Fitting Flats to Points with Outliers

Guilherme D. da Fonseca*

Abstract

Determining the best shape to fit a set of points is a fundamental problem in many areas of computer science. We present an algorithm to approximate the k-flat that best fits a set of n points with $n-m$ outliers. This problem generalizes the smallest m-enclosing ball, infinite cylinder, and slab. Our algorithm gives an arbitrary constant factor approximation in $O(n^{k+2}/m)$ time, regardless of the dimension of the point set. For many practical sets of inliers, the running time is reduced to $O(n^{k+2}/m^{k+1})$, which is linear when $m = \Omega(n)$.

1 Introduction

Determining the best shape to fit a set of points is a fundamental problem in statistics, machine learning, data mining, computer vision, clustering, and pattern recognition. The case of fitting a lower-dimensional space deserves special attention since it can be used to minimize the effects of the curse of dimensionality. A widely used measure of how well a shape S fits a set P of n points in d-dimensional space is $\max_{p \in P} \min_{s \in S} \|ps\|$, the maximum Euclidean distance between any point $p \in P$ and the shape S. Unfortunately, this measure is very sensitive to the presence of outliers.

A more robust measure in the presence of $n-m$ outliers and m inliers consists of minimizing the following cost function: given a parameter $m \leq n$, the cost is the m-th smallest distance between a point in P and the shape S. In this paper, we consider an approximation to the case when S is a k-dimensional flat, for a given value of $k \in \{0, \ldots, d-1\}$. We show that, for an arbitrary $\varepsilon > 0$, we can find in $O_\varepsilon(n^{k+2}/m)$ time[1], with constant probability, a k-dimensional flat S with cost at most $1+\varepsilon$ times the optimum. We refer to this problem as *flat fitting*. We assume that the dimensions k, d are constants, but $1/\varepsilon$ is an asymptotic quantity. It is noteworthy that the complexity depends only on the target dimension k, regardless of the dimension of the point set. Our algorithm is Monte Carlo, but can be made deterministic at the expense of an $O(m)$ factor in the running time.

In the most interesting case when m is a constant fraction of n, the running time of our Monte Carlo algorithm is $O_\varepsilon(n^{k+1})$. While the running time is close to the $\Omega(n^k)$ lower bound, the algorithm is still super-linear for $k \geq 1$. We show that when the set of inliers satisfies some density criterion, the running time is reduced to $O(n^{k+2}/m^{k+1})$, which is linear for $m = \Omega(n)$. This way, we show that despite the high worst-case complexity of the problem, there is a feasible solution for some practical large data sets.

Related work. The case of $k = 0$ corresponds to the well-studied problem of approximating the smallest ball enclosing m points. The problem can be solved in $O(n/\varepsilon^{d-1})$ expected time by using techniques from [2, 5, 9]. An easier variation of this problem, when an inlier is known, is used as a base case for our algorithm.

The case of $k = d-1$ corresponds to approximating the narrowest slab enclosing m points. In contrast to the linear complexity of the $k = 0$ case, the most efficient solution for $k = d-1$ is a high probability Monte Carlo algorithm [6] with running time $O(n^d(\log^{O(1)} \frac{1}{\varepsilon})/m\varepsilon)$. Major improvements are unlikely, since there is a lower bound of $\Omega((n-m)^{d-1} + (n/m)^d)$ for obtaining a constant approximation [6].

The case of $k = 1$ corresponds to approximating the smallest infinite cylinder enclosing m points, which is stated as an open problem by Har-Peled and Mazumdar [9]. A linear time solution for arbitrary values of m is unlikely, since even the planar approximation problem is 3SUM-hard [8]. To see that, note that it is 3SUM-hard to decide if there are three points on a line and that there is a planar cylinder of radius 0 enclosing three points if and only if there are three points on a line.

When the number $n-m$ of outliers is small compared to n, we can use the coreset framework to reduce the number of points to $O((n-m)/\varepsilon^{(d-1)/2})$ and then solve the problem in the reduced point set [1]. The case when d is an asymptotic variable is considered in [10], where an algorithm linear in d but exponential in $1/\varepsilon$ is presented. Approaches based on random sampling such as RANSAC [7] are widely used in practice, but do not guarantee approximation with respect to the optimum.

The non-robust version of the problem (when $m = n$) is generally solved using coresets [4]. The case when d is an asymptotic variable is considered in [11].

*Universidadade Federal do Estado do Rio de Janeiro (UNIRIO), Brazil, fonseca@uniriotec.br

[1]We use the $O_\varepsilon(\cdot)$ notation to hide polynomial ε-dependencies.

When $k = 0$, it is well known that the non-robust exact version can be solved in $O(n)$ time. Exact solutions for other values of k are considerably less efficient, even in the non-robust version. Chan [3] mentions an $O(n^{\lceil d/2 \rceil})$ algorithm for $k = d-1$ and an $O(n^{2d-1+\delta})$ algorithm for $k = 1$, where δ is an arbitrarily small constant.

The exact robust version seems even harder. A trivial solution takes $O(n^{(d-k)(k+1)+2})$ time, by counting the number of points for each potential set of up to $(d-k)(k+1)+1$ farthest inliers. When $k = d-1$, the problem can be solved in $O(n^d)$ expected time [6], improving the trivial solution by two $O(n)$ factors, one by efficiently counting the number of points using arrangements, and one by using Chan's randomized optimization [2].

A lower bound of $\Omega((n-m)^{d-1} + (n/m)^d)$ for obtaining a constant approximation when $k = d-1$ in presented in [6]. The lower bound is based on a conjecture for the complexity of the affine degeneracy problem. We can linearly reduce the flat fitting problem with $k = d-1$ to the flat fitting problem in higher dimension $d' \geq d$ and the same value of k. Therefore, the lower bound for $k = d-1$ implies a lower bound of $\Omega((n-m)^k + (n/m)^{k+1})$ for arbitrary k. In the most interesting case when m is a constant fraction of n, the lower bound is $\Omega(n^k)$ and we present an upper bound of $O(n^{k+1})$.

Next, we present approximate algorithms for the flat fitting problem. We present a Monte Carlo algorithm with running time $O_\varepsilon(n^{k+2}/m)$ and a deterministic algorithm with running time $O_\varepsilon(n^{k+2})$. In Section 3, we show how to reduce the running time of the Monte Carlo algorithm to $O_\varepsilon(n^{k+2}/m^{k+1})$ for some typical sets of inliers. Concluding remarks and open problems are discussed in Section 4.

2 Approximate Algorithm

The general idea of the algorithm consists of finding a vector v that is approximately parallel to the best fitting flat and then projecting the points onto a hyperplane perpendicular to v and recursively solving a lower dimensional problem. We use $k = 0$ as a base case. Actually, the algorithm computes a somewhat small set of vectors that contains v and recurses for each vector in the set, returning the best solution found. We start by providing some definitions.

Let $S_{k,d}(P)$ and $c_{k,d}(P)$ respectively denote the optimal k-dimensional flat for point set P in d-dimensional space and its cost. We refer to the m points $P' \subseteq P$ within distance $c_{k,d}(P)$ of $S_{k,d}(P)$ as *inliers*. Given a d-dimensional set of points P and a vector v, let $P_{|v}$ denote a $(d-1)$-dimensional point set obtained by projecting P onto a hyperplane perpendicular to v. Given a vector v let v' be the unit length projection of v onto the optimal flat $S_{k,d}(P)$,

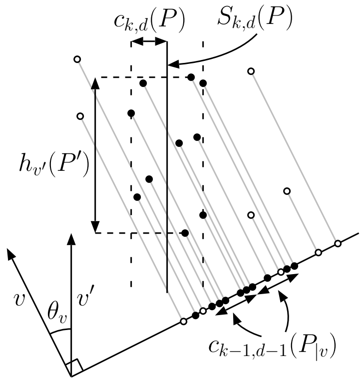

Figure 1: Definitions used to state Lemma 1. The $m = 10$ inliers are represented by solid circles.

$h_{v'}(P') = \max_{p \in P'} v' \cdot p - \min_{p \in P'} v' \cdot p$ be the directional width in direction v' of the inliers, and θ_v be the acute angle between v and v'. See Figure 1 for a diagram of the previous definitions. The following lemma follows from simple trigonometric arguments and shows how to use the solution of a lower dimensional problem in order to approximate the original problem.

Lemma 1 *For any vector v we have*

$$c_{k,d}(P) \leq c_{k-1,d-1}(P_{|v}) \leq c_{k,d}(P) + h_{v'}(P')\theta_v.$$

By Lemma 1, it is possible to obtain a $(1+\varepsilon)$-approximation by finding a vector v with angle

$$\theta_v \leq \frac{\varepsilon\, c_{k,d}(P)}{d\, h_{v'}(P')} = \phi$$

and recursively solving the lower dimensional problem. Our algorithm considers a set of vectors that contains a vector u with $\theta_u < \phi$, returning the solution of minimum cost found. The following lemma is the key to obtain such set.

Lemma 2 *For every inlier $p \in P'$, there is an inlier $q \in P'$ such that the vector $v = q - p$ has*

$$\theta_v \leq \frac{4c_{k,d}(P)}{h_{v'}(P')} \text{ and}$$

$$\frac{h_{v'}(P')}{2} \leq \|v\| \leq 2c_{k,d}(P) + h_{v'}(P').$$

Proof. (sketch) Consider the inlier $q \in P'$ that realizes the maximum directional distance $\max_{q \in P'} |v' \cdot p - v' \cdot q|$ and use simple geometric arguments (see Figure 2). □

Say we have a vector v satisfying the properties of Lemma 2. If $c_{k,d}(P) \geq h_{v'}(P')$, then we obtain a set of size $O(1/\varepsilon^{d-k})$ containing a vector u with $\theta_u \leq \phi$ in the following manner. The intersection

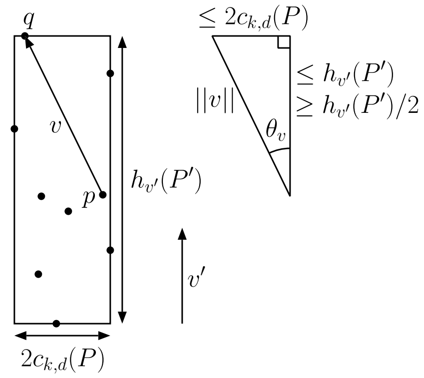

Figure 2: Proof of Lemma 2.

of a $(d-k+1)$-flat F in general position and the optimal flat $S_{k,d}(P)$ is a line ℓ. Using a standard grid of directions, we create a set of $O(1/\varepsilon^{d-k})$ vectors in F that contain a vector u within angle at most ε/d of ℓ, and consequently has $\theta_u \leq \phi$. Next, we focus on the more interesting case when $c_{k,d}(P) < h_{v'}(P')$.

By Lemma 2, we have that $\|v\|$ is a constant factor approximation of $h_{v'}(P')$. By Lemma 1, we can recursively solve the $(d-1)$-dimensional problem with point set $P_{|v}$ in order to obtain a constant factor approximation to $c_{k,d}(P)$. Putting both approximations together, we obtain a constant factor upper bound to θ_v. We use the approximation of θ_v to obtain a set of size $O(1/\varepsilon^{d-k})$ containing a vector u with $\theta_u \leq \phi$. The set is defined by a grid of directions in a $(d-k+1)$-flat as before, but noting that the angle between v and u is upper bounded by the approximation of θ_v.

Now, assume we know an inlier $p \in P'$. By Lemma 2, the set $V = \{p - q : q \in P\}$ of size $O(n)$ contains a vector satisfying the condition of Lemma 2. Therefore, we can obtain a set U of size $O(n/\varepsilon^{d-k})$ that contains a vector u with $\theta_u \leq \phi$.

For each vector $u \in U$, we project the points onto a hyperplane perpendicular to u and recursively solve the lower dimensional problem. Next, we discuss how to solve the base case $k = 0$, given an inlier p. The base case consists of approximating the smallest ball enclosing m points, including the inlier p.

Using techniques from [5, 9], the base case problem can be solved in time $O(n+m(\log \frac{1}{\varepsilon})/\varepsilon^{d-1})$. If we use Chan's randomized optimization [2], we obtain a Las Vegas algorithm with expected time $O(n + m/\varepsilon^{d-1})$. Actually, there is a very practical and straightforward solution with running time $O(n + m/\varepsilon^{d})$, which we present next for completeness. (i) Obtain a 2-approximation a of the radius by finding the m-th farthest point from p. (ii) Create a set Q containing the $\Theta(m)$ points within distance $2a$ of p. (iii) Consider a grid with cells of diameter εa. Compute the radius of the ball enclosing m points from Q centered at each of the $O(1/\varepsilon^d)$ grid vertices within distance a from p, returning the smallest radius found.

Plugging the previous results together, the expected running time $t_{k,d}$ of the flat fitting algorithm, given an inlier is

$$t_{k,d} = \begin{cases} O(n/\varepsilon^{d-k})t_{k-1,d-1} & \text{if } k > 0 \\ O(n + m/\varepsilon^{d-1}) & \text{if } k = 0. \end{cases}$$

Consequently,

$$t_{k,d} = O\left(\frac{n^{k+1}}{\varepsilon^{k(d-k)}} + \frac{n^k m}{\varepsilon^{(k+1)(d-k)-1}}\right).$$

To get rid of the requirement of knowing an inlier, we apply the following random sampling technique used in [9]. Note that the set P contains m inliers. Therefore, a random element of P is an inlier with probability m/n and a random sample of n/m elements of P contains an inlier with probability at least $1-1/e$. Also, the set P of $O(n)$ elements is guaranteed to contain an inlier.

Theorem 3 *There is a Monte Carlo algorithm to compute, with constant probability, a $(1+\varepsilon)$-approximation of the k-flat that best fits m out of n points in d-dimensional space in time $O_\varepsilon(n^{k+2}/m)$ and, showing ε-dependencies,*

$$O\left(\frac{n^{k+2}}{m\varepsilon^{k(d-k)}} + \frac{n^{k+1}}{\varepsilon^{(k+1)(d-k)-1}}\right).$$

There is also also a deterministic algorithm with running time $O_\varepsilon(n^{k+2})$ and

$$O\left(\frac{n^{k+2}}{\varepsilon^{k(d-k)}} + \frac{n^{k+1} m \log(1/\varepsilon)}{\varepsilon^{(k+1)(d-k)-1}}\right).$$

3 Outer-dense Inliers

In this section, we show that for many data sets a random pair of inliers define a vector v satisfying the properties of Lemma 2 with constant probability. Consequently, we obtain a Monte Carlo algorithm with running time $O_\varepsilon(n^{k+2}/m^{k+1})$, which is linear for $m = \Omega(n)$.

We say that a halfspace H with normal vector v' is *deep* if $h_{v'}(P' \cap H) \geq h_{v'}(P')/4$. For a constant $\alpha \leq 1/2$, we say that the set P' is *α-outer-dense* if any deep halfspace H has $|P' \cap H| \geq \alpha |P'|$. The set P' is *outer-dense* if there is a constant α such that P' is α-outer-dense. The following lemma is analogous to Lemma 2 when the set P' is α-outer-dense.

Lemma 4 *If the inliers P' are α-outer-dense, then the vector $v = q - p$ defined by two random elements $p, q \in P'$ has*

$$\theta_v \leq \frac{4c_{k,d}(P)}{h_{v'}(P')} \text{ and}$$

$$\frac{h_{v'}(P')}{2} \leq \|v\| \leq 2c_{k,d}(P) + h_{v'}(P')$$

with probability at least $2\alpha^2$.

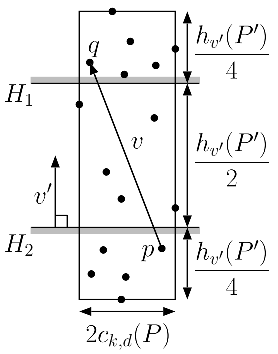

Figure 3: Proof of Lemma 4.

Proof. (sketch) Consider two disjoint deep halfspaces H_1, H_2 with normal vector v' such that v' is parallel to the optimal flat $S_{k,d}(P)$ and $h_{v'}(P' \cap H_1) = h_{v'}(P' \cap H_2) = h_{v'}(P')/4$ (see Figure 3). Since P' is outer-dense $|P' \cap H_1|, |P' \cap H_2| \geq \alpha|P'|$. Therefore, the probability that two random elements $p, q \in P'$ are one in H_1 and the other in H_2 is at least $2\alpha^2$. The lemma follows from the same trigonometric arguments as Lemma 2. $\square$

Note that if a set of points is α-outer-dense, then the projection of the set onto a $(d-1)$-dimensional hyperplane is α-outer-dense in dimension $d-1$. Therefore, we obtain a Monte Carlo algorithm by sampling $n/m\alpha^2$ pairs of points at each step, and then solving the lower dimensional problems.

Theorem 5 *When the set of inliers is outer-dense, there is a Monte Carlo algorithm to compute, with constant probability, a $(1+\varepsilon)$-approximation of the k-flat that best fits m out of n points in d-dimensional space in time $O_\varepsilon(n^{k+2}/m^{k+1})$ and, showing ε-dependencies,*

$$O\left(\frac{n^{k+2}}{m^{k+1}\varepsilon^{k(d-k)}} + \frac{n^{k+1}}{m^{k+1}\varepsilon^{(k+1)(d-k)-1}}\right).$$

4 Conclusions and Open Problems

We address a generalization to several natural problems such as the smallest m-enclosing ball ($k = 0$), infinite cylinder ($k = 1$), and slab ($k = d-1$). Except for the two extreme cases, we present the first solution for the flat fitting problem. When m is a constant fraction of n, the gap between the lower bound and our Monte Carlo upper bound is only $\Theta(n)$.

We show that if the set of inliers is outer-dense, then the problem becomes exceedingly easier, with a linear time solution. Many practical sets of inliers are outer-dense. For example, point sets uniformly distributed in a convex region and on the boundary of a convex region are outer-dense with high probability.

A related decision problem which may be useful to reduce the running time of our Monte Carlo algorithm for general point sets by an $O_\varepsilon(n)$ factor is the following. Given a set P of n points in d-dimensional space and an integer $m \leq n$, determine if there is a line ℓ that passes through the origin and is within distance 1 from m points of P. The algorithm may give an approximate answer in the sense that points within distance between 1 and $1+\varepsilon$ may be counted either way. Except for the planar case, we know of no near linear solution, nor do we know if the problem is 3SUM-hard.

References

[1] P. K. Agarwal, S. Har-Peled, and H. Yu. Robust shape fitting via peeling and grating coresets. *Discrete Comput. Geom.*, 39(1):38–58, 2008.

[2] T. M. Chan. Geometric applications of a randomized optimization technique. *Discrete Comput. Geom.*, 22(4):547–567, 1999.

[3] T. M. Chan. Approximating the diameter, width, smallest enclosing cylinder, and minimum-width annulus. *Internat. J. Comput. Geom. Appl.*, 12(1/2):67–85, 2002.

[4] T. M. Chan. Faster core-set constructions and data-stream algorithms in fixed dimensions. *Comput. Geom.*, 35(1):20–35, 2006.

[5] C. M. H. de Figueiredo and G. D. da Fonseca. Enclosing weighted points with an almost-unit ball. *Inform. Process. Lett.*, 109:1216–1221, 2009.

[6] J. Erickson, S. Har-Peled, and D. M. Mount. On the least median square problem. *Discrete Comput. Geom.*, 36(4):593–607, 2006.

[7] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, 1981.

[8] A. Gajentaan and M. H. Overmars. On a class of $O(n^2)$ problems in computational geometry. *Comput. Geom.*, 5(3):165–185, 1995.

[9] S. Har-Peled and S. Mazumdar. Fast algorithms for computing the smallest k-enclosing circle. *Algorithmica*, 41(3):147–157, 2005.

[10] S. Har-Peled and K. R. Varadarajan. Projective clustering in high dimensions using core-sets. In *Proc. 18th Annu. ACM Sympos. Comput. Geom.*, pages 312–318, 2002.

[11] S. Har-Peled and K. R. Varadarajan. High-dimensional shape fitting in linear time. *Discrete Comput. Geom.*, 32(2):269–288, 2004.