

# On the Least Trimmed Squares Estimator

David M. Mount\*    Nathan S. Netanyahu<sup>†</sup>    Christine D. Piatko<sup>‡</sup>    Ruth Silverman<sup>§</sup>  
Angela Y. Wu<sup>¶</sup>

November 27, 2007

## Abstract

The linear least trimmed squares (LTS) estimator is a statistical technique for estimating the line (or generally hyperplane) of fit for a set of points. It was proposed by Rousseeuw as a robust alternative to the classical least squares estimator. Given a set of  $n$  points in  $\mathbb{R}^d$ , in classical least squares the objective is to find a linear model (that is, nonvertical hyperplane) that minimizes the sum of squared residuals. In LTS the objective is to minimize the sum of the smallest 50% squared residuals. LTS is a robust estimator with a 50%-breakdown point, which means that the estimator is insensitive to corruption due to outliers, provided that the outliers constitute less than 50% of the set. LTS is closely related to the well known LMS estimator, in which the objective is to minimize the median squared residual, and LTA, in which the objective is to minimize the sum of the smallest 50% absolute residuals. LTS has the advantage of being statistically more efficient than LMS. Unfortunately, the computational complexity of LTS is less well understood than LMS. In this paper we present new algorithms, both exact and approximate, for computing the LTS estimator. We also present hardness results for exact and approximate LTS and LTA.

---

\*Department of Computer Science, University of Maryland, College Park, Maryland. Partially supported by grant CCR-0635099. Email: mount@cs.umd.edu.

<sup>†</sup>Department of Mathematics and Computer Science, Bar-Ilan University, Ramat-Gan 52900, Israel and Center for Automation Research, University of Maryland, College Park, Maryland. Email: nathan@macs.biu.ac.il.

<sup>‡</sup>The Johns Hopkins University Applied Physics Laboratory, Laurel, Maryland. Email: christine.piatko@jhuapl.edu.

<sup>§</sup>Center for Automation Research, University of Maryland, College Park, Maryland. Email: ruthsilverman@verizon.net.

<sup>¶</sup>Department of Computer Science, American University, Washington, DC. Email: awu@american.edu.

# 1 Introduction

In statistics the standard *linear regression* (with intercept) of a set of  $n$  data points is given, where each point consists of some number of independent variables and one dependent variable. The objective is to express the dependent variable as a linear function of the independent variables. More formally, for  $1 \leq i \leq n$ , we let  $\mathbf{x}_i = (x_{i,1}, x_{i,2}, \dots, x_{i,t})$ , denote the  $i$ th vector of independent variables, and let  $y_i$  denote the corresponding dependent variable. The objective is to compute a parameter vector  $\boldsymbol{\beta} = (\beta_1, \dots, \beta_t, \beta_{t+1})$  so that

$$y_i = \beta_1 x_{i,1} + \dots + \beta_t x_{i,t} + \beta_{t+1} + e_i, \quad \text{for } i = 1, 2, \dots, n,$$

where the  $e_i$ 's are the errors. Given a  $(t+1)$ -dimensional parameter vector  $\boldsymbol{\beta}$ , define  $r_i(\boldsymbol{\beta})$  to be the *residual* of the  $i$ th point, that is,  $r_i(\boldsymbol{\beta}) = y_i - (b_1 x_{i,1} + \dots + b_t x_{i,t} + \beta_{t+1})$ .

An alternative formulation arises by considering an  $n$ -element point set  $P$  in  $\mathbb{R}^d$ , where  $d = t+1$ , whose  $i$ th point is  $p_i = (x_{i,1}, \dots, x_{i,d-1}, y_i)$ . We say that a  $(d-1)$ -dimensional hyperplane is *nonvertical* if the last coordinate ( $y$ ) of its normal vector is nonzero. (Throughout, we will use the term *hyperplane* to mean a nonvertical  $(d-1)$ -dimensional hyperplane in  $\mathbb{R}^d$ .) A model parameter vector  $\boldsymbol{\beta}$  corresponds to a hyperplane, which we will denote by  $h(\boldsymbol{\beta})$  or simply  $h$ . The residual  $r_i(h)$  is defined to be the signed vertical distance from  $h$  to  $p_i$ . In this formulation the well known least squares problem is equivalent to computing the hyperplane that minimizes the sum of the squared residuals.

Many estimators such as least squares are sensitive to *outliers*, that is, points that fail to follow the linear pattern of the majority of the points. This has motivated interest in the study of *robust estimators*. The basic 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. Perhaps the best known example of a robust estimator is Rousseeuw's *least median-of-squares* (LMS) estimator [14], which is defined to be the linear model that minimizes the median squared residual.

In this paper we will study another 50% breakdown-point estimator. Given an  $n$ -element point set  $P$  and a positive integer *trimming parameter*  $m \leq n$ , the linear *least trimmed squares* (or *LTS*) estimator is defined to be the nonvertical hyperplane that minimizes the sum of the  $m$  smallest squared residuals. More formally, given a nonvertical hyperplane  $h$ , let  $r_{[i]}(h)$  denote the residual that is the  $i$ th smallest in terms of absolute value. The *LTS cost* of  $h$  is defined to be

$$\Delta_h^{(\text{LTS})}(P, m) = \sum_{i=1}^m r_{[i]}^2(h).$$

The *LTS estimator* is a  $(d-1)$ -dimensional hyperplane  $h$  that has minimum LTS cost, and the *LTS problem* is the optimization problem of computing this hyperplane. The points having the  $m$  smallest squared residuals are called the *inliers*. Note that when  $m = n$  this is equivalent to the standard least squares estimator.

LTS has many attractive properties as a robust estimator. It achieves the same asymptotic breakdown point of 50% as LMS. Like LMS, LTS is regression-, scale-, and affine-equivariant. This means the estimator transforms ‘‘properly’’ under these types of transformations [15]. Rousseeuw and van Driessen [16] point out that LTS is in many respects superior to LMS. For example, the LTS objective function is smoother than LMS, making LTS less sensitive to local effects than LMS. It is statistically more efficient, because the LTS estimator is asymptotically normal [12], whereas the LMS estimator has a lower convergence rate [14].

The principal shortcoming of LTS is that its computational complexity is less well understood than other robust estimators. Perhaps the most practical approach is a heuristic by Rousseeuw and Van Driessen [16]. It works by repeatedly computing a small random sample, fitting a hyperplane to the sample, and then performing a series of simple local update steps, each of which replaces the current hyperplane with one that has a slightly lower cost. In practice this approach works quite well, but there are no theoretical assurances of the quality of the resulting fit. Another example of a local search heuristic is Hawkins' feasible point algorithm [11], but it does not offer any formal performance guarantees either. Hössjer [13] presented an exact  $O(n^2 \log n)$  algorithm for LTS in the plane based on plane sweep.

As we shall see, the best known algorithms for computing the LTS estimator have relative high computational complexity, and so it is natural to consider whether this problem can be solved *approximately*. There are a couple of possible ways to formulate LTS as an approximation problem. Given a point set  $P$  and trimming parameter  $m$ , let  $\Delta^{(\text{LTS})}(P, m)$  denote the minimum LTS cost.

**Residual Approximation:** The requirement of minimizing the sum of squared residuals is relaxed. Given  $\varepsilon_r > 0$ , an  $\varepsilon_r$ -residual approximation is any hyperplane  $h$  such that

$$\Delta_h^{(\text{LTS})}(P, m) \leq (1 + \varepsilon) \Delta^{(\text{LTS})}(P, m).$$

**Quantile Approximation:** As we shall see, much of the complexity of LTS arises because of the requirement that exactly  $m$  points be considered. We can relax this requirement by introducing a parameter  $\varepsilon_q > 0$  and requiring that only  $m^- = \lceil m/(1 + \varepsilon_q) \rceil$  inliers be used. An  $\varepsilon_q$ -quantile approximation is any hyperplane  $h$  such that<sup>1</sup>

$$\frac{\Delta_h^{(\text{LTS})}(P, m^-)}{m^-} \leq \frac{\Delta^{(\text{LTS})}(P, m)}{m}.$$

LTS and LMS are trimmed versions of the well known least squares ( $L_2$ ) and Chebyshev ( $L_\infty$ ) estimators, respectively. A third example is the *least trimmed sum of absolute residuals*, or LTA. This is a trimmed version of the  $L_1$  estimator, in which the objective is to minimize the sum of squares of the  $m$  smallest absolute residuals. The associated costs are:

$$\Delta_h^{(\text{LTS})} = \sum_{i=1}^m r_{[i]}^2(h) \quad \Delta_h^{(\text{LMS})} = \max_{1 \leq i \leq m} r_{[i]}^2(h) \quad \Delta_h^{(\text{LTA})} = \sum_{i=1}^m |r_{[i]}(h)|,$$

By analogy, approximations can be defined for the other trimmed estimators, LMS and LTA. (Because LMS is based on a single residual, no weighting is needed for the quantile approximation.) An efficient approximation algorithm for the nonrobust point  $L_1$  estimator was given by Har-Peled [10].

In this paper we present a number of results, both exact and approximate for the LTS and LTA estimators. In Section 2 we give two exact algorithms, the first of which is an  $O(n^2)$  time exact algorithm for LTS in the plane, and the second is an  $O(n^{d+1})$  time exact algorithm for LTS

---

<sup>1</sup>The reason that we divide each of the costs by the number of elements is that the two LTS costs are based on sums involving a different number of elements. To avoid the bias this would introduce favoring the smaller set, we compare averages rather than sums. Observe that if  $h$  is a quantile approximation by this weighted definition then (since  $m^- \leq m$ ) it is certainly a quantile approximation by the nonweighted definition.

in dimension  $d \geq 3$ . Both algorithms use  $O(n)$  space. In Section 3 we present a randomized  $\varepsilon$ -residual approximation algorithm for LTS, whose running time is  $O((n^d/m)(1/\varepsilon)^d \text{polylog } n)$ . The running time holds under the relatively weak assumption that  $1/\varepsilon$  is bounded above by a polynomial function of  $n$ . Note that for the typical case in which  $m = \Theta(n)$  and  $\varepsilon$  is a small constant, this is  $O(n^{d-1} \text{polylog } n)$ , and so is significantly better than our exact algorithm. In Section 4 we give a number of hardness results for LTA and LTS under the assumption of the hardness of solving the affine degeneracy problem. In the case where  $m = \Theta(n)$ , we provide a lower bound of  $\Omega(n^d)$  for the LTA problem. Although we do not discuss it explicitly, our algorithms can easily be extended to handle weighted point sets, where the trimming parameter is given as some fraction of the total weight. We also provide an  $\Omega(n^{d-1})$  lower bound for residual approximation of both LTS and LTA, and we present a lower bound for the  $\varepsilon$ -quantile approximation for both problems for the case  $\varepsilon < 1/(d-1)$ .

The most closely related work to ours from the perspective of techniques is the paper by Erickson, Har-Peled, and Mount [7] on the LMS estimator. They presented both exact and approximation algorithms for LMS in  $\mathbb{R}^d$  and presented hardness results. Many of the results of this paper arise by an adaptation of the methods presented there. There are a number of technical innovations needed due to the fact that, unlike LMS, both LTA and LTS depend on *all* the residual values of the inliers, not just the few that determine the solution.

## 2 Exact Algorithms

In this section we present two exact algorithms for LTS. Our first algorithm solves the problem in  $O(n^2)$  time for point sets in the plane, improving on the  $O(n^2 \log n)$  algorithm due to Hössjer [13]. This algorithm is a modification of the topological plane-sweep algorithm for LMS given by Edelsbrunner and Souvaine [5], which was based on the earlier plane sweep algorithm by Souvaine and Steiger [17]. Our second algorithm runs in any fixed dimension  $d \geq 3$  and takes  $O(n^{d+1})$  time. Both algorithms use  $O(n)$  space.

Recall that we are given an  $n$ -element point set  $P$  in  $\mathbb{R}^d$  and a positive integer trimming parameter  $m \leq n$ . We assume that the points are in general position. Define a *slab* to be the closed region bounded between two parallel (nonvertical) hyperplanes. Define the *height* of a slab to be the vertical distance (along the  $y$ -axis) between its bounding hyperplanes. A slab is *critical* if  $d+1$  points of  $P$  lie on its boundary, with at least one point on each side of the slab. Any set of  $d+1$  points in general position determines up to  $2^{d+1}$  critical slabs, depending on how they are assigned to the top and bottom of the slab. Given any hyperplane  $h$ , the subset of points of  $P$  having the  $m$  smallest squared residuals lies within some slab centered about  $h$  (which need not be critical), and further, no other point of  $P$  can lie within the interior of this slab. We define a *slab set* to be any subset of  $P$  formed by taking all the points interior to some slab and any subset of boundary points. It is easy to prove by a perturbation argument that every slab set is the slab set for some critical slab, and thus (by our general position assumption) there are  $O(n^{d+1})$  distinct slab sets.

A naive solution to the LTS problem involves enumerating all slab sets, eliminating all those that do not contain  $m$  points, computing the LTS cost of each of the remaining slab sets, and returning the minimum. Our exact algorithms are based on an efficient implementation of this simple idea. Rather than describing the algorithm in the primal setting as we have just done, it is a bit easier to visualize in the dual setting [4]. It is well known that in this setting each point is mapped to a nonvertical hyperplane, and a slab of vertical height  $x$  is mapped through the dual transformation

to a vertical segment of length  $x$  (see, e.g., [5]). The points lying within the slab correspond to dual hyperplanes that intersect this segment. The dual of a critical slab is a *critical segment*, that is, a vertical line segment whose endpoints are incident to a total of  $d + 1$  dual hyperplanes of  $P$ . Our planar LTS algorithm is based on an adaptation of aforementioned topological plane sweep algorithm. The principal new element is the need to maintain the LTS cost, which depends on  $m$  residuals, in contrast to the LMS, which depends only on the length of the critical segment. The key observation is that the LTS cost can be expressed in terms of equations involving simple polynomials of the point set. These polynomials can be updated in  $O(1)$  time as we move from event to event.

**Theorem 1** *Given an  $n$ -element planar point set  $P$  and a trimming parameter  $m$ , the linear LTS estimator can be computed in  $O(n^2)$  time and  $O(n)$  space.*

**Proof:** The algorithm is based on topological plane sweep of the dual line arrangement. For each  $i$ , where  $1 \leq i \leq n - m$  consider any vertical line segment joining levels  $i$  and  $i + m - 1$  of the arrangement. (See [4] for terminology.) Let  $\mathcal{L}_i$  and  $\mathcal{L}_{i+m-1}$  denote these levels. Such a segment intersects at least  $m$  dual hyperplanes and so corresponds to a slab that contains at least  $m$  points of  $P$ . Edelsbrunner and Souvaine first show that there are  $O(n^2)$  such critical segments. (To see this observe that each vertex of the arrangement bounds at most four such segments, two from below and two from above.) Then they apply topological plane sweep to generate all these segments in  $O(n^2)$  time and  $O(n)$  space.

Let us consider a fixed pair of levels  $\mathcal{L}_i$  and  $\mathcal{L}_{i+m-1}$  and consider the left-to-right sequence of critical segments joining these two levels (shown as broken vertical segments of Fig. 1). The algorithm of [5] generates these sequences implicitly for all such pairs of levels simultaneously. The principal modification needed to generalize the LMS algorithm to LTS is computing the minimum LTS cost between each pair of consecutive critical segments. To do this, observe that between two critical segments (shown as the shaded parallelogram of Fig. 1) all the vertical segments connecting  $\mathcal{L}_i$  and  $\mathcal{L}_{i+m-1}$  intersect the same lines of the dual arrangement of  $P$ . It follows that the associated slabs in the primal plane contain the same points of  $P$ . Solving the LTS problem reduces therefore to solving the standard least squares problem for each of these slab sets.

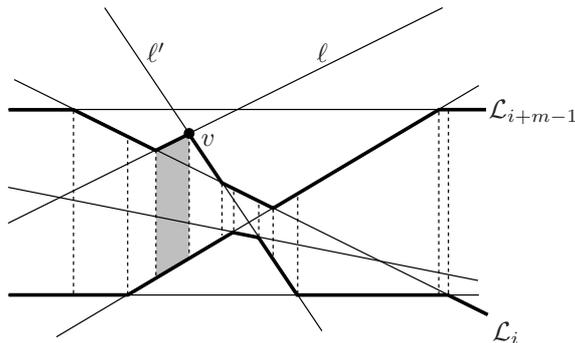


Fig. 1: Updating a critical segment between levels  $\mathcal{L}_i$  and  $\mathcal{L}_{i+m-1}$ . After sweeping vertex  $v$  we delete line  $\ell$  and insert  $\ell'$ .

The solution to least squares for a set of points  $p_i = (x_i, y_i)$  in the plane involves computing equations made of sums involving  $x_i$ ,  $y_i$ ,  $x_i^2$ ,  $y_i^2$ , and  $x_i y_i$  (see, e.g., [3]). The algorithm maintains

these sums for the current set of  $m$  inliers. As the plane sweep proceeds from one event to the next, only a constant number points are inserted or deleted from the slab set. Therefore it is possible to incrementally update the previous equations and solve them in constant time. When the algorithm terminates, we have computed the cost of the standard least squares estimator for all  $O(n^2)$  the slab sets in  $O(1)$  amortized time per set. The overall minimum is returned as the final answer. The space needed is  $O(n)$ .  $\square$

The planar algorithm makes use of the fact that in the plane there is an efficient way to enumerate the slab sets containing  $m$  points. Unfortunately, we know of no way to do this in higher dimensions. The following result shows that LTS can be solved in higher dimensions but is relatively less efficient for this reason. The general approach is to enumerate all  $(d - 1)$ -tuples of  $P$  and consider only slabs that have these points on the boundary. This reduces the number of degrees of freedom to two. The resulting restricted problem is planar in nature and can be solved by topological plane sweep.

**Theorem 2** *Given an  $n$ -element point set  $P$  in  $\mathbb{R}^d$  for any fixed  $d \geq 3$  and a trimming parameter  $m$ , the linear LTS estimator can be computed in  $O(n^{d+1})$  time and  $O(n)$  space.*

**Proof:** We follow the general approach outlined in the algorithm described in the proof of Theorem 1, except that we generate *all* the slab sets, not just those containing  $m$  points. The algorithm enumerates all  $(d - 1)$ -tuples of points of  $P$ . For each tuple we consider the at most  $2^{d-1}$  ways of assigning points to the upper and lower bounding hyperplanes of the slab. Since  $d$  is fixed, the resulting number of tuples with assignments is  $O(n^{d-1})$ . For each such tuple and assignment, we show how to solve the *restricted problem* of computing the least squares cost of all slab sets generated by critical slabs that have these points on their boundary. Finally, we return the overall minimum cost. We shall show how each restricted problem can be solved in  $O(n^2)$  time and  $O(n)$  space, which yields a total time of  $O(n^{d+1})$ .

Observe that for each restricted problem, we have eliminated all but two degrees of freedom and so the result is a planar problem. To make this more formal, first observe that we can parametrize a slab  $S$  as a  $(d + 1)$ -tuple, where  $d - 1$  coefficients suffice to encode the unit-length normal vector for the slab's hyperplanes, and the remaining two coefficients encode the intercepts of the slab's bounding hyperplanes with the vertical axis. We use the term *slab space* to refer to the resulting  $(d + 1)$ -dimensional space of coefficients. It is easy to see that the condition that a slab has a point  $p$  on either its upper or lower boundary can be expressed as a linear equation in slab space. Therefore a point in slab space satisfies the conditions of the restricted problem if and only if it satisfies a system of  $d - 1$  linear equations in  $\mathbb{R}^{d+1}$ . By our general position assumption, this is a 2-dimensional affine subspace (that is, a flat) in slab space. Henceforth, we consider the restricted problem on this 2-dimensional flat, denoted  $F$ .

The condition that any point lies within a slab  $S$  is a naturally mapped to the intersection of two halfplanes  $p^+$  and  $p^-$  on  $F$ , one for the condition that  $p$  lies below  $S$ 's upper hyperplane of  $S$  and one for the condition that  $p$  lies above  $S$ 's lower hyperplane. (Under our encoding, these halfplanes are not parallel to one another, so this region of slab space is not itself a slab.) We can therefore map every point  $p \in P$  to two halfplanes in  $F$ , such that all of the points of any of the  $O(n^2)$  cells of the resulting arrangement has the same slab set. By our general position assumption, each cell differs from its neighbor by the addition or removal of a single point of  $P$ . Therefore, as in the proof of Theorem 1, we can traverse the arrangement by topological plane sweep and

incrementally solve the associated least squares problem for each cell in  $O(1)$  amortized time. Only those cells corresponding to  $m$ -element inlier sets are retained.

□

### 3 Residual Approximation

In this section we present a residual approximation algorithm for LTS. The algorithm is randomized and runs in  $O((n^d/m)(1/\varepsilon)^d \text{polylog } n)$  time with high probability. The algorithm is an adaptation of the approximation algorithm for LMS presented by Erickson, *et al.* [7]. In that paper the LMS problem is reduced to the decision problem of finding a slab of a given vertical height that contains at least  $m$  points, which is then combined with parametric search. The principal difficulty in generalizing this to solving the decision problem for LTS is that even with knowledge of the optimum LTS cost, we cannot localize the inliers as lying within a slab of fixed width. Our approach is to replace the single slab in LMS with a sequence of  $O((1/\varepsilon) \log(n/\varepsilon))$  slabs radiating outwards from the central hyperplane. The slab widths are chosen so that all the points within any given slab are at roughly the same distance from the central hyperplane. By counting the number of points within each of the slabs, we can bound the LTS cost approximately. Although our construction is for LTS, the generalization to LTA is straightforward (and in fact it is somewhat simpler because it does not require handling squared of distances). Throughout this section we assume that  $\varepsilon \leq 1$  for technical reasons. It is easy to modify our proofs to deal with any fixed approximation parameter. We also assume that  $1/\varepsilon$  is polynomial in  $n$ , so  $\log(1/\varepsilon)$  can be absorbed in the polylog  $n$  term when stating complexity bounds.

To make this idea more formal, suppose we are given positive real parameters  $\Delta$  and  $\varepsilon$ . (We will assume for concreteness that  $\varepsilon \leq 1$ , but the results can be generalized to any  $\varepsilon$  that is bounded above by a constant.) Consider an instance of the LTS problem consisting of a point set  $P$  and a trimming parameter  $m$ . Given such an instance and a hyperplane  $h$  we define a  $(\Delta, \varepsilon)$ -slab system about  $h$  as follows. First, let  $a_0 < a_1 < \dots < a_k$  be a set of reals representing a set of distances where:

$$\begin{aligned} a_0^2 &= 0 \\ a_1^2 &= \varepsilon \frac{\Delta}{2m} \\ a_j^2 &= \left(1 + \frac{\varepsilon}{2}\right) a_{j-1}^2 = \left(1 + \frac{\varepsilon}{2}\right)^{j-1} \varepsilon \frac{\Delta}{2m} \quad \text{for } 2 \leq j \leq k. \end{aligned}$$

Let  $k$  be the smallest integer so that the last distance  $a_k$  in the sequence satisfies  $a_k^2 \geq 2\Delta$ . Thus,  $(1 + \varepsilon/2)^{k-1} \varepsilon \Delta / m \geq \Delta$ , from which it easily follows that  $k$  is  $O((1/\varepsilon) \log(m/\varepsilon)) = O((1/\varepsilon) \log n)$ , under our assumption that  $1/\varepsilon$  is polynomial in  $n$ . Let  $f(\varepsilon, m)$  denote the value of  $k$  as a function of the problem parameters.

Next, we define a set of regions  $S_1, S_2, \dots, S_k$ , each of which is the union of two slabs that are parallel to  $h$ . Define  $S_1$  to be the set of points of  $\mathbb{R}^d$  whose squared residuals with respect to  $h$  are at most  $a_1^2$ . For  $2 \leq j \leq k$ , let  $S_j$  be the set of points of  $\mathbb{R}^d$  whose squared residuals with respect to  $h$  lie in the half-open interval  $(a_{j-1}^2, a_j^2]$ . (See Fig. 2.) Clearly  $S_1$  is a slab set and the other regions are the disjoint union of two slab sets, one on each side of  $h$ . Observe that for a fixed value of  $\varepsilon$  and  $m$ , the distances of the slabs to the central hyperplane varies linearly as a function of  $\sqrt{\Delta}$ .

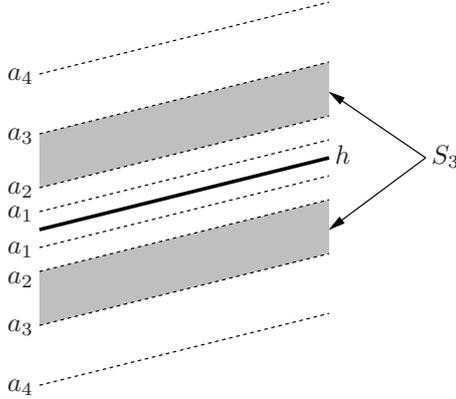


Fig. 2: A slab system surrounding a hyperplane  $h$ .

Given the point set  $P$ , for  $1 \leq j \leq k$ , define the *weight*  $w_j$  of  $S_j$  to be  $|P \cap S_j|$ . Define the *accumulated weight* to be  $W_j = \sum_{i \leq j} w_i$ . Because we are interested only in the  $m$  closest points to  $h$ , we define the  $j$ th *trimmed weight*, denoted  $\hat{w}_j$ , to be

$$\hat{w}_j = \begin{cases} w_j & \text{if } W_j \leq m, \\ \max(0, w_j - (W_j - m)) & \text{otherwise.} \end{cases}$$

Clearly  $\sum_{j=1}^k \hat{w}_j = m$ , and the sum includes the regions containing the  $m$  closest points to  $h$ . Finally, we define the approximate LTS cost to be the trimmed weighted sum of squared distances over all the groups.

$$\hat{\Delta}_h(P, m) = \sum_{j=1}^k \hat{w}_j a_j^2. \quad (1)$$

Given a hyperplane  $h$  and parameters  $\Delta$  and  $\varepsilon$ , we call the resulting collection of regions and weights a  $(\Delta, \varepsilon)$ -*slab system* for  $h$ . We show next that the value of  $\hat{\Delta}_h(P, m)$  can be used to approximately bound the true LTS cost,  $\Delta_h^{(\text{LTS})}(P, m)$ .

**Lemma 1** Consider an  $n$ -element point set  $P$  in  $\mathbb{R}^d$ , a trimming parameter  $1 \leq m \leq n$ , and positive real parameters  $\Delta$  and  $\varepsilon \leq 1$ . Given a hyperplane  $h$  and a  $(\Delta, \varepsilon)$ -slab system for  $h$ , we have

- (i) If  $W_1 > m$ , then  $\Delta_h^{(\text{LTS})}(P, m) < \Delta$ .
- (ii) If  $W_k < m$ , then  $\Delta_h^{(\text{LTS})}(P, m) > \Delta$ .
- (iii) If  $W_1 \leq m \leq W_k$ , then  $\Delta_h^{(\text{LTS})}(P, m) \leq \hat{\Delta}_h(P, m) \leq (1 + \varepsilon)\Delta_h^{(\text{LTS})}(P, m)$ .

**Proof:** To establish (i), observe that if  $W_1 > m$  then there are at least  $m$  points of  $P$  each of whose squared residual with respect to  $h$  is less than  $a_1^2$ . This implies that the LTS cost with respect to  $h$  is at most  $ma_1^2 = \varepsilon\Delta/2$ , which is less than  $\Delta$  under our assumption that  $\varepsilon \leq 1$ . To establish (ii), observe that if  $W_k < m$  then at least one point among the  $m$  closest is at squared distance greater than  $a_k^2 \geq \Delta$ , and hence the total cost is at least this large.

Henceforth, we assume that  $W_1 \leq m \leq W_k$ . Because  $m \leq W_k$ , the trimmed weighted sum of the squared distance values,  $a_j^2$ , for  $j$  ranging over all  $k$  groups is an upper bound on the sum of the  $m$  smallest squared residuals, that is,

$$\widehat{\Delta}_h(P, m) = \sum_{j=1}^k \widehat{w}_j a_j^2 \geq \sum_{i=1}^m r_{[i]}^2(h) = \Delta_h^{(\text{LTS})}(P, m).$$

Because  $W_1 \leq m$ , no more than  $m$  points lie within squared distance  $a_1^2 = \varepsilon\Delta/(2m)$ . Therefore, these point contribute at most  $\varepsilon\Delta/2$  to the total cost. For  $j \geq 2$  we have  $a_j^2 = (1 + \varepsilon/2)a_{j-1}^2$ , and every point that contributes to  $\widehat{w}_j$  is at squared distance at least  $a_{j-1}^2$  from  $h$ . Thus, we have

$$\begin{aligned} \widehat{\Delta}_h(P, m) &= \sum_{j=1}^k \widehat{w}_j a_j^2 = \widehat{w}_1 a_1^2 + \sum_{j=2}^k \widehat{w}_j a_j^2 \leq \frac{\varepsilon\Delta}{2} + \left(1 + \frac{\varepsilon}{2}\right) \sum_{j=2}^k \widehat{w}_j a_{j-1}^2 \\ &\leq \frac{\varepsilon}{2}\Delta + \left(1 + \frac{\varepsilon}{2}\right) \sum_{i=1}^m r_{[i]}^2(h) = \frac{\varepsilon}{2}\Delta_h^{(\text{LTS})}(P, m) + \left(1 + \frac{\varepsilon}{2}\right) \Delta_h^{(\text{LTS})}(P, m) \\ &= (1 + \varepsilon)\Delta_h^{(\text{LTS})}(P, m), \end{aligned}$$

which completes the proof of (iii).  $\square$

The remainder of the construction broadly follows the structure of the LMS approximation algorithm by Erickson *et al.* [7]. Given a point  $p'$  (which need not be in  $P$ ) the  $p'$ -restricted LTS problem involves computing the hyperplane passing through  $p'$  that minimizes the LTS cost. The approach involves first presenting an approximation algorithm to the restricted problem, and then showing that through an appropriate sampling process, it is possible to determine a small set of restriction points such that at least one of them will be sufficiently close to the optimum hyperplane so that a solution to the restricted problem at this point will provide the desired approximation.

Our algorithm is easiest to understand in the dual setting. As mentioned above, in this context, each point in dual space is associated with a hyperplane in primal space, and hence implicitly is associated with the LTS cost of this hyperplane. Our dual transformation will map each point of  $P$  to a  $(d - 1)$ -dimensional hyperplane in a  $d$ -dimensional dual space, but we will not use the standard dual transformation (see, e.g., [4]). To motivate our dual transformation, recall that the  $p'$ -restricted problem involves consideration only of hyperplanes passing through  $p'$ , and hence we are only interested in dual points that lie on its dual hyperplane  $h'$ . We can identify the points of  $h'$  with the points of  $\mathbb{R}^{d-1}$ , say, by projecting them onto the hyperplane spanned by the first  $d - 1$  coordinate vectors.

This determines the first  $d - 1$  coordinates of our dual transformation. The  $d$ -th coordinate of our dual transformation will be determined by the candidate LTS cost, denoted by  $\Delta$ . From Lemma 1 we know that given  $\Delta$ , the LTS cost of a hyperplane in primal space can be determined approximately by the numbers of points of  $P$  that lie within the individual slabs of a  $(\Delta, \varepsilon)$ -slab system. Recall that for fixed  $m$  and  $\varepsilon$ , the distances of the slabs from the central hyperplane vary linearly with  $\sqrt{\Delta}$ .

Combining the remarks of the previous two paragraphs, for fixed  $m$  and  $\varepsilon$ , each point of  $h'$  combined with the value  $\sqrt{\Delta}$  is associated with a unique slab system whose central hyperplane passes through  $h'$ , and vice versa. This implies that each slab system can be uniquely identified with a point in a  $d$ -dimensional *slab configuration space*, where the last coordinate (corresponding

to  $\sqrt{\Delta}$ ) is nonnegative. Each slab system partitions the points of  $P$  into  $O(f(\varepsilon, m))$  subsets, depending on which slab each point lies in. We say that two slab systems are *equivalent* if they define the same partition of  $P$ . The resulting equivalence classes subdivide the configuration space into cells. The boundaries between two cells correspond naturally to slab systems in which a point of  $P$  lies on some slab boundary. Such an event can be associated with a pair  $(p, b, i)$ , where  $p \in P$ ,  $b \in \{-1, +1\}$ , and  $i$  is an integer in the range  $\{0, \dots, f(m, \varepsilon)\}$ , corresponding to the event that  $p$  lies at distance  $a_i$  above the central hyperplane if  $b = +1$  and below the central hyperplane if  $b = -1$ . It is easy to verify that each such triple defines a  $(d - 1)$ -dimensional hyperplane in slab configuration space. Therefore, the subdivision mentioned above is just the arrangement of a set of  $2n(1 + f(m, \varepsilon))$  hyperplanes, which we call the *cost-sensitive dual arrangement*. The total complexity of this arrangement is  $O((nf(m, \varepsilon))^d) = O((n/\varepsilon)^d \text{polylog } n)$ . Also observe that fixing the value of  $\Delta \geq 0$  defines a  $(d - 1)$ -dimensional slice of this arrangement, denoted  $\mathcal{A}_{m, \varepsilon}(\Delta)$ , whose complexity is  $O((n/\varepsilon)^{d-1} \text{polylog } n)$ . When  $m$  and  $\varepsilon$  are clear from context, we refer to this arrangement simply as  $\mathcal{A}(\Delta)$ . (An example is shown in Fig. 3.)

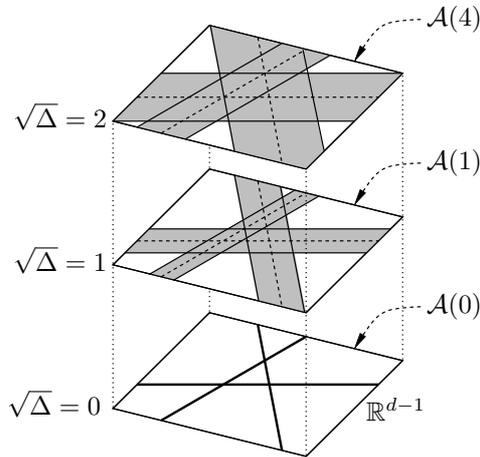


Fig. 3: Dual transformation for a simple case involving three points and a slab system consisting of a single slab.

The principal advantage of expressing matters in terms of the cost-sensitive dual arrangement is that, since one of the coordinates of the configuration space varies monotonically with the LTS cost,  $\Delta$ , we can solve the problem by performing a parametric search on this coordinate. Each probe of the parametric search is given a value  $\Delta$ , and it searches the arrangement  $\mathcal{A}(\Delta)$  to determine whether there exists a feasible solution of cost approximately at most  $\Delta$ . (The LMS algorithm given Erickson *et al.* [7] can be viewed as a special case of this approach where there is only one slab, that is,  $f(m, \varepsilon) = 1$ .)

In order to implement this approach in our case, we first demonstrate that the restricted decision problem can be solved approximately. Given a point set  $P$ , trimming parameter  $m$ , a target cost  $\Delta$  and an approximation parameter  $\varepsilon$ , we say that an algorithm solves the  $\varepsilon$ -approximate LTS decision problem if it accepts whenever  $\Delta \leq \Delta_h^{(\text{LTS})}(P, m)$ , it rejects whenever  $\Delta > (1 + \varepsilon)\Delta_h^{(\text{LTS})}(P, m)$ , and otherwise it might either accept or reject. The  $p'$ -restricted decision problem is defined analogously, subject to the constraint that the hyperplane passes through  $p'$ .

**Lemma 2** *Consider an instance of the restricted LTS decision problem consisting of an  $n$ -element set of points  $P$  in  $\mathbb{R}^d$ , a trimming parameter  $m$ , a target cost  $\Delta > 0$ , an approximation parameter  $0 < \varepsilon \leq 1$ , and a restriction point  $p' \in \mathbb{R}^d$  (not necessarily in  $P$ ) it is possible to solve the  $p'$ -restricted  $\varepsilon$ -approximate LTS decision problem in time  $O((n/\varepsilon)^{d-1} \text{polylog } n)$ .*

**Proof:** Given the value  $\Delta$ , we begin by constructing the  $(d-1)$ -dimensional arrangement  $A_{m,\varepsilon}(\Delta)$ , defined earlier. Recall that each cell of this arrangement corresponds to a set of equivalent partitions of the point set  $P$  into  $O(f(m,\varepsilon))$  slab sets. Let  $k$  denote the exact number of slab sets. For each cell, we maintain a  $k$ -element vector of weights  $(w_1, w_2, \dots, w_k)$ , where  $w_j$  denotes the number of points of  $P$  in the  $i$ th slab set.

By our assumption that  $P$  is in general position, the weight vectors associated with adjacent cells differ by the addition or subtraction of a single point from at most two weight components. Thus, by traversing this arrangement, we can compute these weight vectors for every cell of the arrangement in constant amortized time per cell. Therefore, for each cell of the arrangement, we can compute the associated approximate LTS cost based on the weight values as defined in Eq. (1). If for any cell we have  $W_1 > m$ , then Lemma 1(i) implies that the LTS cost for this cell is strictly less than  $\Delta$ , and so we may immediately accept. Similarly, if the total weight  $W_k$  is less than  $m$  for any cell, assertion (ii) of the lemma implies that the cell may be ignored since it does not contain a sufficient number of inliers. Otherwise, assertion (iii) of the lemma implies that the quantity computed  $\widehat{\Delta}_h(P, m)$  is indeed an  $\varepsilon$ -approximation to the true LTS cost for all primal hyperplanes whose dual point lies within this cell. We therefore accept if this cost is at most  $\Delta$ .

The algorithm's running time is dominated by the total size of the arrangement, which is  $O((nf(\varepsilon, m))^{d-1})$  if  $d \geq 3$ , and  $O(nf(\varepsilon, m) \log n)$  if  $d = 2$ . In either case the overall running time is  $O((n/\varepsilon)^{d-1} \text{polylog } n)$  under our assumption that  $1/\varepsilon$  is polynomial in  $n$ .  $\square$

Next we show how to convert this approximate decision algorithm into an approximation algorithm for the restricted problem. We apply the randomized sample-and-sweep approach given in [7]. Note that Chan's randomized sample-and-recurse optimization technique [2] could be applied here as well, and would result in a slightly better running time by eliminating a logarithmic factor.

**Lemma 3** *Given an  $n$ -element set of points  $P$  in  $\mathbb{R}^d$ , a trimming parameter  $m$ , an approximation parameter  $0 < \varepsilon \leq 1$ , and a point  $p' \in \mathbb{R}^d$  (not necessarily in  $P$ ), it is possible to compute an  $\varepsilon$ -residual approximation to LTS subject to the restriction that the LTS hyperplane passes through  $p'$  in time  $O((n/\varepsilon)^{d-1} \text{polylog } n)$ .*

**Proof:** As mentioned earlier, the approach is to apply a parametric search to the cost-sensitive dual arrangement. Each of the  $O((n/\varepsilon)^d \text{polylog } n)$  vertices of this arrangement corresponds to a slab system that passes through  $p'$  and has  $d$  points of  $P$  lying on the boundaries of its slabs. Each such slab system is associated with a unique cost value  $\Delta$ , which is given by the square root of the  $d$ th coordinate of the dual vertex.

We first compute a random sample  $R$  of  $O(n^{d-1})$  arrangement vertices (by sampling random  $d$ -element subsets of  $P$  and constructing the associated vertex in the cost-sensitive dual arrangement). We then sort them according to their  $\Delta$  values and apply Lemma 2 in concert with a binary search to bracket a pair of consecutive cost values,  $[\Delta^-, \Delta^+]$  such that  $\Delta^- \leq (1+\varepsilon)\Delta^{(\text{LTS})}(P, m)$  and  $\Delta^+ \geq \Delta^{(\text{LTS})}(P, m)$ . Therefore, an approximate solution to the restricted problem lies somewhere between these values. The sorting and binary search can be performed in time  $O((n/\varepsilon)^{d-1} \text{polylog } n)$ .

With high probability, the number of arrangement vertices sandwiched between the bracketing values is  $O(n \log n)$ . To locate the final value we construct the arrangement  $\mathcal{A}(\Delta^-)$  and then

perform a sweep from  $\Delta^-$  to  $\Delta^+$ , updating the arrangement as we go. The complexity of each arrangement slice is  $O((n/\varepsilon)^{d-1} \text{polylog } n)$  and there are (with high probability)  $O(n \log n)$  events processed in the sweep, each of which involves  $O(\log n)$  time to process. The overall running time is larger than the decision procedure by a factor of  $\log n$ , which is absorbed by the polylog  $n$  factor.  $\square$

Before proving the main result, we present one more technical lemma, which relates the LTS costs of two parallel hyperplanes as a function of the vertical distance between them.

**Lemma 4** *Consider a set of points  $P$  in  $\mathbb{R}^d$  and a trimming parameter  $m$ . Let  $h$  and  $h'$  be two parallel hyperplanes, and let  $\Delta$  and  $\Delta'$  denote their respective LTS costs. Suppose that these hyperplanes are separated by a vertical distance of  $\alpha\sqrt{\Delta/m}$ , for some  $\alpha \geq 0$ . Then  $\Delta' \leq (1+\alpha)^2\Delta$ .*

**Proof:** Let  $\{r_1, r_2, \dots, r_m\}$  denote the residuals of the  $m$  closest points of  $P$  to  $h$ . We have  $\Delta = \sum_{i=1}^m r_i^2$ . Let  $y = \alpha\sqrt{\Delta/m}$  denote the vertical distance between  $h$  and  $h'$ . It follows that the LTS cost of  $h'$  satisfies

$$\Delta' \leq \sum_{i=1}^m (|r_i| + y)^2.$$

For any sequence of reals  $\langle a_1, \dots, a_m \rangle$ , it is well known that  $\sum_{i=1}^m |a_i| \leq (m \sum_{i=1}^m a_i^2)^{1/2}$ . (This follows either from Schwartz's inequality [9] or the observation that in dimension  $m$  the  $L_1$  norm is larger than the  $L_2$  norm by a factor of at most  $\sqrt{m}$ .) By combining this with the above inequality we have

$$\begin{aligned} \Delta' &\leq \sum_{i=1}^m (r_i^2 + 2|r_i|y + y^2) = \sum_{i=1}^m r_i^2 + 2y \sum_{i=1}^m |r_i| + m y^2 \\ &\leq \Delta + 2y \left( m \sum_{i=1}^m r_i^2 \right)^{1/2} + m y^2 = \Delta + 2y\sqrt{m\Delta} + m y^2 \\ &= \left( \sqrt{\Delta} + \sqrt{m} y \right)^2 = \left( 1 + \sqrt{\frac{m}{\Delta}} y \right)^2 \Delta. \end{aligned}$$

Substituting the value  $y = \alpha\sqrt{\Delta/m}$  above yields the desired conclusion.  $\square$

Finally we present the main result of this section, which shows how to use the approximate solution to the restricted problem to produce an approximation algorithm for the general problem.

**Theorem 3** *Given an  $n$ -element set of points  $P$  in  $\mathbb{R}^d$ , a trimming parameter  $m$ , and an approximation parameter  $0 < \varepsilon \leq 1$ , it is possible to compute an  $\varepsilon$ -residual approximation to the linear LTS estimator in time  $O((n^d/m)(1/\varepsilon)^d \text{polylog } n)$  with high probability.*

**Proof:** Let  $\Delta = \Delta^{(\text{LTS})}(P, m)$  denote the optimum LTS cost of  $P$ , and let  $h$  denote the optimum LTS hyperplane. Consider a slab  $S$  centered about  $h$  whose boundaries are at vertical squared distance  $2\Delta/m$  from  $h$ . Observe that at least  $m/2$  points of  $P$  must lie within  $S$ , for otherwise the LTS cost of  $h$  would be strictly greater than  $(m/2)(2\Delta/m) = \Delta$ .

Let  $P'$  be a random sample of  $O((n/m) \log n)$  points of  $P$ . Apply the restricted approximate LTS algorithm of Lemma 3 for every point  $p' \in P'$  with  $\varepsilon$  set to 1, and let  $\Delta'$  be the minimum LTS cost that results. Clearly  $\Delta' \geq \Delta$ . Also, by standard results on epsilon-nets [1], with high

probability at least one of the points  $p' \in P'$  lies within  $S$ , that is, it is within distance  $\sqrt{2\Delta/m}$  of the optimum hyperplane  $h$ .

Consider the hyperplane  $h'$  parallel to  $h$  passing through  $p'$ . By applying Lemma 4 with  $\alpha = \sqrt{2}$ , it follows that  $\Delta' \leq (1 + \sqrt{2})^2 \Delta \leq 6\Delta$ . Let  $\delta = (\varepsilon/3)\sqrt{\Delta'/(6m)}$ , and consider a collection of values  $\{a_0, a_1, a_2, \dots, a_k\}$ , where  $a_i = i\delta$  and

$$k = \left\lceil \frac{\sqrt{2\Delta'/m}}{\delta} \right\rceil = O\left(\frac{1}{\varepsilon}\right).$$

We create a set  $P''$  of size  $(2k+1)|P'| = O((1/\varepsilon)(n/m) \log n)$  by generating  $2k+1$  copies of each point  $p' \in P'$  by translating it vertically up and down by distance  $a_i$ . We then invoke the algorithm of Lemma 3 to each of the points of  $P''$  using an approximation parameter of  $\varepsilon/3$ , and return the overall minimum.

To see that this achieves the desired approximation bound, recall that some  $p' \in P'$  is within distance  $\sqrt{2\Delta/m}$  of  $h$ . Since

$$k\delta \geq \sqrt{\frac{2\Delta'}{m}} \geq \sqrt{\frac{2\Delta}{m}},$$

it follows that some point of  $P''$  lies within distance

$$\frac{\delta}{2} \leq \frac{\varepsilon}{6} \sqrt{\frac{\Delta'}{6m}} \leq \frac{\varepsilon}{6} \sqrt{\frac{\Delta}{m}}$$

of  $h$ . Let  $p''$  be this point, and let  $h''$  be the hyperplane parallel to  $h$  passing through  $p''$ . By applying Lemma 4 with  $\alpha = \varepsilon/6$ , it follows that the optimum LTS cost of the restricted problem at  $p''$ , which we denote by  $\Delta''$ , is at most  $(1 + \alpha)^2 \Delta = (1 + \varepsilon/6)^2 \Delta$ . Thus, when the restricted problem is solved approximately at  $p''$  using the approximation parameter  $\varepsilon/3$ , the result has cost at most  $(1 + \varepsilon/3)\Delta'' \leq (1 + \varepsilon/3)(1 + \varepsilon/6)^2 \Delta$ . Under our assumption that  $\varepsilon \leq 1$ , it follows that this is at most  $(1 + \varepsilon)\Delta$ , as desired.

The running time is dominated by the  $O((n/m\varepsilon) \log n)$  invocations of the algorithm of Lemma 3, each of which takes  $O((n/\varepsilon)^{d-1} \text{polylog } n)$  time. This yields the desired running time.  $\square$

Note that for the typical case in which  $m = \Theta(n)$  and  $\varepsilon$  is a fixed constant, the running time is  $O(n^{d-1} \text{polylog } n)$  with high probability. A closer analysis of the polylogarithmic factors shows that the logarithmic factor in the running time of Lemma 2 is  $O(\log^{d-1} n)$ , and the two subsequent constructions each add an  $O(\log n)$  factor, for a total factor of  $O(\log^{d+1} n)$ . We add in closing that the methods applied in this section, computing a  $(\Delta, \varepsilon)$ -slab system, the cost-sensitive dual arrangement, the sample-and-sweep algorithm, and the final sampling step can all, through straightforward modifications be generalized from squared distances (that is, the  $L_2$  cost) to any  $L_p$  cost function for any fixed  $p \geq 1$ . In particular, it follows that this approach can be applied to the LTA problem with the same computational cost.

## 4 Hardness Results

In this section we present hardness results for both exact and approximate versions of the LTS and LTA problems. These results are based on reductions to the following well known decision problem, called the *affine degeneracy problem*: Given a set of  $n$  points on the  $d$ -dimensional integer lattice  $\mathbb{Z}^d$ , do any  $d+1$  of the points lie on a common hyperplane? It has been conjectured that this

problem requires  $\Omega(n^d)$  time to solve in the worst case. Erickson and Seidel [6, 8] proved an  $\Omega(n^d)$  lower bound on the number of sidedness queries required to solve the affine degeneracy problem, but the model of computation in which their lower bound holds is not strong enough to solve our problems.

Some of our hardness results rely on the following lemma. It bounds the minimum width of a slab that contains  $d+1$  points that are affinely independent. The first part of the following theorem was proved by Erickson, *et al.* [7] and the second part is a straightforward consequence.

**Lemma 5** *Let  $d$  be a fixed constant. For some integer  $M \geq 1$ , define  $\mathbb{Z}^d(M)$  to be the integer grid  $[-M .. M]^d$ . Then there exists a positive constant  $\alpha$  (depending on  $d$  but not on  $M$ ) such that the following hold.*

- (i) *Let  $P \subseteq \mathbb{Z}^d(M)$  be a set of  $d+1$  affinely independent points from the grid. The vertical height of any slab containing  $P$  is at least  $\alpha/M^d$ .*
- (ii) *Let  $P \subseteq \mathbb{Z}^d(M)$  be a set of  $m \geq d+1$  affinely independent points. Let  $h$  be any hyperplane and let  $\Delta^{(\text{LTA})}$  and  $\Delta^{(\text{LTS})}$  denote, respectively, the sum of the absolute and squared residuals of  $P$  with respect to  $h$ . Then*

$$\Delta^{(\text{LTA})} \geq \frac{(m-d)\alpha}{M^d} \quad \text{and} \quad \Delta^{(\text{LTS})} \geq \frac{(m-d)\alpha^2}{M^{2d}}.$$

**Proof:** Recall that a slab is critical if at least  $d+1$  points of  $P$  lie on its boundary. A simple perturbation argument implies that for any set  $P$  of at least  $d+1$  points, the slab of minimum vertical height containing  $P$  is critical. Lemma 5.1(c) from reference [7] shows that if  $P$  is affinely independent then any critical slab has vertical height at least  $\alpha/M^d$ , for some  $\alpha$  depending only on dimension. Therefore, any slab containing  $P$  will have at least this height.

To prove assertion (ii) consider the slab of minimum vertical width centered about  $h$  that contains at least  $d+1$  points of  $P$ . Let  $v$  denote its vertical height. By assertion (i),  $v \geq \alpha'/M^d$  for some  $\alpha'$ . By minimality at least one of these points is at distance at least  $v/2$  from  $h$  as are the remaining  $m - (d+1)$  points of  $P$ . Therefore each of these  $m-d$  points contributes at least  $(\alpha'/2)/M^d$  to the LTA cost, and at least  $(\alpha'/2)^2/M^{2d}$  to the LTS cost. Setting  $\alpha = \alpha'/2$  satisfies all the requirements of the lemma.  $\square$

#### 4.1 Hardness of Exact LTA

The main result of this section is the following theorem regarding the LTA estimator. For technical reasons the proof method makes critical use of the linearity of absolute residuals in the definition of the LTA cost, and hence it is unclear whether a similar lower bound applies to LTS.

**Theorem 4** *Under the assumption of the hardness of affine degeneracy, computing the linear LTA estimator for a given set of  $n$  points in  $\mathbb{Z}^d$  and for  $m \geq d+1$  inliers, where  $m-d$  is odd, requires  $\Omega(\min(m, n-m)^d)$  time in the worst case.*

Before giving the proof of this theorem, we present two straightforward technical lemmas. The first presents three simple inequalities involving the sums of absolute values.

**Lemma 6** *Let  $h$ ,  $p$ , and  $t$  be reals, such that  $t \geq 0$ . Then*

- (i)  $|p - h| + |(p + t) - h| \geq t$  and equality holds if  $p \leq h \leq p + t$ .
- (ii)  $|p - h| + |(p - t) - h| \geq t$  and equality holds if  $p - t \leq h \leq p$ .
- (iii)  $|p - h| + |(p + t) - h| + |(p - t) - h| \geq 2t + |p - h|$  and equality holds if  $p - t \leq h \leq p + t$ .

**Proof:** Clearly  $|a| + |b| \geq |a + b|$ , for any  $a$  and  $b$ , and equality is achieved if both  $a$  and  $b$  are of the same sign. To prove assertion (i) observe that  $|-(p - h)| + |(p + t) - h| \geq |-(p - h) + (p + t - h)| = t$ . If  $p \leq h \leq p + t$  then both  $-(p - h)$  and  $(p + t) - h$  are nonnegative, and so equality holds. To prove (ii) observe that  $|p - h| + |-(p - t) - h| \geq |p - h - (p - t - h)| = t$ . If  $p - t \leq h \leq p$  then both  $p - h$  and  $-((p - t) - h)$  are nonnegative, and so equality holds. Finally, to prove (iii) we consider two cases. If  $h \geq p$  then by (i) we have

$$(|p - h| + |(p + t) - h|) + |(p - t) - h| \geq t + |(p - t) - h| \geq t + |-(h - p) + t|.$$

Since  $h - p$  and  $t$  are both positive, this is at least  $2t + |h - p|$ . Equality holds if  $h \leq p + t$ . A symmetric argument applies when  $h < p$ .  $\square$

**Lemma 7** *Let  $d, n, n_0, n_1, n_2,$  and  $n_3$  be nonnegative integers such that  $n_0 + n_1 + n_2 + n_3 = n$  and  $n_1 + 2n_2 + 3n_3 = 2n + (d + 1)$ . Then*

- (i)  $n_2 + 2n_3 \geq n + (d + 1)$ ,
- (ii)  $n_3 - n_1 \geq d + 1$ .

**Proof:** If we subtract the first equation from the second equation we have  $-n_0 + n_2 + 2n_3 = n + (d + 1)$ . If we multiply the first equation by 2 and subtract it from the second equation we have  $-2n_0 - n_1 + n_3 = d + 1$ .  $\square$

Before giving the proof of Theorem 4, we present a special case, when  $m = 2n/3 + (d + 1)$ . The proof is similar in spirit to Theorem 5.3 of [7] for LMS, but there are a number of elements. The construction of [7] involved creating two copies of the point set, one stacked above the other at some distance  $t$ . The presence of an affine degeneracy in the original point set implies the existence of an LMS slab of vertical height exactly  $t$ . The significant added complication here is that the LTA estimator is determined by the residuals of *all* the  $m$  inliers, and not just the  $d + 1$  points that bound the minimal slab. Thus the distribution of the points lying within the slab needs to be taken into consideration. At a glance this would seem to complicate matters excessively. There is a remarkably simple fix, however. Rather than making two copies, we make three copies. We show that by symmetry, all but  $d + 1$  of the residuals can be grouped into pairs, such that sum of residuals of each pair is exactly  $t$ .

**Lemma 8** *Let  $n$  and  $m$  be positive integers, where  $n$  is divisible by 3 and  $m = (2n/3) + (d + 1)$ . Under the assumption of the hardness of affine degeneracy, computing the LTA hyperplane for a given set of  $n$  points in  $\mathbb{Z}^d$  and for  $m$  inliers requires  $\Omega(n^d)$  time in the worst case.*

**Proof:** Consider a point set  $P$  consisting of  $n'$  points for which we wish to solve the affine degeneracy problem. Let  $n = 3n'$ , and let the trimming parameter be  $m = 2n' + (d + 1)$ . Clearly,  $m = (2n/3) + (d + 1)$ , as desired. Given a positive distance  $t$ , let  $\mathbf{t}$  denote a vertical vector directed upwards of length  $t$ . Let  $t$  be a positive real value that is sufficiently large that if  $h$  is any nonvertical hyperplane passing through  $d$  or more points of  $P$ , then  $P + \mathbf{t}$  lies entirely above  $h$  and  $P - \mathbf{t}$  lies

entirely below.<sup>2</sup> For example, if we know that the points of  $P$  lie on an integer grid  $[-M, +M]^d$  for some  $M$ , then  $t = O(M^2)$  suffices.

Let  $\Delta = (n' + (d+1))t$ . We create a point set  $Q$  of size  $n$  by taking the union of three translated copies of  $P$ , called *clones*,  $P$ ,  $P - \mathbf{t}$ , and  $P + \mathbf{t}$ . (See Fig. 4.) We claim that  $P$  contains  $d+1$  points lying on a common hyperplane  $h$  if and only if  $\Delta_h^{(\text{LTA})}(Q, m) \leq \Delta$ . Assuming the hardness of affine degeneracy, this claim will imply a lower bound of  $\Omega((n')^d) = \Omega(n^d)$  on the hardness of LTA.

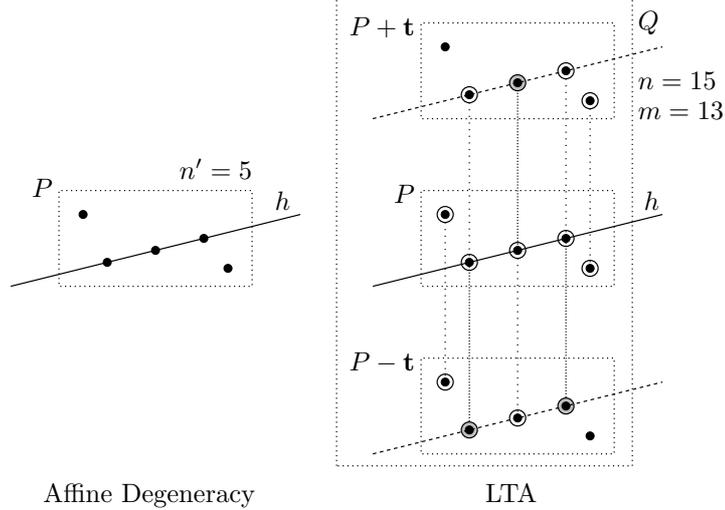


Fig. 4: Proof of Lemma 8. (Not drawn to scale.) The circled points are in  $Q'$ .

It suffices to establish the above claim. To prove the “only if” part, let us assume that there exists a hyperplane  $h$  containing at least  $d+1$  points of  $P$ . Let  $P^0$  denote any  $d+1$  points of  $P$  lying on  $h$ , and among the remaining points of  $P$ , let  $P^+$  be the points of  $P$  that lie on or above  $h$  and let  $P^-$  be the points that lie strictly below it. Thus,  $|P^-| + |P^+| + |P^0| = n'$ . By definition of  $t$ , all the points of  $P + \mathbf{t}$  lie above  $h$  and all the points of  $P - \mathbf{t}$  lie below  $h$ . Define a subset  $Q' \subseteq Q$  of size  $m$  as follows. For each  $p \in P^+$ , add  $p$  and  $p - \mathbf{t}$  to  $Q'$ . For each  $p \in P^-$ , add  $p$  and  $p + \mathbf{t}$  to  $Q'$ . Finally, for each  $p \in P^0$ , add  $p$ ,  $p - \mathbf{t}$ , and  $p + \mathbf{t}$  to  $Q'$ . Clearly,

$$|Q'| = 2(|P^+| + |P^-|) + 3|P^0| = 2(|P^+| + |P^-| + |P^0|) + |P^0| = 2n' + (d+1) = m.$$

If  $p \in P^+$  then the combined contribution of  $p$  and  $p - \mathbf{t}$  to  $\Delta_h^{(\text{LTA})}(Q, m)$  is  $|p - h| + |(p - \mathbf{t}) - h|$  which by Lemma 6(ii) is  $t$ . Symmetrically, if  $p \in P^-$ , the combined contribution of its two points is  $t$ . Finally, if  $p \in P^0$  its three points contribute a total of  $0 + t + t = 2t$ . Thus, the total LTA cost is  $t|P^+| + t|P^-| + 2t|P^0| = t|P| + t(d+1) = (n' + (d+1))t = \Delta$ , as desired.

To establish the “if” part of the claim we show that if no  $d+1$  points of  $P$  lie on a nonvertical hyperplane, then the LTA cost will strictly exceed  $\Delta$ . Let  $h$  be any nonvertical hyperplane. Let  $\Delta_h$  denote the sum of the  $d+1$  smallest absolute residuals of  $P$  with respect to  $h$ . By hypothesis, no  $d+1$  points of  $P$  lie on any hyperplane, and therefore  $\Delta_h > 0$ .

To complete the proof, it suffices to show that  $\Delta_h^{(\text{LTA})}(Q, m) \geq \Delta + \Delta_h$ , since it will then follow that  $\Delta_h^{(\text{LTA})}(Q, m) > \Delta$ . Let  $Q' \subseteq Q$  denote the  $m$  points having the smallest absolute residuals

<sup>2</sup>This condition is needed in the proof Theorem 5.3 of [7] as well, but may not hold in the construction presented there. Selecting  $t$  as we do here fixes this minor error.

with respect to  $h$ . We classify each point of  $P$  according to the number of points its three clones provide to  $Q'$ . More precisely, for  $1 \leq i \leq 3$ , define  $P_i$  to be the subset of points  $p \in P$  such that  $|\{p - \mathbf{t}, p, p + \mathbf{t}\} \cap Q'| = i$ . Let  $n_i = |P_i|$ . Clearly  $n_0 + n_1 + n_2 + n_3 = |P| = n'$ , and by summing their contribution to the size of  $Q'$  we have  $n_1 + 2n_2 + 3n_3 = |Q'| = 2n' + (d + 1)$ . Therefore the conditions of Lemma 7 hold (using  $n'$  in place of  $n$ ).

Let us consider the contribution of each of these subsets to the LTA cost for  $h$ .

- If  $p \in P_0$  or  $p \in P_1$ , we ignore its contribution to the LTA cost.
- If  $p \in P_2$ , we consider two cases depending on whether  $p$  lies above or below  $h$ . If  $p$  lies below  $h$ , the upper two copies of  $p$  will be the closest of  $p$  copies to  $h$ , and so their contribution will total  $|p - h| + |(p + t) - h|$ . By Lemma 6(i) this is at least  $t$ . On the other hand, if  $p$  lies above  $h$ , the lower two copies of  $p$  will be closest and will contribute a total of  $|p - h| + |(p - t) - h|$ . By Lemma 6 (ii) this is also at least  $t$ .
- If  $p \in P_3$ , then all three copies of  $p$  contribute, and so their contribution is  $|p - h| + |(p + t) - h| + |(p - t) - h|$ , which by Lemma 6 (iii) is at least  $2t + |p - h|$ .

Summing the contributions for all four cases and applying Lemma 7(i), it follows that the LTA cost of  $h$  is at least

$$\begin{aligned}
\sum_{p \in P_2} t + \sum_{p \in P_3} (2t + |p - h|) &\geq n_2 t + 2n_3 t + \sum_{p \in P_3} |p - h| \\
&\geq (n' + (d + 1))t + \sum_{p \in P_3} |p - h| \\
&= \Delta + \sum_{p \in P_3} |p - h|. \tag{2}
\end{aligned}$$

From Lemma 7(ii) and the fact that  $n_1 \geq 0$ , we have  $n_3 \geq d + 1$ . Therefore at least  $d + 1$  points of  $P_3$  contribute to the final summation term, and this final term is at least  $\Delta_h$ .  $\square$

We can now provide the proof of the main theorem.

**Proof:** (of Theorem 4) Our proof is based on two cases, depending on the relationship between  $m$  and  $n$ . For the first case, let us assume that  $m \leq (2n/3) + (d + 1)$ . We will show that solving LTA for  $n$  points and  $m$  inliers requires  $\Omega(m^d)$  time under the assumption of the hardness of affine degeneracy. Let  $n' = (m - (d + 1))/2$ . Since  $m - d$  is odd and  $m \geq d + 1$ ,  $n'$  is a nonnegative integer. Let  $P$  be a point set of size  $n'$  for which we want to solve the affine degeneracy problem. We construct an instance of LTA as follows. Let  $\hat{m} = 2n' + (d + 1)$ ,  $\hat{n} = 3n'$ , and  $k = n - \hat{n}$ . Observe that  $\hat{m}$  and  $\hat{n}$  satisfy the conditions on  $m$  and  $n$ , respectively, in the statement of Lemma 8, and therefore we may apply the construction of that lemma to  $P$ . The resulting number of inliers is

$$\hat{m} = 2n' + (d + 1) = 2((m - (d + 1))/2) + (d + 1) = m,$$

as desired. The resulting number of points may be too small, since

$$\hat{n} = 3n' = \frac{3(m - (d + 1))}{2} \leq \frac{3\left(\left(\frac{2n}{3} + (d + 1)\right) - (d + 1)\right)}{2} \leq n.$$

Therefore, to complete the construction, we generate  $k = n - \hat{n}$  points that are sufficiently far from all the others that they cannot affect the LTA solution. Thus, we have generated an instance of

LTA involving  $n$  points and  $m$  inliers, and by Lemma 8 the time needed to solve such an instance in the worst case is  $\Omega(\widehat{n}^d) = \Omega((n')^d) = \Omega(m^d)$ . This completes the first case.

For the second case, let us assume that  $m > (2n/3) + (d + 1)$ . We will show that solving LTA for  $n$  points and  $m$  inliers requires  $\Omega((n - m)^d)$  time under the assumption of the hardness of affine degeneracy. Let  $P$  be a point set of size  $n' = n - m + (d + 1)$  for which we want to solve the affine degeneracy problem. Let  $\widehat{m} = 2n' + (d + 1)$ ,  $\widehat{n} = 3n'$ , and  $k = 3m - 2n - 3(d + 1)$ . Note that by the lower bound on  $m$  and the fact that  $m - d$  is odd, it follows that  $k \geq 0$ , and  $k$  is even. As before,  $\widehat{m}$  and  $\widehat{n}$  satisfy the conditions of Lemma 8 in the roles of  $m$  and  $n$ , respectively, and therefore we may apply the construction of the lemma to  $P$ . We apply an additional modification to the construction. Recall the translation distance  $t$  used in the lemma. We generate an additional set of points  $R$  of size  $k/2$  densely clustered near  $P$ 's centroid (see Fig. 5). We add two copies of  $R$  to the construction, one translated vertically up by distance  $t$  and the other down by distance  $t$ . Because of their proximity to  $P$ , the points lying within the two copies of  $R$  will be taken as inliers for any hyperplane that passes through  $P$ . We then apply the same construction of Lemma 8, but using the translation distance of  $2t$  rather than  $t$  (see Fig. 5). (Note that the construction is correct for all sufficiently large values of  $t$ .)

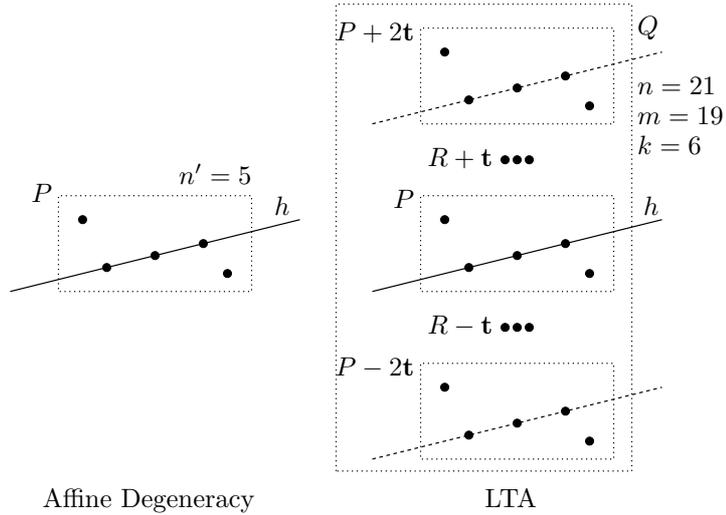


Fig. 5: Proof of the case  $m > (2n/3) + (d + 1)$  of Lemma 8. (Not drawn to scale.).

Observe that the total number of points in the resulting construction is

$$\widehat{n} + k = 3n' + (3m - 2n - 3(d + 1)) = 3(n - m + (d + 1)) + (3m - 2n - 3(d + 1)) = n,$$

as desired. The total number of inliers is

$$\begin{aligned} \widehat{m} + k &= (2n' + (d + 1)) + (3m - 2n - 3(d + 1)) \\ &= (2(n - m + (d + 1)) + (d + 1)) + (3m - 2n - 3(d + 1)) = m, \end{aligned}$$

as desired. Thus, by Lemma 8 the time needed to solve such an instance in the worst case is  $\Omega(\widehat{n}^d) = \Omega((n')^d) = \Omega((n - m)^d)$ . This completes the proof.  $\square$

Theorem 4 does not provide an interesting lower bound if  $m$  is very small. The following result shows that both the exact LTA and LTS problems are hard even for small values of  $m$ . The proof is a relatively straightforward adaptation of the analogous result in [7].

**Theorem 5** *Under the assumption of the hardness of affine degeneracy, computing either linear LTA or LTS estimator for a set of  $n$  points in  $\mathbb{Z}^d$  with  $m$  inliers requires  $\Omega((n/m)^d)$  time in the worst case.*

**Proof:** We show the result for LTS, and the LTA version is a straightforward modification. Suppose we are given a set  $P$  of  $n' = n(d+1)/m$  points in  $\mathbb{Z}^d$  for which we want to solve the affine degeneracy problem. In linear time, we compute an upper bound  $M$  on the absolute value of every coordinate. Let  $\delta = \delta'(d+1)/(mM^d)$  for some sufficiently small constant  $\delta' > 0$ , possibly depending on  $d$ . Let  $m' = m/(d+1)$ . We construct a new set  $Q$  consisting of  $m'$  copies of  $P$ , where the  $i$ th copy is shifted upward a distance of  $i\delta$ . The set consisting of the  $m'$  copies of each point of  $P$  is called a *group*.

If some hyperplane  $h$  contains  $d+1$  points of  $P$ , then let  $t = (m' - 1)\delta$ , and consider the slab bounded by  $h$  and  $h+t$ . Observe that this slab is high enough to contain the union of the associated groups, and so it contains at least  $m'(d+1) = m$  of the points of  $Q$ . The optimal LTS cost is not greater than the LTS cost of  $h$  which is at most

$$\Delta_0 \leq (d+1) \sum_{i=0}^{m'-1} (i\delta)^2 < m'(d+1)(m'\delta)^2 \leq m'(d+1) \left( \frac{\delta'}{M^d} \right)^2.$$

If, on the other hand,  $P$  is affinely independent then by Lemma 5(i) any slab containing  $d+1$  points of  $P$  has height at least  $\alpha/M^d$  for some constant  $\alpha$ . Let  $h$  be the optimal LTS hyperplane for  $Q$ , and let  $S$  be the narrowest slab centered at  $h$  that contains  $m$  points of  $Q$ . Since  $S$  contains only  $m'$  points from each individual group,  $S$  contains at least one point from each of the at least  $m/m' = d+1$  distinct groups. Since the height of each group is at most  $m'\delta$ , it follows that the height of  $S$  is at least

$$\frac{\alpha}{M^d} - m'\delta = \frac{\alpha}{M^d} - \frac{\delta'}{M^d} = \frac{\alpha - \delta'}{M^d}.$$

At least one of the  $d+1$  groups contributing to the LTS cost of  $h$  is at distance at least half the slab's height from  $h$ . Considering the contribution from this group alone, the LTS cost of  $h$  is at least

$$\Delta_1 \geq m' \left( \frac{\alpha - \delta'}{2M^d} \right)^2.$$

By setting  $\delta' < \alpha/(1 + (\sqrt{d+1}/2))$  it follows that  $\Delta_0 < \Delta_1$ . Thus, the optimum LTS cost is at most  $\Delta_0$  if and only if  $P$  is affinely degenerate.  $\square$

## 4.2 Hardness of Residual and Quantile Approximations

In this section we provide hardness results for both residual and quantile approximations, under the assumption of the hardness of affine degeneracy. Our result for the residual approximation holds for any approximation factor, and our result for quantile approximation holds for approximation factors smaller than  $1/(d-1)$  in dimension  $d$ . Both results hold for both LTS and LTA (through a small adjustment in the parameters), but we present proofs only for the LTS case. Both proofs are an adaptation of the proofs given in Erickson, *et al.* [7] for approximating LMS.

**Theorem 6** *Under the assumption of the hardness of affine degeneracy, for any  $\varepsilon_r > 0$  (depending possibly on  $n$  and  $m$ ), computing an  $\varepsilon_r$ -residual approximation to the linear LTS or LTA estimator for a set of  $n$  points and  $m$  inliers in  $\mathbb{Z}^d$  requires  $\Omega((n-m)^{d-1})$  time in the worst case.*

**Proof:** We prove the result for LTS, and the LTA version is a straightforward modification. Suppose we are given a set  $P$  of  $n' = n/2 - m/2 - d$  points on the integer lattice  $\mathbb{Z}^{d-1}$ . Let  $M$  be an upper bound on the maximum absolute value of any coordinate of  $P$ , and let  $\delta > 0$  be a sufficiently small constant to be defined later. We can compute these values in  $O(n')$  time.

In  $O(n)$  time, we construct a new set  $Q$  consisting of three subsets: (1) a copy of  $P$  on the vertical hyperplane  $x_1 = 1$ , (2) a copy of  $-P$  (the reflection of  $P$  through the origin) on the hyperplane  $x_1 = -1$ , and (2) a set of  $m - 2d$  points placed on a vertical line passing through the origin, half above and half below, such that their squared distances from the origin are all very nearly equal to  $\delta$ .

If any  $d$  points of  $P$  lie on a common  $(d-2)$ -flat, then by symmetry there is a nonvertical  $(d-1)$  hyperplane  $h$  that passes through this flat on the hyperplane  $x_1 = 1$ , its reflection on  $x_1 = -1$ , and the origin. The total contribution to the LTS cost of the  $2d$  points lying on the two vertical hyperplanes is zero, and the contribution of the  $m - 2d$  points near the origin is roughly  $(m - 2d)\delta$ . Thus the total LTS cost  $\Delta_0$  in this case is at most  $(m - 2d)\delta$ .

If, on the other hand the points of  $P$  are affinely independent, let  $\Delta_1$  denote the optimum LTS cost, and let  $h$  denote the hyperplane generating this cost. By applying Lemma 5(ii) in dimension  $d-1$  to the smallest  $d$  squared residuals with respect to the intersection of  $h$  with  $x_1 = 1$ , it follows that their contribution to the LTS cost is at least  $(d - (d-1))\alpha^2/M^{2d} = \alpha^2/M^{2d}$  for some positive  $\alpha$ . By selecting  $\delta < \alpha^2/((1 + \varepsilon_r)M^{2d}(m - 2d))$  we have

$$\frac{\Delta_1}{\Delta_0} \geq \frac{\alpha^2/M^{2d}}{(m - 2d)\delta} > 1 + \varepsilon_r.$$

Thus, by approximating  $\Delta^{(\text{LTS})}(Q, m)$  to within a factor of  $1 + \varepsilon_r$ , we can determine whether the original set  $P$  contains a degeneracy. Under the assumption of the hardness of affine degeneracy, this implies that this requires  $\Omega((n')^{d-1}) = \Omega((n - m)^{d-1})$  time in the worst case.  $\square$

**Theorem 7** *Under the assumption of the hardness of affine degeneracy, for any  $0 < \varepsilon_q < 1/(d-1)$ , computing an  $\varepsilon_q$ -quantile approximation to the linear LTS or LTA estimator for a set of  $n$  points and  $m$  inliers in  $\mathbb{Z}^d$  requires  $\Omega((n - m)^{d-1})$  time in the worst case.*

**Proof:** As before, we prove the result for LTS, and the LTA version is a straightforward modification. The construction of the set  $Q$  is essentially the same as the construction of Theorem 6, except the value of  $\delta$  (given below) will differ and the number of points on the vertical line through the origin is  $m$ , rather than  $m - 2d$ . Let  $m^- = \lceil m/(1 + \varepsilon_q) \rceil$ , denote the number of inliers in the quantile approximation.

Let  $(1/m)\Delta(Q, m)$  denote the minimum *average* squared residual for the points of  $Q$  assuming  $m$  inliers. Consider first that some  $d$  points of  $P$  lie on a common  $(d-2)$ -flat, and let  $h$  denote the hyperplane passing through the origin and these  $2d$  points on the vertical hyperplanes at  $x_1 = 1$  and  $x_1 = -1$ . By the same argument made in the proof of Theorem 6, the contribution to the sum of squared residuals by the  $2d$  points lying on the two vertical hyperplanes is zero, and the contribution of the  $m - 2d$  points near the origin is roughly  $(m - 2d)\delta$ . Thus we have

$$\frac{1}{m}\Delta(Q, m) \leq \frac{m - 2d}{m}\delta = \delta \left(1 - \frac{2d}{m}\right).$$

On the other hand, if the points of  $P$  are affinely independent, let us consider the minimum average LTS cost assuming  $m^-$  inliers,  $(1/m^-)\Delta(Q, m^-)$ . Let  $h$  denote the hyperplane generating

this minimum cost. As in Theorem 6, we may apply Lemma 5(ii), and by making  $\delta$  sufficiently small, we may assume that all but  $2(d-1)$  of the  $m^-$  closest squared residuals to  $h$  arise from the points that are close to the origin. (Recall that there are at least  $m \geq m^-$  such points.) The contribution of these points to the sum of squared residuals is at least roughly  $(m^- - 2(d-1))\delta$ , and so we have

$$\frac{1}{m^-} \Delta(Q, m^-) \geq \frac{m^- - 2(d-1)}{m^-} \delta = \delta \left( 1 - \frac{2(d-1)}{m^-} \right).$$

Let us assume that  $m$  is sufficiently large so that the effect of the ceiling is negligible. Thus,  $m^- = m/(1 + \varepsilon_q)$ . By our hypothesis that  $\varepsilon_q < 1/(d-1)$  for all sufficiently large  $m$  we have

$$\begin{aligned} \frac{1}{m^-} \Delta(Q, m^-) &\geq \delta \left( 1 - \frac{2(d-1)}{m^-} \right) \geq \delta \left( 1 - \frac{2(d-1)}{m/(1 + \varepsilon_q)} \right) \\ &> \delta \left( 1 - \frac{2d}{m} \right) \geq \frac{1}{m} \Delta(Q, m). \end{aligned}$$

Thus, by computing a  $\varepsilon_q$ -quantile approximation to  $\Delta^{(\text{LTS})}(Q, m)$ , we can determine whether the original set  $P$  contains an affine degeneracy. Under the assumption of the hardness of affine degeneracy, this implies that this requires  $\Omega((n')^{d-1}) = \Omega((n-m)^{d-1})$  time in the worst case.  $\square$

## 5 Conclusions

We have presented a number of results on the computational complexity of the linear LTS and LTA estimators, including both exact and approximate formulations. We showed that the LTS estimator for a set of  $n$  points can be computed in  $O(n^2)$  time in the plane and in  $O(n^{d+1})$  time for dimension  $d \geq 3$ . Both algorithms use linear space. Under the (typical) assumption that the number of inliers  $m$  is  $\Theta(n)$ , we presented a randomized  $\varepsilon$ -residual approximation algorithm for LTS, whose running time is  $O(n^{d-1}(1/\varepsilon)^d \text{polylog } n)$ . Under the assumption of the hardness of affine degeneracy, we also presented a number of lower bound results. Assuming that  $m = \Theta(n)$ , we presented an  $\Omega(n^d)$  lower bound for the exact LTA problem, and we presented lower bounds of the form  $\Omega(n^{d-1})$  for both residual and quantile approximations to both LTS and LMS.

There are still a number of open problems that remain. Given the  $O(n^d)$  complexity of LMS [7], it is natural to conjecture that it should be possible to compute the LTS estimator in  $O(n^d)$  time. Computing LTA seems to be harder. The incremental update step used in our LTS algorithm exploits properties of least-squares solutions, which do not apply for minimizing sums of absolute residuals. In contrast, our lower bound for the exact LTA problem exploits an algebraic cancellation, which works for absolute residuals but does not apply to the squared residuals used in the LTS problem. It is natural to conjecture that the same lower bound holds for LTS as well.

## References

- [1] N. Alon and J. Spencer. *The Probabilistic Method*. J. Wiley and Sons, New York, NY, 2000.
- [2] T. M. Chan. Geometric applications of a randomized optimization technique. *Discrete Comput. Geom.*, 22:547–567, 1999.

- [3] S. D. Conte and C. W. De Boor. *Elementary Numerical Analysis: An Algorithmic Approach*. McGraw-Hill, New York, 3rd edition, 1980.
- [4] H. Edelsbrunner. *Algorithms in Combinatorial Geometry*, volume 10 of *EATCS Monographs on Theoretical Computer Science*. Springer-Verlag, Heidelberg, West Germany, 1987.
- [5] H. Edelsbrunner and D. L. Souvaine. Computing median-of-squares regression lines and guided topological sweep. *J. Amer. Statist. Assoc.*, 85:115–119, 1990.
- [6] J. Erickson. Lower bounds for linear satisfiability problems. *Chicago J. Comput. Sci.*, 1999(8), 1999.
- [7] J. Erickson, S. Har-Peled, and D. M. Mount. On the least median square problem. *Discrete Comput. Geom.*, 36:593–607, 2006.
- [8] J. Erickson and R. Seidel. Better lower bounds on detecting affine and spherical degeneracies. *Discrete Comput. Geom.*, 13:41–57, 1995.
- [9] W. Feller. *An Introduction to Probability Theory and its Applications*. John Wiley & Sons, New York, NY, 3rd edition, 1968.
- [10] S. Har-Peled. How to get close to the median shape. In *Proc. 22nd Annu. ACM Sympos. Comput. Geom.*, pages 402–410, 2006.
- [11] D. M. Hawkins. The feasible solution algorithm for least trimmed squares regression. *Comput. Statist. Data Anal.*, 17:185–196, 1994.
- [12] O. Hössjer. Rank-based estimates in the linear model with high breakdown point. *J. Amer. Statist. Assoc.*, 89:149–158, 1994.
- [13] O. Hössjer. Exact computation of the least trimmed squares estimate in simple linear regression. *Comput. Statist. Data Anal.*, 19:265–282, 1995.
- [14] P. J. Rousseeuw. Least median-of-squares regression. *J. Amer. Statist. Assoc.*, 79:871–880, 1984.
- [15] P. J. Rousseeuw and A. M. Leroy. *Robust Regression and Outlier Detection*. Wiley, New York, 1987.
- [16] P. J. Rousseeuw and K. van Driessen. Computing LTS regression for large data sets. *Data Min. Knowl. Discov.*, 12:29–45, 2006.
- [17] D. L. Souvaine and J. M. Steele. Time- and space- efficient algorithms for least median of squares regression. *J. Amer. Statist. Assoc.*, 82:794–801, 1987.