On the Least Median Square Problem*

Jeff Erickson† Sariel Har-Peled‡ David M. Mount§

December 13, 2003

Abstract

We consider the exact and approximate computational complexity of the multivariate LMS linear regression estimator. The LMS estimator is among the most widely used robust linear statistical estimators. Given a set of n points in $\mathbb{R}^d$ and a parameter k, the problem is equivalent to computing the slab bounded by two parallel hyperplanes of minimum separation that contains k of the points. We present algorithms for the exact and approximate versions of the multivariate LMS problem. We also provide nearly matching lower bounds on the computational complexity of these problems. The lower bounds hold if deciding whether $d + 1$ points are coplanar requires $\Omega(n^d)$ time.

1 Introduction

Fitting a hyperplane to a finite collection of points in space is a fundamental problem in statistical estimation. Robust estimators are of particular interest because of their insensitivity to outlying data. The principal measure of the robustness of an estimator is its *breakdown point*, that is, the fraction (up to 50%) of outlying data points that can corrupt the estimator. Rousseeuw's least median-of-squares (LMS) linear regression estimator [Rou84] is among the best known and most widely used robust estimators.

The *LMS estimator* (with intercept) is defined formally as follows. Consider a set P of n points $(\mathbf{x}_i, y_i) = (x_{i,1}, \ldots, x_{i,d-1}, y_i)$ in $\mathbb{R}^d$. The problem is to compute a parameter vector $\theta = (\theta_1, \ldots, \theta_d)$ which best fits the data by the linear model

$$y_i = x_{i,1}\theta_1 + x_{i,2}\theta_2 + \cdots + x_{i,d-1}\theta_{d-1} + \theta_d + e_i, \qquad i = 1, \ldots, n,$$

*See http://www.uiuc.edu/~jeffe/pubs/halfslab.html for the most recent version of this paper.

†Department of Computer Science, University of Illinois at Urbana-Champaign; jeffe@cs.uiuc.edu; http://www.uiuc.edu/~jeffe/. Partially supported by NSF CAREER award CCR-0093348 and NSF ITR grants DMR-0121695 and CCR-0219594.

‡Department of Computer Science, University of Illinois at Urbana-Champaign; sariel@cs.uiuc.edu; http://www.uiuc.edu/~sariel/. Partially supported by NSF CAREER award CCR-0132901.

§Department of Computer Science and Institute for Advanced Computer Studies, University of Maryland mount@cs.umd.edu; http://www.cs.umd.edu/~mount/. Partially supported by NSF grant CCR-0098151.

where $(e_1, \ldots, e_n)$ are the (unknown) errors. Given an arbitrary parameter vector $\theta \in \mathbb{R}^d$, let $r_i = y_i - \mathbf{x}_i \cdot \theta$ denote the ith residual. The LMS estimator is defined to be the parameter vector that minimizes the median of the squared residuals. More generally, given a parameter k, where $d+1 \leq k \leq n$, the problem is to find a parameter vector that minimizes the kth smallest squared residual. This is also called the least-quantile squared (LQS) estimator [RL87]. Typically, k is $\Omega(n)$. Henceforth, for uniformity, we refer to the dependent variable y_i as the dth point coordinate, $x_{i,d}$.

This estimator is widely used in practice, for example in finance, chemistry, electrical engineering, process control, and computer vision (see [Rou97]). In addition to having a high breakdown-point, the LMS estimator is regression-, scale-, and affine-equivariant, which means that the estimate transforms properly under these types of transformations [RL87]. The LMS estimator may be used on its own or as an initial step in more complex estimation schemes [Yoh87].

From a geometric perspective, it is easy to see that computing the LMS estimator is equivalent to computing the slab (that is, the closed region bounded between two parallel hyperplanes) of minimum vertical separation that encloses at least k of the points. The LMS estimator is obtained as the hyperplane that bisects this slab. In the dual setting, the problem is equivalent to finding the shortest vertical segment in $\mathbb{R}^d$ that intersects at least k of a set of n given hyperplanes. A closely related formulation, which we shall also consider, involves computing the slab enclosing k points that minimizes the perpendicular width (that is, where the width is measured normal to the hyperplanes).

The most efficient exact algorithm for computing the LMS estimator in the plane is the topological plane-sweep algorithm due to Edelsbrunner and Souvaine [ES90], which runs in $O(n^2)$ time and requires $O(n)$ space. Mount, *et al.* [MNR$^+$97] presented a practical approximation algorithm for the LMS line estimator in the plane, based approximating the quantile and/or the vertical width of the slab. Their algorithm, however, does not guarantee a better than $O(n^2)$ running time when the quantile is required to be exact. In higher dimensions, an $O(n^{d+1} \log n)$ time algorithm has long been known [MMRK91, RL87].

The LMS problem can be solved using linear programming with violations [Mat95, Cha02]. In particular, it can be solved in $O(n(n-k)^{d+1})$ time using those techniques. An alternative approach that can also be deployed in this case, is to shape fitting with outliers [HW02], which extracts a small coreset, and finds the best possible solution on this coreset. However, since we are interested in the case where $k = \Omega(n)$, this is the worst case for those algorithms, which excels in handling a small number of outliers. In our settings this is when k is very large (roughly, $k \geq n - o(n^{1/2d})$), and then those algorithms run in near linear time. It is natural to ask whether all those algorithms can be extended to handle the case where the number of outliers is large. Finally, the problem of fitting the data with two slabs of minimum width, seems to be inherently connected to the problem of LMS, as intuitively, we want to find a good clustering for most of the points by the first slab, and also a good clustering for the remaining (outlier) points. Thus, a better understanding of the LMS problem might lead to a better understanding of the 2-slab problem, which seems to be surprisingly hard [Har03], and currently no near linear time approximation algorithm is known for $d > 3$. Thus, LMS is a fundamental problem, and a better understanding of it would lead to a better understanding of several central optimization problems.

Given the high complexity of computing LMS exactly, it is natural to consider whether more efficient approximation algorithms exist. Olson presented a 2-approximation algorithms for LMS, which run in $O(n \log^2 n)$ time in the plane and in $O(n^{d-1} \log n)$ time for fixed $d \geq 3$ [Ols97].

In this paper we consider the computational complexity of the both the exact and approximate versions of LMS. We provide a $O(n^d \log n)$ time randomized algorithm for solving the LMS problem exactly in $\mathbb{R}^d$. We also provide a randomized ε-approximation algorithm whose running time is $O((n^d/k\varepsilon) \log n)$, with high probability. For the most interesting case where $k = \Omega(n)$, this is roughly $O(n^{d-1})$.

We also show a lower bound that any exact algorithm for the LMS problem requires $\Omega(n^d)$ time, and any approximation requires $\Omega(n^{d-1})$ time, thus providing a strong indication that our algorithms are close to optimal. Our lower bounds are derived by reduction from the *affine degeneracy problem*: Given a set of n points on the d-dimensional integer[1] lattice $\mathbb{Z}^d$, do any $d+1$ of the points lie on a common hyperplane? This problem can be solved by constructing the dual hyperplane arrangement in $O(n^d)$ time, and this time bound is believed to be optimal. Erickson and Seidel [ES95, Eri99b] proved a matching $\Omega(n^d)$ lower bound on the number of sidedness queries required to solve this problem; however, the model of computation in which their lower bound holds is not strong enough to solve the LMS problem, since it does not allow us to compare widths of different slabs. The strongest lower bound known in any general model of computation is $\Omega(n \log n)$, for any fixed dimension, in the algebraic decision and computation tree models [Ben83, Yao91], although the problem is known to be NP-complete when d is not fixed [Kha95, Eri99b]. The two-dimensional affine degeneracy problem is one of Gajentaan and Overmars canonical 3SUM-hard problems [GO95]. (The d-dimensional affine degeneracy problem is actually $(d+1)$-SUM-hard; however, the best lower bound that this could imply is only $\Omega(n^{\lfloor d/2 \rfloor + 1})$ [Eri99a].) In particular, our lower bounds hold if the following conjecture, which is widely believed to be correct, holds.

Conjecture 1.1 *Determining whether a set of n points on the integer lattice $\mathbb{Z}^d$ contains $d+1$ points on a common hyperplane requires $\Omega(n^d)$ time in the worst case.*

The remainder of the paper is organized as follows. In the next section we provide basic definitions and notation. In Section 3 we present our exact algorithms for LMS. In Section 4 we present the approximation algorithms. Finally, in Section 5 we present the lower bounds.

2 Nomenclature and Notation

Whenever we work in the space $\mathbb{R}^d$, we will refer to the x_d-coordinate direction as *vertical* and each of the other coordinate directions as *horizontal*. A hyperplane is vertical if it contains a vertical line, and horizontal if its normal direction is vertical.

A *slab* is the non-empty intersection of two closed halfspaces whose bounding hyperplanes are parallel. We will distinguish between two different natural notions of the "size" of a slab. The *height* of a slab σ, denoted $\mathrm{h}(\sigma)$, is the length of a vertical line segment with one

[1] In the algebraic decision tree model of computation, the restriction to integers can be removed by using formal infinitesimals in our reductions [Eri95, Eri99a].

endpoint on each bounding hyperplane. If one slab has smaller height than another, we say that the first slab is *shorter* and the second is *taller*. On the other hand, the *width* of a slab σ, denoted $\mathrm{w}(\sigma)$, is the distance between the two bounding hyperplanes, measured along their common normal direction. If one slab has smaller width than another, we say that the first slab is *narrower* and the second is *wider*. A slab whose height or width is zero is just a hyperplane. Vertical slabs, even those with zero width, have infinite height.

For a set H of n hyperplanes in $\mathbb{R}^d$, let $s_H(k)$ denote the shortest vertical segment that intersects k hyperplanes of H, and let $\ell_H(k)$ denote its length. For a hyperplane h and a real number t, let $h+t$ denote the hyperplane resulting from translating h vertically by distance t. Let $B(h,t)$ denote the slab bounded by h and $h+t$; this slab obviously has height t.

3 Exact Algorithms

Theorem 3.1 *Let H be a set of n hyperplanes in $\mathbb{R}^d$. We can compute, in $O\big(n^d \log n\big)$ time, the shortest vertical segment that stabs at least k hyperplanes of H. The algorithm is randomized and running time bound holds with high probability.*

Proof: Let $s = s_H(k)$. Clearly, the endpoints of s lie on $d+1$ hyperplanes of H, and as such let $\mathcal{B}(t) = \{B(h,t) \mid h \in H\}$. The critical value $t^* = |s|$ corresponds to a vertex in the arrangement of the hyperplanes of $\mathcal{B}(t^*)$ which has $d+1$ hyperplanes passing through it. Let $\mathcal{C}$ denote the set of values of t for which $\mathcal{B}(t)$ contains such a vertex that is defined by the boundaries of $d+1$ slabs. Thus, we have $|\mathcal{C}| = O(n^{d+1})$.

In particular, given a candidate value t, we can decide if $t^* > t$ by just computing the arrangement of the hyperplanes of $\mathcal{B}(t)$, and computing the depth of each vertex. This results in a decision procedure for this problem that runs in $O(n^d)$ time.

Next, randomly pick $O(n^d \log n)$ vertical distances from $\mathcal{C}$ by picking $d+1$ hyperplanes of H and computing the shortest vertical segment stabbing all of them. By using the decision procedure, we can now restrict our search to an interval $[u^-, u^+]$ that contains the length of the shortest segment.

Note that the interval $[u^-, u^+]$ contains, with high probability, only $O(n)$ values of $\mathcal{C}$. Thus, to compute t^*, we keep track of the vertices of the arrangement $\mathcal{A}(\mathcal{B}(t))$, by sweeping from $t = u^-$ through $t = u^+$. Clearly, only $O(n)$ events are encountered during this sweep, and as such, we can compute t^* in $O(n^d \log n)$ time. Note that we need only to keep track of the 1-skeleton of the arrangement during this sweeping process, which guarantees that we indeed can perform this efficiently in the time specified. ■

Corollary 3.2 *Given a set P of n points in $\mathbb{R}^d$, we can compute, the shortest slab containing at least k points of P in $O\big(n^d \log n\big)$ time, with high probability.*

Theorem 3.3 *Given a set P of n points in $\mathbb{R}^d$, we can compute, the narrowest slab containing at least k points of P in $O\big(n^d \log n\big)$ time, with high probability.*

Proof: Let p be any point of P. We show how one can find the narrowest slab that contains k points in P and whose boundary passes through p. To compute the narrowest slab overall, we simply run this fixed-point algorithm for every point $p \in P$ and return the thinnest result.

We first solve the decision problem, where the width r of the slab is known. Any slab can be defined by its normal direction. Thus, we consider the sphere of directions $S^{(d)}$ centered at p. For any point $q \in P$, the family of slabs of width r having q inside them defines a strip on $S^{(d)}$, which is the intersection of two parallel half-spaces and $S^{(d)}$, where one of the half-spaces' boundaries passes through p.

Let $\mathcal{U}$ be this set of strips on $S^{(d)}$. Deciding whether there is a slab that contains k points is now equivalent to deciding whether there is a point in this arrangement covered by $k-1$ strips. This can be easily decided by computing the arrangement, in $O(n^{d-1})$ time.

The exact solution can now be found easily by performing random sampling and sweeping, as was done in Theorem 3.1. We omit the remaining straightforward details. ■

4 Approximation Algorithm

Lemma 4.1 *Let H be a set of n hyperplanes in $\mathbb{R}^d$, and let h be a hyperplane. The shortest vertical segment $s_H^h(k)$ intersecting k hyperplanes of H and having its midpoint on h can be computed in $O(n^{d-1}\log n)$ time, with high probability.*

Proof: We assume h is the horizontal hyperplane $x_d = 0$. Alternatively, we can linearly transform space, without changing vertical distances, such that h is this horizontal hyperplane. Let $r \geq 0$ be a parameter, and we want to decide whether there is a vertical segment with its midpoint on h of length r that intersects k hyperplanes of H.

Let B be the slab of width r having h as its center; namely, B is the loci of all points in distance at most $r/2$ from h. For every hyperplane $g \in H$, let $\text{Proj}(g) = \{x \mid x \in h, d(x,g) \leq r/2\}$ be the feasible region for g on h, where $d(x,g)$ is the distance between x and the plane g. Clearly, $\text{Proj}(g)$ is just a $(d-1)$-dimensional slab in h.

Let $U_r = \{\text{Proj}(g) \mid g \in H\}$. Clearly, the decision procedure requires us to decide if there is any point in the arrangement of $\mathcal{A}(r) = \mathcal{A}(U_r)$ that is covered by more than k slabs. This is done by computing the arrangement $\mathcal{A}(r)$, and finding the deepest covered point. This takes $O(n^{d-1})$ time.

Consider the arrangement $\mathcal{A}(r)$ as being a function of r. Clearly, we want to compute the minimum value of r for which $\mathcal{A}(r)$ contains a point covered by k slabs. This clearly happens when d hyperplanes in this arrangement pass through a common point. Let T denote the set of all those critical values of r. Sample $O(n^{d-1}\log n)$ values of T by randomly picking d slabs in this arrangement and computing when their boundaries all pass through a common point. Using binary search, we can perform $O(\log n)$ calls to the decision procedure and find an interval $I = [r^-, r^+]$ where the required value lies, and furthermore, with high probability I contains $O(n)$ values of T.

Next, we compute the 1-skeleton of the arrangement $\mathcal{A}(u^-)$, and maintain this skeleton as we vary r from u^- to u^+. During this period, the 1-skeleton undergoes only $O(n)$ changes, and every such change can be performed in $O(\log n)$ time using standard sweeping techniques. Furthermore, one can easily maintain the depth of every vertex and edge in the 1-skeleton during this sweeping. Thus, we will compute during process the required segment $s_H^h(k)$. ■

Theorem 4.2 *Let H be a set of n hyperplanes in $\mathbb{R}^d$, and let $\varepsilon > 0$ and k be prescribed parameters. One can compute a vertical segment of length at most $(1+\varepsilon)\ell_H(k)$ that intersects k hyperplanes of H in $O\big((n^d/k\varepsilon)\log^2 n\big)$ time.*

Proof: Let R be a random sample of $O((n/k)\log n)$ hyperplanes of H. By ϵ-net theory, one of the hyperplanes of R intersects the segment $t = s_H(k)$, with high probability. Next, for every hyperplane of R, apply the algorithm of Lemma 4.1. Clearly, the length of the shortest vertical segment computed is a 2-approximation to the length of t. The running time is $O(|R|n^{d-1}\log n) = O\big((n^d/k)\log^2 n\big)$. Let u be the length of this segment, which is a 2-approximation to $\ell_H(k)$.

Next, let $R' = \{h + iu\varepsilon/4 \mid h \in R, i = -\lceil 4/\varepsilon\rceil, \ldots, \lceil 4/\varepsilon\rceil\}$, where $h + x$ is the hyperplane resulting by translating h vertically by distance x. With high probability, one of the hyperplanes of R' intersects t and is in distance at most $\varepsilon \cdot \ell_H(k)$ from the midpoint of t. As such, the shortest vertical segment computed by applying the algorithm of Lemma 4.1 to each hyperplane of R' results in the required approximation. ■

The algorithm of Theorem 4.2 clearly also solves the dual problem of finding an approximately shortest slab containing k points.

Corollary 4.3 *Let H be a set of n hyperplanes in $\mathbb{R}^d$, and let $\varepsilon > 0$ and k be prescribed parameters. One can compute a segment that intersects k hyperplanes of H of width at most $(1+\varepsilon)\ell_H(k)$ in $O\big((n^d/k\varepsilon)\log^2 n\big)$ time.*

In the special case $k = \Omega(n)$, we can improve the running time of our algorithm by a factor of $O(\log n)$ using Chan's deterministic algorithm for constructing ϵ-nets [Cha00].

Finding an approximately *narrowest* slab is only slightly different. Using a similar algorithm to the one described above, together with the techniques of Theorem 3.3, we obtain the following result. We omit the straightforward details.

Theorem 4.4 *Let P be a set of n points in $\mathbb{R}^d$, and let $\varepsilon > 0$ and k be prescribed parameters. One can compute a slab that covers k points of P of width at most $(1+\varepsilon)W_k(P)$ in $O\big((n^d/k\varepsilon)\log^2 n\big)$ time, where $W_k(P)$ is the minimum width of any slab that covers k points of P.*

5 Lower Bounds

In this section, we prove relative lower bounds for both the exact and approximate LMS hyperplane-fitting problems. Our results suggest that the exact problem requires $\Omega(n^d)$ time and that the approximate problem requires $\Omega(n^{d-1})$ time, in the worst case. Thus, it is unlikely that our LMS algorithms can be sped up by more than polylogarithmic factors, at least when k and $n-k$ are both $\Omega(n)$.

Many of our reductions rely on the following observations, whose proofs we defer to the appendix. We say that a slab is *minimal* for a set of points if its boundary contains at least $d+1$ affinely independent points from the set. The shortest and narrowest slabs containing a set are always minimal.

Lemma 5.1 *Let S be a set of n points on the integer grid $[-M \,..\, M]^d$, and let σ be any minimal slab containing at least $d+1$ points in S.*

(a) $\mathrm{h}(\sigma)$ *can be written as a ratio of integers p/q, where p and q are both $O(M^d)$.*

(b) Either $\mathrm{h}(\sigma) = 0$ *or* $\mathrm{h}(\sigma) = \Omega(1/M^d)$.

(c) Either $\mathrm{w}(\sigma) = 0$ *or* $\mathrm{w}(\sigma) = \Omega(1/M^d)$.

(c) σ has an integer normal vector $\vec{n}(\sigma)$ whose coefficients have absolute value $O(M^d)$.

(d) $\mathrm{w}(\delta)^2$ *can be written as a ratio of integers p'/q', where p' and q' are both $O(M^{4d})$.*

(All asymptotic bounds hide constant factors exponential in d.)

5.1 Exact Height

Theorem 5.2 *Conjecture 1.1 implies that computing the shortest slab containing $d+1$ points from a given set of n points in $\mathbb{Z}^d$ requires $\Omega(n^d)$ time in the worst case.*

Proof: The given points are affinely degenerate if and only if the shortest slab containing $d+1$ points has height zero. ■

Theorem 5.3 *Conjecture 1.1 implies that computing the shortest slab containing half of a given set of n points in $\mathbb{Z}^d$ requires $\Omega(n^d)$ time in the worst case.*

Proof: Suppose we are given a set S of $m = n/2 - d - 1$ points in $\mathbb{Z}^d$. Let x_d^+ and x_d^- be the largest and smallest x_d-coordinates of any point in S, respectively. In $O(n)$ time, we can construct a new set S' of $2m + 2(d+1) = n$ points by taking the union of S, a copy of S shifted upwards by $2(x_d^+ - x_d^-)$, and a set of $n - 2m = 2(d+1)$ extra points at least $5(x_d^+ - z_d^-)$ above everything else. The original set S contains $d+1$ points on a common hyperplane if and only if the shortest slab that contains $n/2$ points in S' has height exactly $2(x_d^+ - x_d^-)$. ■

This reduction can be generalized easily to either larger or smaller numbers of points in the target slab, as follows:

Theorem 5.4 *Conjecture 1.1 implies that computing the shortest slab containing k points from a given set of n points in $\mathbb{Z}^d$ requires $\Omega(\min\{k, n-k\}^d)$ time in the worst case.*

Proof: If $k < n/2$, we start with a set S of $m = k - d - 1$ points. We construct a new set S' containing two copies of S, one directly above the other, with $n - 2m$ extra points far above both copies.

Similarly, if $k > n/2$, we start with a set S of $m = n - k + d + 1$ points. We construct a new set S' contains two copies of S, one directly above the other, with $n - 2m$ extra points directly between the two copies.

In both cases, the shortest slab containing k points of S' has height equal to the vertical distance between the two copies of S if and only if S is affinely degenerate. Thus, Conjecture 1.1 implies that computing the shortest k-slab in S' requires $\Omega(m^d)$ time. In the first case, the extra points lie above the shortest slab; in the second case, the extra points are inside the shortest slab. ■

Theorem 5.5 *Conjecture 1.1 implies that computing the shortest slab containing k points from a given set of n points in $\mathbb{Z}^d$ requires $\Omega((n/k)^d)$ time in the worst case.*

Proof: Suppose we are given a set S of $m = n(d+1)/k$ points in $\mathbb{Z}^d$. In linear time, we can compute an upper bound M on the absolute value of any coordinate. Let $\varepsilon = O(1/M^{4d})$. We construct a new set S^* consisting of $k/(d+1)$ copies of S, where the ith copy is shifted upwards a distance of $i \cdot \varepsilon$. Let σ^* be the shortest slab containing k points in S^*. Lemma 5.1 implies that $S^* \cap \sigma^*$ consists of all $k/(d+1)$ copies of the points in $S \cap \sigma$, where σ is the shortest slab containing $(d+1)$ points in S. Indeed, if not, then the set K of points of S that have one of their copies in $S^* \cap \sigma^*$, must be of height larger than $\Omega(1/M^d)$, as $|K| \geq d+1$. In particular, $\mathrm{h}(\sigma^*) = (k/(d+1) - 1)\varepsilon$ if and only if $d+1$ points in S lie on a common hyperplane. ∎

There is still a gap between the lower and upper bound, but we believe that the true complexity is $\Omega(n^d)$ for any k.

5.2 Approximate Height

Theorem 5.6 *Let $H_k(S)$ be the height of the shortest slab containing k points in a set S. Conjecture 1.1 implies that computing a slab of height at most $2H_k(S)$ containing k points from a given set S of n points in $\mathbb{Z}^d$ requires $\Omega((n-k)^{d-1})$ time in the worst case.*

Proof: Suppose we are given a set S of $m = n/2 - k/2 - d - 1$ points on the integer lattice $\mathbb{Z}^{d-1}$. Let M be an upper bound the maximum absolute value of any coordinates in S, and let $\delta = 1/(d-2)!(2M)^{d-2}$; we can compute these values in $O(m)$ time.

In $O(n)$ time, we construct a new set S' comprised of three subsets: (1) a copy of S on the vertical hyperplane $x_1 = 1$, (2) a set of $k - 2(d+1)$ points within distance $\delta/5$ of the origin, all on the hyperplane $x_1 = 0$, and (3) a copy of $-S$ (the reflection of S through the origin) on the hyperplane $x_2 = -1$. For any non-vertical slab σ, let σ_x denote the intersection of σ with the hyperplane $x_1 = x$; this is a $(d-1)$-dimensional slab with the same height as σ.

If any d points of S lie on a common $(d-2)$-flat, then there is a slab of height at most $\delta/5$ containing k points of S'. Otherwise, let σ be any slab containing k points of S'. Without loss of generality, σ_1 contains at least $d+1$ points of S, so by Lemma 5.1(1), we have $\mathrm{h}(\sigma) = \mathrm{h}(\sigma_1) \geq \delta$. Thus, by approximating $H_k(S')$ within a factor of 2, we can determine whether the original set S contains a degeneracy. Conjecture 1.1 implies that this requires $\Omega(m^{d-1}) = \Omega((n-k)^{d-1})$ time in the worst case. ∎

5.3 Reducing Height to Width

Finally, we describe a general reduction from computing slabs with minimum height to computing slabs of minimum width. This reduction implies that all our lower bounds for minimizing height apply verbatim to the corresponding width problem. The key observation is that horizontally scaling $\mathbb{Z}^d$ does not change the height of any slab, although it does change the width. If we scale any point set S far enough, then sorting the non-vertical minimal slabs by width would be the same as sorting them by height; in particular, the narrowest non-vertical slab containing k points of S will also be the shortest slab containing k points of S. There are two main technical difficulties: quantifying the amount of scaling required and eliminating vertical slabs from consideration.

Suppose we want to find the shortest slab containing $k \geq d+1$ points from a given set S of n points on the integer lattice $[-M \mathinner{..} M]^d$. If M is not given, we can easily compute it in $O(n)$ time. Let S' be the set obtained by scaling S horizontally (that is, in every direction except vertically) by a large integer factor $\boxed{\Delta := \Omega(M^{6d})}$. Scaling any slab σ horizontally by Δ gives us a slab σ' with the same height, containing the corresponding subset of points.

Fix a minimal non-vertical slab σ containing at least $d+1$ points of S. Let $\vec{n}$ be the integer normal vector of σ described by Lemma 5.1(c). To obtain a normal vector $\vec{n}'$ for the scaled slab σ', we can simply scale $\vec{n}$ in the vertical direction by a factor of Δ. We can decompose $\vec{n}'$ into a vertical component $\vec{n}'_v$ and a horizontal component $\vec{n}'_h$. Lemma 5.1(c) implies that $\|\vec{n}'_h\| = O(M^d)$, and since σ is not vertical, $\|\vec{n}'_v\| \geq \Delta = \Omega(M^{6d})$. We have the following bound on the width of σ' in terms of its height:

$$\begin{aligned} \mathrm{h}(\sigma') = \mathrm{w}(\sigma') \frac{\sqrt{\|\vec{n}'_v\|^2 + \|\vec{n}'_h\|^2}}{\|\vec{n}'_v\|} &\leq \mathrm{w}(\sigma') \sqrt{1 + \frac{1}{\Omega(M^{5d})}} \\ &= \mathrm{w}(\sigma') \left(1 + \frac{1}{\Omega(M^{5d})}\right) = \mathrm{w}(\sigma') + \frac{1}{\Omega(M^{3d})} \end{aligned}$$

Lemma 5.1(a) implies that the heights of any two minimal slabs σ_1 and σ_2 either are equal or differ by at least $\Omega(1/M^{2d})$. It follows that $\mathrm{h}(\sigma_1) < \mathrm{h}(\sigma_2)$ implies $\mathrm{w}(\sigma'_1) < \mathrm{w}(\sigma'_2)$; the height order and width order of the non-vertical minimal slabs is the same, except that some equal-height pairs may not have equal width. In particular, the narrowest non-vertical slab containing k points in S' is also the shortest such slab. The entire reduction requires only linear time, and increases the bit length of the input by at most a factor of $O(d)$.

Theorem 5.7 *Conjecture 1.1 implies that computing the narrowest non-vertical slab containing k points from a given set of n points in $\mathbb{Z}^d$ requires $\Omega(\min\{k, n-k\}^d)$ time in the worst case.*

Theorem 5.8 *Let $W_k(S)$ be the width of the narrowest non-vertical slab containing k points in a set S. Conjecture 1.1 implies that computing a non-vertical slab of width at most $2W_k(S)$ containing k points from a given set S of n points in $\mathbb{Z}^d$ requires $\Omega((n-k)^{d-1})$ time in the worst case.*

What about vertical slabs? If no vertical hyperplane contains k points of S, the reduction goes through immediately. Lemma 5.1(c) implies that the narrowest *vertical* slab σ_v containing k points of S' is either a single hyperplane or it has width $\Omega(\Delta/M^{d-1}) = \Omega(M^{5d})$. In the latter case, σ_v cannot be the narrowest k-slab for S', since the entire point set fits in a slab of width $2M$. However, if some vertical hyperplane passes through k points, the shortest and narrowest k-slabs no longer coincide.

To avoid this problem, we first perturb the initial set S, essentially following the infinitesimal perturbation method of Emiris and Canny [EC95, ECS97]. Let $\boxed{\varepsilon = 1/M^{4d}}$. For any point $p \in S$, let $\tilde{p}$ denote a point at distance at most ε from S, and let $\widetilde{S} = \{\tilde{p} \mid p \in S\}$. For any slab σ that is minimal for S, we define a corresponding slab $\tilde{\sigma}$ that is minimal for $\widetilde{S}$. Lemma 5.1 implies that $\mathrm{h}(\sigma) \le O(M^d)\mathrm{w}(\sigma)$, so $\mathrm{h}(\tilde{\sigma}) \le \mathrm{h}(\sigma) + O(1/M^{3d})$, and that two minimal slabs for S differ in height by at least $1/M^{2d}$. It follows that if σ_1 is shorter than σ_2, then $\tilde{\sigma}_1$ is shorter than $\tilde{\sigma}_2$. In other words, perturbing the points by ε does not change which points are inside the shortest k-slab.

Arbitrarily index the points in S as $p_1, p_2, \ldots, p_n$, and let q be the smallest prime number larger than n (and therefore less than $2n$). We choose the following specific perturbation:

$$\tilde{p}_i = p_i + \frac{\varepsilon}{q}(i, i^2 \bmod q, i^3 \bmod q, \ldots, i^d \bmod q).$$

We can express the volume of any simplex in $\widetilde{S}$ as a polynomial in ε. Lemma 5.1 implies that the sign of this polynomial is determined by the sign of the largest term. Moreover, the coefficient ε^d term is the volume of a simplex with vertices on the modular moment curve $\frac{1}{q}(t, t^2 \bmod q, t^3 \bmod q, \ldots, t^d \bmod q)$, and is therefore not equal to zero. We conclude that no $d+1$ points in $\widetilde{S}$ lie on a common hyperplane; in particular, to k points lie on a vertical hyperplane.

Scaling the set $\widetilde{S}$ by a factor of q/ε gives us an integer point set, where every coordinate has absolute value at most $O(M^{5d})$. Thus, to find the shortest k-slab in S, we can apply our earlier reduction to $\widetilde{S}$. The entire reduction requires only linear time, and increases the bit length of the input by at most a factor of $O(d^2)$.

Theorem 5.9 *Conjecture 1.1 implies that computing the narrowest slab containing k points from a given set of n points in $\mathbb{Z}^d$ requires $\Omega(\min\{k, n-k\}^d)$ time in the worst case.*

Theorem 5.10 *Let $W_k(S)$ be the width of the shortest slab containing k points in a set S. Conjecture 1.1 implies that computing a slab of width at most $2W_k(S)$ containing k points from a given set S of n points in $\mathbb{Z}^d$ requires $\Omega((n-k)^{d-1})$ time in the worst case.*

References

[Ben83] M. Ben-Or. Lower bounds for algebraic computation trees. In *Proc. 15th Annu. ACM Sympos. Theory Comput.*, pages 80–86, 1983.

[Cha00] B. Chazelle. *The Discrepancy Method.* Cambridge University Press, 2000.

[Cha02] T. M. Chan. Low-dimensional linear programming with violations. In *Proc. 43th Annu. IEEE Sympos. Found. Comput. Sci.*, pages 570–579, 2002.

[EC95] I. J. Emiris and J. F. Canny. A general approach to removing degeneracies. *SIAM J. Comput.*, 24:650–664, 1995.

[ECS97] I. J. Emiris, J. F. Canny, and R. Seidel. Efficient perturbations for handling geometric degeneracies. *Algorithmica*, 19(1–2):219–242, September 1997.

[Eri95] J. Erickson. On the relative complexities of some geometric problems. In *Proc. 7th Canad. Conf. Comput. Geom.*, pages 85–90, 1995.

[Eri99a] J. Erickson. Lower bounds for linear satisfiability problems. *Chicago J. Comput. Sci.*, 1999(8), 1999.

[Eri99b] J. Erickson. New lower bounds for convex hull problems in odd dimensions. *SIAM J. Comput.*, 28:1198–1214, 1999.

[ES90] H. Edelsbrunner and D. L. Souvaine. Computing median-of-squares regression lines and guided topological sweep. *J. Amer. Statist. Assoc.*, 85:115–119, 1990.

[ES95] J. Erickson and R. Seidel. Better lower bounds on detecting affine and spherical degeneracies. *Discrete Comput. Geom.*, 13:41–57, 1995.

[GO95] A. Gajentaan and M. H. Overmars. On a class of $O(n^2)$ problems in computational geometry. *Comput. Geom. Theory Appl.*, 5:165–185, 1995.

[Har03] S. Har-Peled. On core-sets and slabs. manuscript, 2003.

[HW02] S. Har-Peled and Y. Wang. Shape fitting with outliers. In *Proc. 19th Annu. ACM Sympos. Comput. Geom.*, pages 29–38, 2002.

[Kha95] L. Khachiyan. On the complexity of approximating determinants in matrices. *J. Complexity*, 11:138–153, 1995.

[Mat95] J. Matoušek. On geometric optimization with few violated constraints. *Discrete Comput. Geom.*, 14:365–384, 1995.

[MMRK91] P. Meer, D. Mintz, A. Rosenfeld, and D. Y. Kim. Robust regrsssion methods for computer vision: A review. *Intl. J. of Computer Vision*, 6:59–70, 1991.

[MNR+97] D. M. Mount, N. S. Netanyahu, K. R. Romanik, R. Silverman, and A. Y. Yu. A practical approximation algorithm for the lms line estimator. In *Proc. 8th ACM-SIAM Sympos. Discrete Algorithms*, pages 473–482, January 1997.

[Ols97] C. F. Olson. An approximation algorithm for least median of squares regression. *Information Processing Letters*, 63:237–241, 1997.

[RL87] P. J. Rousseeuw and A. M. Leroy. *Robust Regression and Outlier Detection.* John Wiley & Sons, New York, 1987.

[Rou84] P. J. Rousseeuw. Least median of squares regression. *J. Amer. Statist. Assoc.*, 79(388):871–880, December 1984.

[Rou97] P. J. Rousseeuw. Introduction to positive breakdown methods. In G. S. Maddala and C. R. Rao, editors, *Handbook of Statistics, Vol 15: Robust Inference*, pages 101–121. Elsevier, Amsterdam, 1997.

[Yao91] A. C. Yao. Lower bounds for algebraic computation trees with integer inputs. *SIAM J. Comput.*, 20(4):655–668, 1991.

[Yoh87] V. J. Yohai. High breakdown-point and high efficiency robust estimates for regression. *The Annals of Statistics*, 15:642–656, 1987.

A Proof of Limited Resolution

Proof of Lemma 5.1: Let $\triangle$ be the convex hull of $d+1$ arbitrary points in S, and let σ be any minimal slab containing $\triangle$. Without loss of generality, assume that σ is not vertical. To simplify our discussion, we refer to d-dimensional Lebesgue measure as *volume*, and $(d-1)$-dimensional Lebesgue measure as *area*.

(a) The volume V of $\triangle$ is equal to $\mathrm{h}(\sigma)B/d$, where A is the sum of the signed areas of the vertical projections of certain facets of $\triangle$ onto $\mathbb{Z}^{d-1}$. Specifically, a facet contributes its area to B if it touches the lower bounding hyperplane of σ, positively if $\triangle$ is locally above the facet and negatively otherwise. Since every point coordinate is an integer, the volume of $\triangle$ is an integer multiple of $1/d!$, and the vertically projected area of each facet is an integer multiple of $1/(d-1)!$. Moreover, $\triangle$ has volume $O(M^d)$, and each projected facet has area $O(M^{d-1})$. Finally, we obtain $\mathrm{h}(\sigma) = d\,V/B = (d!\,V)/((d-1)!A)$, where $d!V$ and $(d-1)!A$ are integers in the desired range.

(b) This follows directly from part (a).

(c) Consider a line through the origin normal to the hyperplanes bounding σ. This line forms an angle of at most 45° with at least one coordinate axis. Without loss of generality, that axis is vertical, in which case we have $\mathrm{w}(\sigma) \geq \mathrm{h}(\sigma)/\sqrt{2}$.

(d) Let $p_0, p_1, \ldots, p_k, q_{k+1}, \ldots, q_d$ be affinely independent points in S on the boundary of σ, where each point p_i lies on the lower bounding hyperplane, and each point q_i lies on the upper bounding hyperplane. We define a set of $d-1$ integer vectors $\vec{v}_1, \ldots, \vec{v}_{d-1}$ parallel to σ as follows: if $i \leq k$, we take $\vec{v}_i = p_i - p_{i-1}$, and if $i > k$, we take $\vec{v}_i = q_i - q_{i+1}$. These vectors are linearly independent, since otherwise the points p_i and q_j would lie on a common hyperplane, so σ would not be minimal. The exterior product $\vec{v}_1 \wedge \vec{v}_2 \wedge \cdots \wedge \vec{v}_d$ is a vector normal to σ. Each component of this exterior product is the determinant of a $(d-1) \times (d-1)$ minor of the $(d-1) \times d$ matrix of coordinates $\vec{v}_{ij}$. Since each of these coordinates is an integer with absolute value $O(M)$, each component of the normal vectors is an integer with absolute value $O(M^{d-1})$.

(e) This follows immediately from the identity $\mathrm{w}(\sigma) = \mathrm{h}(\sigma)\, n_d(\sigma)/\|\vec{n}(\sigma)\|$, where $n_d(\sigma)$ is the vertical component of the normal vector $\vec{n}(\sigma)$. ■