applications in [1].

[☆] Part of the work done when the authors were in Dept. of CSE, IIT Kharagpur, India.

* Corresponding author.

E-mail addresses: pworah2@uiuc.edu (P. Worah), ssen@cse.iitd.ernet.in (S. Sen).

Lemma 1. (See [2].) Let $T \subset S$ be a uniformly chosen random subset with $|T| = m$ (sampled with replacement and hence T is a multiset). Then

$$E[c(T) - c(S)] = 0 \quad \text{and}$$

$$E[\|c(T) - c(S)\|^2] = \text{Var}(S)/m,$$

where $E[\cdot]$ denotes the expectation over all choices of T .

Using Markov's inequality in conjunction with the second equality in the previous Lemma, we obtain

$$\Pr(\|c(T) - c(S)\|^2 > \text{Var}(S)/(\delta m)) < \delta.$$

Or in other words, there exists a subset T satisfying the condition $\|c(T) - c(S)\|^2 \leq \text{Var}(S)/m$ (using $\delta = 1$ in the previous inequality). Since for any point t , $\sum_{s_i \in S} \|s_i - t\|^2 / |S| = \sum_{s_i \in S} \|s_i - c(S)\|^2 / |S| + \|c(S) - t\|^2$, there exists some subset T satisfying Theorem 1 with $m = 1/\varepsilon$. In the next section we will design an efficient deterministic algorithm to find such a T .¹

3. Efficient derandomization

The derandomization of the algorithm in [2] is done by the method of conditional probabilities [2]. The trivial method of derandomization based on checking all m combinations leads to an $O(n^{O(m)})$ time algorithm that is not practical even for $\varepsilon = 0.1$ (i.e. $m = 10$). Our goal is to design a linear (in n) time algorithm when m is $O(1)$.

Let X_1, X_2, \dots, X_m be random variables which take the values from the point set S . A direct application of the Raghavan–Spencer method of conditional probabilities (see, for example, [4]) implies the following procedure:

Algorithm Approx-Centroid

Input: A point set S of size n .

Output: A point set $T \subset S$ of size m ($m < n$) satisfying $\|c(T) - c(S)\|^2 \leq \text{Var}(S)/m$.

Notation: T^i is a vector $[x_1, \dots, x_i, X_{i+1} \dots X_m]$ where $x_i \in S$ and X_j $i < j \leq m$ are random variables. Initially $T^0 = [X_1 \dots X_m]$ and finally $T^m = T = [x_1, \dots, x_m]$.

The expectation of T^i is with respect to the random variables $X_{i+1} \dots X_m$.

for each random variable X_i $1 \leq i \leq m$,

{
find $x_i \in S$ such that

$$E[\|c(T^i) - c(S)\|^2 \mid X_1 = x_1, X_2 = x_2, \dots, X_i = x_i] \\ \leq \text{Var}(S)/m$$

(given $E[\|c(T^{i-1}) - c(S)\|^2 \mid X_1 = x_1, X_2 = x_2, \dots, X_{i-1} = x_{i-1}] \leq \text{Var}(S)/m$)

$X_i = x_i$
}

Lemma 2. We can always find $x_i \in S$, i.e., an assignment of the random variable X_i in the i th iteration which satisfies $E[\|c(T^i) - c(S)\|^2 \mid X_1 = x_1, X_2 = x_2, \dots, X_i = x_i] \leq \text{Var}(S)/m$.

Proof. The proof of the Lemma follows from an inductive argument on the number of random variables fixed. The base case, i.e., the value of the expectation with no random variables fixed is less than $\text{Var}(S)/m$. The formula for the conditional expectation after the $(i-1)$ th round (i.e., the $i-1$ random variables X_1, \dots, X_{i-1} are fixed to values x_1, \dots, x_{i-1} , respectively) can be written as

$$E[\|c(T^i) - c(S)\|^2 \mid X_1, \dots, X_{i-1}] \\ = E \left[\left\| (X_1 + \dots + X_m) \frac{1}{m} - c(S) \right\|^2 \mid X_1, \dots, X_{i-1} \right] \\ = E \left[(X_1 + \dots + X_m)^2 \frac{1}{m^2} - 2c(S)(X_1 + \dots + X_m) \frac{1}{m} + c(S)^2 \mid X_1, \dots, X_{i-1} \right] \\ = (x_1^2 \dots + x_{i-1}^2 + 2x_1x_2 \dots + 2x_{i-2}x_{i-1} + E[X_i^2] \\ + \dots + E[X_m^2] + 2x_1E[X_i] \dots + 2x_{i-1}E[X_m] \\ + 2E[X_iX_{i+1}] \dots 2E[X_{m-1}X_m]) \frac{1}{m^2} \\ - (2c(S)x_1 + \dots 2c(S)x_{i-1} + 2c(S)E[X_i] \\ + 2c(S)E[X_m]) \frac{1}{m} + c(S)^2. \quad (1)$$

This quantity is less than $\text{Var}(S)/m$ by the induction hypothesis. We note that the random variables X_j, X_{j+1}, \dots, X_m are independent for all $i < j \leq m$. Hence the terms not involving X_i can be considered as constants for the i th iteration and we can rewrite the above expression in the following simplified form

$$E[\|c(T^i) - c(S)\|^2 \mid X_1, \dots, X_{i-1}] \\ = C_i + \frac{1}{m^2} E[X_i^2 + D_i X_i], \quad (2)$$

where

¹ A randomized algorithm is simply a uniform random sample of size m/δ where $1 - \delta$ is the success probability of the algorithm.

$$C_i = \frac{1}{m^2} \left(\sum_{1 \leq j \leq i-1} x_j + \sum_{i+1 \leq j \leq m} E[X_j] \right)^2 - \frac{2c(S)}{m} \left(\sum_{1 \leq j \leq i-1} x_j + \sum_{i+1 \leq j \leq m} E[X_j] \right) + c(S)^2,$$

$$D_i = 2 \left(\sum_{1 \leq j \leq i-1} x_j + \sum_{i+1 \leq j \leq m} E[X_j] - mc(S) \right).$$

Note that C_i , D_i are values dependent only on S , m and i and hence can be evaluated separately from X_i . Note also that for random variables X , Y and a function $f(X, Y)$, $E[f(X, Y) | Y] \geq f(x_o, Y)$ for some x_o from the definition of conditional expectation. Hence there exists at least one value of X_i (say x_i) such that expression (2) above is at most $\text{Var}(S)/m$ when $X_i = x_i$. This ‘good’ value of X_i can be found by an exhaustive search in S . In other words we evaluate expression (2) by replacing X_i with the coordinates of each point $x_j \in S$ for all $1 \leq j \leq n$. Since $E[\|c(T^i) - c(S)\|^2 | X_1, \dots, X_{i-1}] \leq \text{Var}(S)/m$ (induction hypothesis) the argument above implies there exists some point x_i such that expression (2) evaluates to at most $\text{Var}(S)/m$.

Hence the value of the expectation with the i random variables X_1, \dots, X_i fixed to x_1, \dots, x_i , respectively, is less than $\text{Var}(S)/m$. This completes the induction proof.

Time complexity: In each of the m iterations we may have to examine n potential values for X_i before we find a ‘good’ value. Every time we try out a different value for X_i we re-evaluate Eq. (2) for computing the conditional expectation. The evaluation of $E[X_j]$, $E[X_j^2]$ ($j \neq i$) takes $O(n)$ time. This implies an $O(n^2)$ running time for the algorithm. However the following argument allows the formula for conditional expectation in Eq. (2) to be evaluated in constant time in each iteration (except the first iteration in which $O(n)$ time is required) giving us a linear time algorithm.

Since we do independent sampling with replacement, the values for the expectation terms like $E[X_j^2]$, $E[X_j]$ remain unchanged through all the iterations and

$$E[X_1^2] = E[X_2^2] = \dots = E[X_m^2],$$

$$E[X_1] = E[X_2] = \dots = E[X_m].$$

Therefore we can precompute the values of $E[X_1]$, $E[X_1^2]$ initially in $O(n)$ time and reuse these values in each iteration when re-evaluating the formula. By inspecting Eq. (2), and the previous observations, the simplified formula for the i th iteration can be computed in $O(1)$ steps by keeping track of partial sums. The follow-

ing relations hold for the various constants in Eq. (2) between the i th and $(i-1)$ th iterations.

$$C_i = C_{i-1} + (x_{i-1}^2 - E[X_i^2])/m^2 + 2x_{i-1}B_{i-1}/m^2 - 2E[X_i](B_{i-1} - E[X_i] + x_{i-1})/m^2 - 2c(S)(x_{i-1} - E[X_i])/m, \quad (3)$$

$$B_{i-1} = (x_1 + \dots + x_{i-2} + E[X_i] + \dots + E[X_m]),$$

$$B_i = B_{i-1} - E[X_i] + x_{i-1}, \quad (4)$$

$$D_i = D_{i-1} + 2x_{i-1} - 2E[X_i]. \quad (5)$$

In the i th iteration one can find B_i , C_i , D_i by using the precalculated values of $c(S)$, x_{i-1} , $E[X_i]$, $E[X_i^2]$ and Eqs. (3)–(5) in $O(1)$ time. Hence the formula for the conditional expectation can be updated in $O(1)$ time across successive iterations. This yields a total running time of $m(O(n) + O(1))$, i.e., $O(mn)$ steps.

After m iterations, the conditional expectation will be the unconditional value of $\|c(T) - S\|^2$, so the result will hold deterministically. The values chosen for X_i s form the multiset $T \subset S$, satisfying $\|c(T) - c(S)\|^2 \leq \text{Var}(S)/m$ that completes the proof of Theorem 1. \square

4. Applications and remarks

The k -means (k -centroid) algorithm of Kumar et al. [1] makes crucial use of the result of [2] to approximate the centroid from a small random sample. It also uses other random sampling techniques to achieve a PTAS for the k -means problem that grows linearly with n and dimensions for a fixed k . Derandomizing the algorithm is a challenging exercise and the result in this paper achieves some progress in this direction.

There has been considerable interest in *semi-streaming* algorithms that makes a bounded number of passes through the input data (Feigenbaum et al. [3]). In this context, our algorithm makes $O(1/\epsilon)$ passes over the data and uses an additional $O(1/\epsilon)$ space.

References

- [1] A. Kumar, Y. Sabharwal, S. Sen, A simple linear time $(1 + \epsilon)$ -approximation algorithm for k -means clustering in any dimensions, in: Proceedings of the 45th Annual Symposium on Foundations of Computer Science, 2004, pp. 454–462.
- [2] M. Inaba, N. Katoh, H. Imai, Applications of weighted Voronoi diagrams and randomization to variance-based k -clustering (extended abstract), in: Proceedings of the Tenth Annual Symposium on Computational Geometry, 1994, pp. 332–339.
- [3] J. Feigenbaum, S. Kannan, McGregor, S. Suri, J. Zhang, On graph problems in a semi-streaming problem, Theoretical Computer Science 348 (2) (2005).
- [4] R. Motwani, P. Raghavan, Randomized Algorithms, Cambridge University Press, 2000.