On the Least Median Square Problem*

Jeff Erickson† Sariel Har-Peled‡

Department of Computer Science
University of Illinois at Urbana-Champaign
{jeffe,sariel}@cs.uiuc.edu
http://www.cs.uiuc.edu/~{jeffe,sariel}

David M. Mount§

Department of Computer Science and
Institute for Advanced Computer Studies
University of Maryland
mount@cs.umd.edu
http://www.cs.umd.edu/~mount

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 narrowest slab bounded by two parallel hyperplanes 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 for these problems, under the assumption that deciding whether n given points in $\mathbb{R}^d$ are affinely nondegenerate requires $\Omega(n^d)$ time.

Categories and Subject Descriptors: F.2.2 [Analysis of algorithms and problem complexity]: Nonnumerical problems and algorithms—*Geometric problems and algorithms*; G.3 [Probability and statistics]: Robust regression

Keywords: LMS regression, robust statistics, approximation algorithms, lower bounds

General Terms: Algorithms, Theory

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 value*, 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 [22] is among the best known and most widely used robust estimators.

The *LMS estimator* (with intercept) is defined formally as follows. Consider a set $P = \{p_1, p_2, \ldots, p_n\}$ of n points in $\mathbb{R}^d$, where $p_i = (x_{i,1}, x_{i,2}, \ldots, x_{i,d-1}, x_{i,d})$. We would like to compute a parameter vector $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_d)$ that best fits the data by the linear model

$$x_{i,d} = x_{i,1}\theta_1 + x_{i,2}\theta_2 + \cdots + x_{i,d-1}\theta_{d-1} + \theta_d + r_i$$

for all i, where $r_1, r_2, \ldots, r_n$ are the (unknown) errors, or *residuals*. The LMS estimator is defined to be the parameter vector $\boldsymbol{\theta}$ that minimizes the median of the squared residuals. More generally, given a parameter k, where $d+1 \le k \le n$, the problem is to find a parameter vector $\boldsymbol{\theta}$ that minimizes the kth smallest squared residual. This more general problem is also called the *least-quantile squared* (LQS) estimator [24]. Typically, we are interested in the case where both $k = \Omega(n)$ and $n - k = \Omega(n)$.

This estimator is widely used in practice, for example in finance, chemistry, electrical engineering, process control, and computer vision [23]. In addition to having a high breakdown value, the LMS estimator is regression-, scale-, and affine-equivariant, which means that the estimate transforms properly under these types of transformations [24]. The LMS estimator may be used either alone or as an initial step in more complex estimation schemes [26].

The LMS estimator has an easy geometric interpretation. A *slab* is the volume enclosed by two parallel hyperplanes; the *height* of a slab is the vertical separation between its bounding hyperplanes. Computing the LMS estimator is clearly equivalent to computing a slab of minimum height 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. We also consider the closely related problem of computing the slab of minimum *width* enclosing k points, where the width of a slab is measured normal to its bounding hyperplanes. This problem is also called *LMS orthogonal regression* [24].

The most efficient exact algorithm for computing the LMS estimator in the plane is a topological plane-sweep algorithm of Edelsbrunner and Souvaine [7], which runs in $O(n^2)$ time and requires $O(n)$ space. In higher dimensions, an $O(n^{d+1} \log n)$-time algorithm has long been known [19, 24].

The LMS problem can also be expressed as a linear programming problem with violations; using this formulation,

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

†Partially supported by NSF CAREER award CCR-0093348 and NSF ITR grants DMR-0121695 and CCR-0219594.

‡Partially supported by NSF CAREER award CCR-0132901.

§Partially supported by NSF grant CCR-0098151.

SCG'04, June 8–11, 2004, Brooklyn, New York, USA.

the problem can be solved in $O(n(n-k)^{d+1})$ time [18, 4]. An alternative approach that can also be deployed in this case is an algorithm by Har-Peled and Wang for approximate shape fitting with outliers [16], which extracts a small coreset and finds the best possible solution within this coreset. However, this algorithm excels only when the number of outliers is small; specifically, the running time is near-linear only if $k \geq n - o(n^{1/2d})$. The case of most interest to us, where $n - k = \Omega(n)$, is the worst case for these algorithms. It is natural to ask whether either of these approaches 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; 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 two-slab problem, which seems to be surprisingly hard [15]. 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 the LMS estimator exactly, it is natural to consider whether more efficient approximation algorithms exist. Mount *et al.* [20] presented a practical approximation algorithm for the LMS line estimator in the plane, based on 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. Olson presented a 2-approximation algorithms for LMS, which runs in $O(n \log^2 n)$ time in the plane and in $O(n^{d-1} \log n)$ time for any fixed $d \geq 3$ [21].

In this paper we consider the computational complexity of the both the exact and approximate versions of the LMS problem. In Section 3, we describe a randomized algorithm to compute the exact LMS estimator for n points in $\mathbb{R}^d$ in $O(n^d \log n)$ time with high probability. We also describe, in Section 4, a randomized ε-approximation algorithm whose running time is $O((n^d/k\varepsilon) \operatorname{polylog})$ with high probability. For the most interesting case $k = \Omega(n)$, this is faster than our exact algorithm by roughly a factor of n/ε.

In Section 5, we provide results suggesting that any exact algorithm for the LMS problem requires $\Omega(n^d)$ time, and any constant-factor approximation algorithm requires $\Omega(n^{d-1})$ time, thus providing a strong indication that our algorithms are close to optimal. Specifically, we describe linear-time reductions 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? These reductions imply lower bounds if the following widely-believed conjecture is true.

Conjecture 1.1 *Solving the d-dimensional affine degeneracy problem requires $\Omega(n^d)$ time in the worst case.*

If this conjecture is true, the $\Omega(n^d)$ lower bound is tight; the problem can be solved in $O(n^d)$ time by constructing the dual hyperplane arrangement. Erickson and Seidel [13, 12] proved an $\Omega(n^d)$ 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 [2, 25], although the problem is known to be NP-complete when d is not fixed [17, 12]. The planar affine degeneracy problem is one of Gajentaan and Overmars' canonical 3SUM-*hard* problems [14]. (In higher dimensions, the affine degeneracy problem is actually $(d+1)$-SUM-hard [12]; however, the best lower bound that this could imply is only $\Omega(n^{\lfloor d/2 \rfloor + 1})$ [11].)

[1] In the algebraic decision tree model of computation, the restriction to integers can be removed by using formal infinitesimals in our reductions [10, 11].

2. NOTATION AND TERMINOLOGY

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 "thickness" of a slab. The *height* of a slab σ, denoted $\mathrm{ht}(\sigma)$, is the length of a vertical line segment with one 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*. We denote by $\mathrm{ht}(P, k)$ the height of the shortest slab containing at least k points from a point set P. On the other hand, the *width* of a slab σ, denoted $\mathrm{wd}(\sigma)$, 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*. We denote by $\mathrm{wd}(P, k)$ the width of the narrowest slab containing at least k points in a point set P. A slab whose height or width is zero is just a hyperplane. Vertical slabs, even those with zero width, have infinite height.

The LMS problem has a natural dual formulation, which is crucial in the development of our algorithms. We use the standard duality transformation that maps any point $(a_1, a_2, \ldots, a_d)$ to the hyperplane $x_d = a_1x_1 + a_2x_2 + \cdots + a_{d-1}x_{d-1} - x_d$ and vice versa. Under this duality map, a point p is above (resp. below) a hyperplane h if and only if the dual hyperplane p^* is below (resp. above) the dual point h^*; moreover, the vertical distance between the point and the hyperplane is preserved. The dual of a slab σ is a vertical line segment σ^* of length $\mathrm{ht}(\sigma)$; a point p lies inside the slab σ if and only if the dual hyperplane p^* intersects the dual segment σ^*. Thus, the LMS problem is equivalent to finding, given a set H of hyperplanes, the shortest vertical segment that intersects at least k of them. To be consistent with the primal formulation, we let $\mathrm{ht}(H, k)$ denote the length of this segment.

For any hyperplane h and any real value t, let $h + t$ denote the hyperplane resulting from translating h upward by (vertical) distance t, and let $\sigma(h, t)$ denote the slab bounded by h and $h + t$.

3. EXACT ALGORITHMS

3.1 Minimum Height

In this section, we develop efficient exact algorithms for computing the LMS estimator in any fixed dimension. The two-dimensional problem can be solved in $O(n^2)$ time using Edelsbrunner and Souvaine's dual topological sweep algorithm [7]. Our higher-dimensional algorithm improves the previous best running time by a factor of n.

Theorem 3.1 *Given a set H of n hyperplanes in $\mathbb{R}^d$ and an integer k, we can compute the shortest vertical segment that stabs k hyperplanes of H in $O(n^d \log n)$ time with high probability.*

PROOF: Clearly, the endpoints of the required segment lie on at least $d+1$ hyperplanes of H. Let T be the (multi)set of critical values t such that the arrangement of slabs $\Sigma(t) = \{\sigma(h,t) \mid h \in H\}$ contains a vertex on $d+1$ bounding hyperplanes. For notational convenience, let $t^* = \mathrm{ht}(H,k)$. We easily observe that $|T| = O(n^{d+1})$ and that $t^* \in T$.

A vertical line segment of length t stabs k hyperplanes in H if and only if some point lies in at least k slabs in $\Sigma(t)$, or equivalently, if and only if some vertex has depth at least k in the arrangement of $\Sigma(t)$. Thus, given a candidate value t, we can decide whether $t < t^*$ in time $O(n^d)$ by constructing the arrangement of $\Sigma(t)$ and computing the depth of every vertex.

We transform this decision procedure into a search algorithm as follows. We randomly choose a sample R of $O(n^d)$ vertical distances from T, by repeatedly choosing $d+1$ hyperplanes of H and computing the shortest vertical segment stabbing all of them. We then use the decision procedure to guide a binary search through the random sample R. In $O(n^d \log n)$ time, this binary search finds an interval $[t^-, t^+]$ that contains t^* but no other elements of R. With high probability, the interval $[t^-, t^+]$ contains only $O(n \log n)$ values from T.

To locate t^* within this interval, we maintain the arrangement of slabs $\Sigma(t)$ in a kinetic data structure [1] as t varies continuously from $t = t^-$ to $t = t^+$. The combinatorics of the arrangement changes only at times in T. At each such critical event, a simplicial cell in the arrangement of $\Sigma(t)$ collapses to a point and reverses orientation. We maintain a priority queue of collapse times for the simplicial cells in $\Sigma(t)$, ignoring any cell that collapses after time t^+. We prepare the initial priority queue by constructing the arrangement of $\Sigma(t^-)$, computing the collapse time of each simplicial cell (in $O(1)$ time), and inserting only the relevant future events into the queue. Each critical event requires a constant number of priority queue operations, plus $O(1)$ additional work. Thus, the total cost of the sweep is $O(n^d + E \log E)$, where E is the number of critical events. With high probability, $E = O(n \log n)$, so the sweep requires only $O(n^d)$ time. □

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

Chan [5] observes that his randomized optimization technique [3] can be used in place of sampling and sweeping, using a constant-complexity cutting to generate subproblems, as in his other results on linear programming with violations [4]. Chan's techniques reduce the expected running time of our algorithm to $O(n^d)$ but do not improve our high-probability time bound.

3.2 Minimum Width

We can use similar techniques to solve the orthogonal LMS regression problem. A simple modification of Edelsbrunner and Souvaine's topological sweep algorithm [7] solves the two-dimensional version of this problem in $O(n^2)$ time and $O(n)$ space.

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

PROOF: We easily observe that the target slab has at least one point (in fact, $d+1$ points) from P on its boundary. We describe an algorithm to find the narrowest slab that contains k points in P and has a specific point $q \in P$ on its boundary. To find the narrowest unrestricted slab containing k points in P, we simply run this algorithm once for every reference points $q \in P$ and return the narrowest result.

We first solve the following decision problem: Is there a slab of width w, whose boundary contains q, that contains k points in P? Any slab with q on its boundary is uniquely determined by the normal vector pointing into the slab from the bounding hyperplane that contains q. Thus, we have a 'duality' transformation from slabs with q on their boundary and points on the sphere of directions S^{d-1} centered at q. For any point $p \in P$, the family of slabs of width w that contain p defines a strip $s(p,w)$ on S^{d-1}; this strip is the intersection of S^{d-1} and a slab with q on its boundary. Thus, our decision problem can be rephrased as follows: Is any point in S^{d-1} covered by k of the strips in $S(w) = \{s(p,w) \mid p \in P\}$? This problem can be decided in $O(n^{d-1} + n \log n)$ time by computing the arrangement of strips $S(w)$.

We can now transform this decision procedure into a search algorithm, increasing the running time by a factor of $O(\log n)$, using the same random sampling and sweeping techniques used to prove Theorem 3.1. We omit the remaining straightforward details. □

Again, Chan's randomized optimization technique [3, 4] can be used to reduce the expected running time of our algorithm to $O(n^d)$.

4. APPROXIMATION ALGORITHMS

Our algorithm for approximating the LMS estimator rely on the same techniques used in our exact algorithms. Specifically, we exploit the dual formulation of the problem, we use random sampling and sweeping to reduce the search to a decision problem, and we begin by considering only slabs with a given reference point (or dual hyperplane).

Lemma 4.1 *Give a set H of n hyperplanes in $\mathbb{R}^d$, a hyperplane g in $\mathbb{R}^d$ (not necessarily in H), and an integer k, we can compute the shortest vertical segment that stabs k hyperplanes in H and whose midpoint lies on g, in $O(n^{d-1} \log n + n \log^2 n)$ time with high probability.*

PROOF: As usual, we reduce the search problem to a decision problem: Given a parameter $\ell \geq 0$, is there a vertical segment of length ℓ, whose midpoint lies on g, that stabs k hyperplanes of H?

For every hyperplane $h \in H$, let $\tau(h, \ell)$ be the set of points on g that lie at vertical distance at most $\ell/2$ from h, or equivalently, the intersection of g with the slab of height ℓ centered at h. Generically, $\tau(h, \ell)$ is a $(d-1)$-dimensional slab in g; however, if h is parallel to g, then $\tau(h, \ell)$ is either the empty set or all of g. Finally, let $T(\ell) = \{\tau(h, \ell) \mid h \in H\}$. Our decision problem can now be rephrased as follows: Is any point in g covered by k or more regions in $T(\ell)$? This problem can be decided in $O(n^{d-1} + n \log n)$ time by simply constructing the arrangement of $T(\ell)$ and computing the depth of every vertex.

To complete the proof of the theorem, we apply the usual random sampling and sweeping techniques to our decision procedure. □

If $d > 2$, applying this lemma once for every hyperplane in $g \in H$ gives us an alternate proof of Theorem 3.1. Once again, we can save a factor of $O(\log n)$ in the expected running time using Chan's randomized optimization technique [3, 4].

Theorem 4.2 *Given a set H of n hyperplanes in $\mathbb{R}^d$, an integer k, and a real parameter $\varepsilon > 0$, we can compute a vertical segment of length at most $(1+\varepsilon)\,\mathrm{ht}(H, k)$ that stabs k hyperplanes of H in $O((n^d/k\varepsilon)\,\mathrm{polylog}\, n)$ time with high probability.*

PROOF: Let s be the shortest vertical segment that stabs k hyperplanes in H. Let R be a random sample of $O((n/k) \times \log n)$ hyperplanes in H. By ε-net theory, s stabs at least one of the hyperplanes of R with high probability. For every hyperplane in R, apply the algorithm of Lemma 4.1, and let $\tilde{s}$ be the shortest segment thus computed. Clearly, the length of $\tilde{s}$ is at most twice the length of s.

Now let $\delta = \varepsilon|\tilde{s}|/4$, and consider the set of hyperplanes

$$R' = \{h + i\delta \mid h \in R,\ i = -\lceil 4/\varepsilon \rceil, \ldots, \lceil 4/\varepsilon \rceil\},$$

where as usual $h+t$ is the hyperplane resulting by translating h vertically by distance t. With high probability, at least one of the hyperplanes of R' intersects s and is within distance $\delta = \varepsilon|\tilde{s}|/4 < \varepsilon|s|/2 = \varepsilon\,\mathrm{ht}(H, k)/2$ from the midpoint of s. Thus, the shortest vertical segment computed by applying the algorithm of Lemma 4.1 to each hyperplane of R' is the required $(1+\varepsilon)$-approximation.

The algorithm of Lemma 4.1 is invoked $O((n/k\varepsilon)\log n)$ times, so the overall running time is $O((n^2/k\varepsilon)\log^3 n)$ if $d = 2$ and $O((n^d/k\varepsilon)\log^2 n)$ otherwise. □

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

Corollary 4.3 *Given a set P of n points in $\mathbb{R}^d$, an integer k, and a real parameter $\varepsilon > 0$, we can compute a slab of height at most $(1+\varepsilon)\,\mathrm{ht}(P, k)$ that contains k points of P in $O((n^d/k\varepsilon)\,\mathrm{polylog}\, n)$ time with high probability.*

In the special case $k = \Omega(n)$, we can improve the running time of our algorithm by a factor of $O(\log n)$ using Chazelle's deterministic algorithm for constructing ε-nets [6].

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 further details.

Theorem 4.4 *Given a set P of n points in $\mathbb{R}^d$, an integer k, and a real parameter $\varepsilon > 0$, we can compute a slab of width at most $(1+\varepsilon)\,\mathrm{wd}(P, k)$ that contains k points of P in $O((n^d/k\varepsilon)\,\mathrm{polylog}\, n)$ time with high probability.*

5. HARDNESS RESULTS

In this section, we prove relative lower bounds for both the exact and approximate LMS hyperplane-fitting problems. Our reductions suggest that computing the exact LMS estimator requires $\Omega(n^d)$ time and that computing any constant-factor approximation 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 $\Omega(n)$. Our reductions also directly imply $\Omega(n \log n)$ lower bounds for both the exact and approximate LMS problems in any fixed dimension, in the algebraic decision tree model; they also imply that when the dimension is not fixed, even the approximate LMS problem is NP-hard.

Our reductions rely on the following observations. 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. For any set P of more than d points, the shortest and narrowest slabs containing P are both minimal.

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

(a) $\mathrm{ht}(\sigma)$ *can be written as a ratio of integers p/q, where p and q are both $O(M^d)$.*
(b) Either $\mathrm{ht}(\sigma) = 0$ *or* $\mathrm{ht}(\sigma) = \Omega(1/M^d)$.
(c) Either $\mathrm{wd}(\sigma) = 0$ *or* $\mathrm{wd}(\sigma) = \Omega(1/M^d)$.
(d) σ has an integer normal vector $\vec{n}(\sigma)$ whose coefficients have absolute value $O(M^d)$.
(e) $\mathrm{wd}(\delta)^2$ *can be written as a ratio of integers p'/q', where p' and q' are both $O(M^{4d})$.*

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

PROOF: Without loss of generality, we assume the slab σ is not vertical. We refer to d-dimensional Lebesgue measure as *volume* and $(d-1)$-dimensional Lebesgue measure as *area*.

(a) Let $\triangle$ denote the convex hull of P and let V denote the volume of $\triangle$. We can express V as the ratio $A \cdot \mathrm{ht}(\sigma)/d$, where A is the sum of the signed areas of the vertical projections of certain facets of $\triangle$ onto $\mathbb{R}^{d-1}$. Specifically, a facet contributes its area to A if it touches the lower bounding hyperplane of σ, positively if $\triangle$ is locally above that facet and negatively otherwise. Clearly, $V = O(m^d)$, and the projected area of each facet is at most $O(M^{d-1})$. Moreover, since every coordinate is an integer, V is an integer multiple of $1/d!$, and the projected area of each facet is an integer multiple of $1/(d-1)!$. We conclude that

$$\operatorname{ht}(\sigma) = \frac{dV}{A} = \frac{d!\,V}{(d-1)!\,A},$$

where the integers $d!\,V$ and $(d-1)!\,A$ are both $O(M^d)$.

(b) This follows immediately 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. If necessary, reflect P and σ across some hyperplane $x_i = x_d$ so that that axis is vertical. We now have $\operatorname{wd}(\sigma) \geq \operatorname{ht}(\sigma)/\sqrt{2}$, and the result follows from part (b).

(d) Write $P = \{p_0, p_1, \ldots, p_k, q_{k+1}, \ldots, q_d\}$, where each point p_i lies on the lower bounding hyperplane of σ and each point q_i lies on the upper bounding hyperplane of σ. We define $d-1$ 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 P would be affinely degenerate. 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 vector is an integer with absolute value $O(M^{d-1})$.

(e) This follows immediately from the identity $\operatorname{wd}(\sigma) = \operatorname{ht}(\sigma)\, n_d(\sigma)/\|\vec{n}(\sigma)\|$, where $\vec{n}(\sigma)$ is the vector normal to σ constructed in part (d), and $n_d(\sigma)$ is its vertical component. □

5.1 Exact Height

We now establish our (relative) lower bounds for computing minimum-height slabs exactly.

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 $n/2$ points from 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 P 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 P, respectively. In $O(n)$ time, we can construct a new set Q of $2m + 2(d+1) = n$ points by taking the union of P, a copy of P shifted upward by $2(x_d^+ - x_d^-)$, and a set of $n - 2m = 2(d+1)$ extra points at least $5(x_d^+ - x_d^-)$ above everything else. The original set P contains $d+1$ points on a common hyperplane if and only if the shortest slab that contains $n/2$ points in Q 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 P of $m = k - d - 1$ points. We construct a new set Q containing two copies of P, one directly above the other, with $n - 2m$ extra points far above both copies.

Similarly, if $k > n/2$, we start with a set P of $m = n - k + d + 1$ points. We construct a new set Q contains two copies of P, 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 Q has height equal to the vertical distance between the two copies of P if and only if P is affinely degenerate. Thus, Conjecture 1.1 implies that computing the shortest slab containing k points in Q 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 P of $m = n(d+1)/k$ points in $\mathbb{Z}^d$. In linear time, we compute an upper bound M on the absolute value of every coordinate. Choose an appropriate constant $\delta = O(1/kM^{d+1})$. We construct a new set P' consisting of $k/(d+1)$ copies of P, where the ith copy is shifted upward a distance of $i\delta$.

If some hyperplane h contains $d+1$ points in P, then the slab $\sigma(h, (k/(d+1) - 1)\delta)$ contains k points in P', and thus $\operatorname{ht}(P', k) \leq (k/(d+1) - 1)\delta = O(1/M^{d+1})$.

On the other hand, suppose P is affinely nondegenerate. Let σ' be any slab containing k points in P'. This slab contains copies of at least $d+1$ points in P, so its height is at least $\operatorname{ht}(P, d+1) - (k/(d+1) - 1)\delta = \Omega(1/M^d)$ by Lemma 5.1. □

These reductions suggest $\Omega(n^d)$ lower bounds for exact LMS for the most interesting case, when both k and $n-k$ are $\Omega(n)$, as well as the simplest nontrivial case $k = O(1)$. We conjecture that the true complexity is $\Omega((n-k)^d)$ for any k, but there is still a gap between the upper and lower bounds for most small values of k.

Since the affine degeneracy problem is NP-hard when the dimension d is not fixed [17, 12], our reductions imply a similar NP-hardness result for the LMS estimator.

Corollary 5.6 *For any $k \leq n - \Omega(n^c)$ for some constant c, computing the shortest slab containing k points from a given set of n points in $\mathbb{Z}^n$ is NP-hard.*

5.2 Approximate Height

Theorem 5.7 *Conjecture 1.1 implies that computing a slab of height at most $2\operatorname{ht}(P, k)$ containing k points from a given set P 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 P 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 P, 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 Q comprised of three subsets: (1) a copy of P on the vertical hyperplane $x_1 = 1$, (2) a set of $k - 2(d+1)$ points within distance $\delta/10$ of the origin, all on the hyperplane $x_1 = 0$, and (3) a copy of $-S$ (the reflection of P through the origin) on the hyperplane $x_1 = -1$. 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 P lie on a common $(d-2)$-flat, then there is a slab of height at most $\delta/5$ containing k points of Q. Otherwise, let σ be any slab containing k points of Q. Without loss of generality, σ_1 contains at least $d+1$ points of P, so by Lemma 5.1(1), we have $\mathrm{ht}(\sigma) = \mathrm{ht}(\sigma_1) \geq \delta$. Thus, by approximating $H_k(S')$ within a factor of 2, we can determine whether the original set P contains a degeneracy. Conjecture 1.1 implies that this requires $\Omega(m^{d-1}) = \Omega((n-k)^{d-1})$ time in the worst case. □

The proof of Theorem 5.5 implies an even stronger hardness result when k is small.

Theorem 5.8 *Conjecture 1.1 implies that for any function $f(n,k)$, computing a slab of height at most $f(n,k)\,\mathrm{ht}(P,k)$ containing k points from a given set P of n points in $\mathbb{Z}^d$ requires $\Omega((n/k)^d)$ time in the worst case.*

PROOF: Suppose we are given a set P of $m = n(d+1)/k$ points in the finite integer lattice $[-M\,..\,M]^d$. Following the proof of Theorem 5.5, we can construct a set P' of n points in $O(n)$ time, such that approximating $\mathrm{ht}(P',k)$ up to a factor of M tells us whether P is affinely degenerate. We can replace this factor M by any function $f(n,k)$ by setting $\delta = 1/kM^d f(n,k)^d$ in the reduction. □

Our reductions strongly suggest that our approximation algorithm is within a polylogarithmic factor of optimal, at least when $k = \Omega(n)$ or $k = O(1)$. They also imply that the approximate LMS problem is NP-hard when the dimension is not fixed.

Corollary 5.9 *For any $k \leq n - \Omega(n^c)$ for some constant c, computing a 2-approximation of the shortest slab containing k points from a given set of n points in $\mathbb{Z}^n$ is NP-hard.*

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 P 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 P will also be the shortest slab containing k points of P. 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 P of n points on the integer lattice $[-M\,..\,M]^d$. If M is not given, we can easily compute it in $O(n)$ time. Let P' be the set obtained by scaling P horizontally (that is, in every direction except vertically) by a large integer factor $\Delta := \Omega(M^{6d})$. 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 P. 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:

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

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 $\mathrm{ht}(\sigma_1) < \mathrm{ht}(\sigma_2)$ implies $\mathrm{wd}(\sigma'_1) < \mathrm{wd}(\sigma'_2)$; 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 P' 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 $O(d)$.

This completes the reduction from finding the shortest slab containing k of n given points to finding the narrowest such slab that is not vertical, but what about vertical slabs? Lemma 5.1(c) implies that the narrowest *vertical* slab σ_v containing k points of P' is either a single hyperplane or it has width $\Omega(\Delta/M^{d-1}) = \Omega(M^{5d})$. Thus, if no vertical hyperplane contains k points in P, the narrowest slab containing k points in P' is not vertical, since the entire point set fits in a slab of width $2M$, so our reduction is complete. However, if k points in P lie on a vertical hyperplane, the shortest and narrowest slabs containing k points in P' may not coincide.

To avoid this problem, we first perturb the initial set P, essentially following the infinitesimal perturbation method of Emiris and Canny [8, 9]. Let $\delta = 1/M^{4d}$. For any point $p \in P$, let $\tilde{p}$ denote a point at distance at most δ from P, and let $\tilde{P} = \{\tilde{p} \mid p \in P\}$. For any minimal slab σ containing some subset of P, we define a corresponding slab $\tilde{\sigma}$ that is minimal for the corresponding subset of $\tilde{P}$. Lemma 5.1 implies that $\mathrm{ht}(\sigma) \leq O(M^d)\,\mathrm{wd}(\sigma)$, so $\mathrm{ht}(\tilde{\sigma}) \leq \mathrm{ht}(\sigma) + O(1/M^{3d})$, and that two minimal slabs for P 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, the shortest slab containing

k points in $\tilde{P}$ has the same combinatorial description as some shortest slab containing k points in P.

Arbitrarily index the points in P 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 specific perturbation $\tilde{p}_i = p_i + \delta\mu_q(i)$, where μ_q is the modular moment curve

$$\mu_q(t) := \frac{1}{q}(t, t^2 \bmod q, t^3 \bmod q, \ldots, t^d \bmod q).$$

We can express the volume of any simplex in $\tilde{P}$ as a polynomial in δ. Lemma 5.1 implies that the sign of this polynomial is determined by the sign of the largest term. Moreover, the coefficient δ^d term is the volume of a simplex whose vertices are integer points on the modular moment curve, and is therefore not equal to zero. We conclude that no $d+1$ points in $\tilde{P}$ lie on a common hyperplane; in particular, no k points lie on a vertical hyperplane.

Scaling the set $\tilde{P}$ by a factor of q/δ gives us an integer point set, where every coordinate has absolute value at most $O(M^{5d})$. Thus, to find the shortest slab containing k points in P, we can apply our earlier reduction to $\tilde{P}$. 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 5.10 *Conjecture 1.1 implies that computing the narrowest slab containing k points from a given set of n points in $\mathbb{Z}^d$ requires time $\Omega(\min\{k, n-k\}^d)$ and $\Omega((n/k)^d)$ in the worst case.*

Theorem 5.11 *Conjecture 1.1 implies that computing a slab of width at most $2\,\mathrm{wd}(P, k)$ containing k points from a given set P of n points in $\mathbb{Z}^d$ requires time $\Omega((n-k)^{d-1})$ and $\Omega((n/k)^d)$ in the worst case.*

Corollary 5.12 *For any $k \le n - \Omega(n^c)$ for some constant c, computing a 2-approximation of the thinnest slab containing k points from a given set of n points in $\mathbb{Z}^n$ is NP-hard.*

Acknowledgments

We thank Timothy Chan for suggesting the use of his randomized optimization technique.

References

[1] J. Basch, L. J. Guibas, and J. Hershberger. Data structures for mobile data. *Proc. 8th ACM-SIAM Sympos. Discrete Algorithms*, 747–756, 1997.

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

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

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

[5] T. M. Chan. Personal communication, Feb. 2004.

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

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

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

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

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

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

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

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

[14] 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.

[15] S. Har-Peled. On core-sets and slabs. Manuscript, 2003.

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

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

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

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

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

[21] C. F. Olson. An approximation algorithm for least median of squares regression. *Inform. Process. Lett.* 63:237–241, 1997.

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

[23] P. J. Rousseeuw. Introduction to positive breakdown methods. *Handbook of Statistics, Vol. 15: Robust Inference*, 101–121, 1997. Elsevier.

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

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

[26] V. J. Yohai. High breakdown-point and high efficiency robust estimates for regression. *Ann. Stat.* 15:642–656, 1987.