Algorithmica (2014) 69:148–183
DOI 10.1007/s00453-012-9721-8

On the Least Trimmed Squares Estimator

David M. Mount · Nathan S. Netanyahu · Christine D. Piatko · Ruth Silverman · Angela Y. Wu

Received: 12 December 2011 / Accepted: 18 November 2012 / Published online: 4 December 2012

Abstract The linear least trimmed squares (LTS) estimator is a statistical technique for fitting a linear model to a set of points. Given a set of n points in $\mathbb{R}^d$ and given an integer trimming parameter $h \leq n$, LTS involves computing the $(d-1)$-dimensional hyperplane that minimizes the sum of the smallest h 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

Dedicated to the memory of our dear friend and longtime colleague, Ruth Silverman.

D.M. Mount is partially supported by NSF grant CCF-1117259 and ONR grant N00014-08-1-1015.

D.M. Mount (✉)
Department of Computer Science, University of Maryland, College Park, MD, USA
e-mail: mount@cs.umd.edu

N.S. Netanyahu
Department of Computer Science, Bar-Ilan University, Ramat-Gan 52900, Israel
e-mail: nathan@cs.biu.ac.il

N.S. Netanyahu · R. Silverman
Center for Automation Research, University of Maryland, College Park, MD, USA

N.S. Netanyahu
e-mail: nathan@cfar.umd.edu

C.D. Piatko
The Johns Hopkins University Applied Physics Laboratory, Laurel, MD, USA
e-mail: christine.piatko@jhuapl.edu

A.Y. Wu
Department of Computer Science, American University, Washington, DC, USA
e-mail: awu@american.edu

statistically more efficient than LMS. Unfortunately, the computational complexity of LTS is less 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. A number of our results apply to the LTA estimator as well.

Keywords Robust estimation · Linear estimation · Least trimmed squares estimator · Approximation algorithms · Lower bounds

1 Introduction

In standard *linear regression* (with intercept), an n-element point set $P = \{p_1, \ldots, p_n\}$ is given, where each point consists of some number of independent variables and one dependent variable. Letting d denote the total number of variables, we wish to express the dependent variable as a linear function of $d-1$ independent variables. More formally, for $1 \le i \le n$, let $p_i = (x_{i,1}, \ldots, x_{i,d-1}, y_i) \in \mathbb{R}^d$. The objective is to compute a $(d-1)$-dimensional hyperplane, represented as a coefficient vector $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_d) \in \mathbb{R}^d$ so that

$$y_i = \sum_{j=1}^{d-1} \beta_j x_{i,j} + \beta_d + e_i, \quad \text{for } i = 1, 2, \ldots, n,$$

where the e_i's are the errors. Given such a vector $\boldsymbol{\beta}$, the i*th residual*, denoted by $r_i(\boldsymbol{\beta}, P)$, is defined to be $y_i - (\sum_{j=1}^{d-1} \beta_j x_{i,j} + \beta_d)$. Let $r_{[i]}(\boldsymbol{\beta}, P)$ denote the ith smallest residual in terms of absolute value. Throughout, we consider the y-coordinate axis to be the *vertical* direction, and so the ith residual is just the signed vertical distance from the hyperplane to p_i.

The standard least squares linear estimator is the hyperplane that minimizes the sum of the squared residuals. It is well known that the least squares estimator is very 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 arbitrarily.

The study of efficient algorithms for robust statistical estimators has been an active area of research in computational geometry. The most widely studied robust linear estimator is Rousseeuw's *least median of squares estimator* (LMS) [28], which is defined to be the hyperplane that minimizes the median squared residual. (In general, an integer trimming parameter h is given, and the objective is to find the hyperplane that minimizes the hth smallest squared residual.) A number of papers, both practical and theoretical, have been devoted to solving this problem in the plane and in higher dimensions [3, 11, 13, 24, 26, 27, 32]. Another example of a well-studied robust estimator in computational geometry is the Tukey median and the related topic of regression depth [4, 5, 21, 23].

Although the vast majority of work on robust linear estimation in the field of computational geometry has been devoted to the study of the LMS estimator, it has been observed by Rousseeuw and Leroy [29] that LMS is not the estimator of choice from the perspective of statistical properties. They argue that a better choice is the *least* *trimmed squares* (or LTS) linear estimator [28]. Given an n-element point set P and a positive integer *trimming parameter* $h \le n$, this estimator is defined to be the non-vertical hyperplane that minimizes the *sum* (as opposed to the maximum) of the h smallest squared residuals. More formally, given a trimming parameter h, the *LTS cost* of a hyperplane $\boldsymbol{\beta}$ is defined to be

$$\Delta_{\boldsymbol{\beta}}^{(\mathrm{LTS})}(P, h) = \sum_{i=1}^{h} r_{[i]}^2(\boldsymbol{\beta}, P).$$

The *LTS estimator* is a $(d-1)$-dimensional hyperplane $\boldsymbol{\beta}$ of minimum LTS cost, which we denote by $\boldsymbol{\beta}^{(\mathrm{LTS})}(P, h)$. We let $\Delta^{(\mathrm{LTS})}(P, h)$ denote the associated LTS cost of this hyperplane. The *LTS problem* is that of computing this hyperplane. The points having the h smallest squared residuals are called *inliers*, and the remaining points are *outliers*. Note that, when $h = n$, this is equivalent to the standard least squares estimator. Typically, h is set to some constant fraction of n based on the expected number of outliers. To guarantee a unique solution it is often assumed that h is at least $n/2$.

The LTS estimator has the same breakdown point as LMS. Like LMS, the LTS estimator is regression-, scale-, and affine-equivariant, which means that the estimator transforms "properly" under these types of transformations [29]. However, LTS has a number of advantages over LMS. The LTS objective function is smoother than that of LMS. LTS has better statistical efficiency because it is asymptotically normal [28]. Intuitively, LTS converges faster because the LMS estimator is influenced by just the $d+1$ inliers that have the largest squared residuals—the remaining inliers play essentially no role in the estimator's value. In contrast, the LTS objective function is influenced by all the inliers. Rouseeuw and van Driessen [30] remark that, for these reasons, LTS is more suitable as a starting point for two-step robust estimators, such as the MM-estimator [33] and generalized M-estimators [6, 31].

To date, the computational complexity of LTS is less well understood than that of LMS. Hössjer [20] presented an exact $O(n^2 \log n)$ algorithm for LTS in the plane based on plane sweep. The most practical approach is the Fast-LTS heuristic of Rousseeuw and Van Driessen [30], which is based on a combination of random sampling and local improvement. In practice this approach works well, but it provides no assurance of the quality of the resulting fit. Another example of a local search heuristic is Hawkins' feasible point algorithm [18], but it does not offer any formal performance guarantees either.

As we shall see, the best known algorithms for computing the LTS estimator have relatively high computational complexity, and so it is natural to consider whether this problem can be solved approximately. There are a few possible ways to formulate LTS as an approximation problem, either by approximating the residual, by approximating the quantile, or both. We introduce the approximation parameters ε_r and ε_q to denote the allowed residual and quantile errors, respectively.

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

$$\Delta_{\boldsymbol{\beta}}^{(\mathrm{LTS})}(P, h) \le (1 + \varepsilon_r)\Delta^{(\mathrm{LTS})}(P, h).$$

Quantile Approximation: As we shall see, much of the complexity of LTS arises because of the requirement that *exactly h* points be considered. We can relax this requirement by introducing a parameter $0 < \varepsilon_q < h/n$ and requiring that the fraction of inliers used is smaller by ε_q. Let $h^- = h - \lfloor n\varepsilon_q \rfloor$. An *$\varepsilon_q$-quantile approximation* is any hyperplane $\boldsymbol{\beta}$ such that

$$\frac{1}{h^-}\Delta_{\boldsymbol{\beta}}^{(\mathrm{LTS})}\left(P, h^-\right) \le \frac{1}{h}\Delta^{(\mathrm{LTS})}(P, h).$$

(The normalizing factors $1/h^-$ and $1/h$ are provided since the costs involve sums over a different number of terms.)

Hybrid Approximation: The above approximations can be merged into a single approximation. Given ε_r and ε_q as in the previous two approximations, let h^- be as defined above. An *$(\varepsilon_r, \varepsilon_q)$-hybrid approximation* is any hyperplane $\boldsymbol{\beta}$ such that

$$\frac{1}{h^-}\Delta_{\boldsymbol{\beta}}^{(\mathrm{LTS})}\left(P, h^-\right) \le (1+\varepsilon_r)\frac{1}{h}\Delta^{(\mathrm{LTS})}(P, h).$$

LTS and LMS are robust versions of the well known least squares (L_2) and Chebyshev (L_∞) 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 h smallest *absolute values* of the residuals. From the perspective of optimization problems, the associated cost functions are:

$$\Delta_{\boldsymbol{\beta}}^{(\mathrm{LTS})} = \sum_{i=1}^{h} r_{[i]}^2(\boldsymbol{\beta}), \qquad \Delta_{\boldsymbol{\beta}}^{(\mathrm{LMS})} = \max_{1\le i\le h} r_{[i]}^2(\boldsymbol{\beta}), \quad \text{and} \quad \Delta_{\boldsymbol{\beta}}^{(\mathrm{LTA})} = \sum_{i=1}^{h} \left|r_{[i]}(\boldsymbol{\beta})\right|.$$

By analogy, approximations can be defined for the other trimmed estimators, LMS and LTA. Har-Peled [17] gave an efficient approximation algorithm for the nonrobust point L_1 estimator. Fonseca [15] presented a constant-factor approximation generalization of LMS, where the objective is to fit a flat of a given dimension k.

In this paper, we present a number of results, both exact and approximate, for the LTS and LTA estimators. (See Table 1 for a summary.) In Sect. 2 we give two exact algorithms. The first is an $O(n^2)$ time exact algorithm for both estimators in the plane (see Theorem 1), which provides a modest improvement over the $O(n^2 \log n)$ algorithm due to Hössjer [20]. The second is an $O(n^{d+1})$ time exact algorithm for both estimators in dimension $d \ge 3$. Recalling that h is typically a constant fraction of n, this is a significant improvement over brute-force search, which runs in $O(hn^{d+1})$ time (see Theorem 2). Both algorithms use $O(n)$ space. In Sect. 3 we present a randomized ε-residual approximation algorithm, which works for both LTS and LTA. Throughout, we make the relatively weak assumption that $1/\varepsilon$ is bounded above by some polynomial function of n. Under this assumption, we show that the running time[1] of this algorithm is $\widetilde{O}((n^d/h)(1/\varepsilon)^d)$ (see Theorem 3). When $h = \Theta(n)$ and ε

[1]We use the asymptotic forms $\widetilde{O}$ and $\widetilde{\Omega}$ as a shorthand for O and Ω, respectively, where factors of the form $\log^{O(1)} n$ have been ignored.

Table 1 Summary of results for LTS and LTA given n points in $\mathbb{R}^d$ and h inliers. With the exception of the $\Omega(\min(h, n-h)^d)$ lower bound, which applies only to LTA, all results apply to both LTS and LTA

	Upper bound time/space	Lower bound	Theorems
Exact			
	$d=2: O(n^2)/O(n)$ $d \geq 3: O(n^{d+1})/O(n)$	$\Omega(\min(h, n-h)^d)$ [LTA] $\Omega((n/h)^d)$	1, 2, 4, 5
Approx.			
ε_r-Residual	$\widetilde{O}((n^d/h)(1/\varepsilon_r{}^d))/\widetilde{O}((n/\varepsilon_r)^{d-1})$	$\Omega((n-h)^{d-1})$	3, 6
ε_q-Quantile	–	$\Omega((n-h)^{d-1})$	7
$(\varepsilon_r, \varepsilon_q)$-Hybrid	$\widetilde{O}(n + 1/\varepsilon_m^{2(d+1)})/O(n)$ where $\varepsilon_m = \min(\varepsilon_q, \varepsilon_r h/n)$	–	8

is a constant, this is $\widetilde{O}(n^{d-1})$. Thus, it is asymptotically superior to our exact algorithm by almost a quadratic factor when $d \geq 3$.

In Sect. 4 we give a number of hardness results for both LTS and LTA under the assumption of the hardness of solving the affine degeneracy problem. (Affine degeneracy is the problem of determining whether any $d+1$ points of a given n-element point set in $\mathbb{R}^d$ lie on a common hyperplane [14]). In the case where $h = \Theta(n)$, we provide a lower bound of $\Omega(n^d)$ for computing the exact LTA estimator (see Theorem 4). This lower bound exploits the linear nature of the LTA objective function, and does not seem to generalize to LTS. Remarkably, as h becomes very small, the problem is not easier to solve. (Perhaps because when there are very few inliers, it is harder to identify them.) In particular, for the case where $h = O(1)$, we provide a lower bound of $\Omega(n^d)$ for both LTS and LTA (see Theorem 5).

We also provide lower bounds for approximate versions of both problems. We present an $\Omega(n^{d-1})$ lower bound for computing residual and quantile approximations to both LTS and LTA (see Theorems 6 and 7). Our lower bound for the quantile approximation holds under the assumption that $\varepsilon_q < h/(dn)$. Under the hypothesis of the hardness of affine degeneracy, these results imply that, if $h = \Theta(n)$ and ε is a constant, our exact algorithm is suboptimal by a factor of at most $O(n)$, and our approximation algorithm is suboptimal by only a polylogarithmic factor in n.

These results suggest that computing LTS and LTA, even approximately, is computationally intensive except in very low dimensions. On the positive side, we show that hybrid approximations can be computed efficiently in any fixed dimension. In Sect. 5, we show that, for fixed ε_r and ε_q, it is possible to compute an $(\varepsilon_r, \varepsilon_q)$-hybrid approximation in $O(n)$ time with high probability. (The general bound for arbitrary ε_r and ε_q is given in Theorem 8.) The approach is based on computing an exact solution for a sufficiently large random sample of the points.

Ours are the first nontrivial results on the asymptotic computational complexity of the LTS and LTA problems, which are both of practical importance in computational statistics. The most closely related work to ours from the perspective of techniques is the paper by Erickson, Har-Peled, and Mount [13] on the LMS estimator. They presented both exact and approximation algorithms for LMS in $\mathbb{R}^d$ and proved hardness results. Many of the results of this paper arise by an adaptation of the methods presented there, but, as we shall see, many new ideas are introduced to deal with the additional complexities of the LTS and LTA problems.

In all of our algorithms, we assume that the inputs are in general position. (We mean this is the strong sense, forbidding degenerate configurations not only in the input itself, but also in the intermediate structures arising in the course of our algorithms, whenever such configurations can be avoided by an infinitesimal perturbation of the original coordinates.) Throughout, we assume that the dimension d is a constant, but we treat the parameters h, ε_r and ε_q as asymptotic quantities.

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. This algorithm is a modification of the topological plane-sweep algorithm for LMS given by Edelsbrunner and Souvaine [11], which was based on the earlier plane sweep algorithm by Souvaine and Steiger [32]. 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 $h \leq n$. 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. 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 the points of P that lie on the slab's boundary.

Given any hyperplane $\boldsymbol{\beta}$, there exists a slab centered at $\boldsymbol{\beta}$ such that the associated slab set consist of the points having the h smallest squared residuals with respect to $\boldsymbol{\beta}$ (see Fig. 1(a)). 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. It is easy to prove by a perturbation argument that every slab set is the slab set for some critical slab (see Fig. 1(b)). Assuming general position, any set of $d+1$ points determines up to 2^{d+1} critical slabs, depending on how the points are assigned to the top and bottom of the slab. (The exact number is smaller, since, for example, all the points cannot be on the same side of the slab.) Thus, there are at most $2^{d+1}\binom{n}{d+1} = O(n^{d+1})$ distinct slab sets.

A simple brute-force solution to the LTS problem would involve enumerating all critical slab sets, eliminating all those that do not contain h points, computing the LTS cost of each of the remaining slab sets, and returning the minimum. The resulting algorithm would run in $\Theta(hn^{d+1})$ time. In a typical setting, where $h = \Theta(n)$, this is $\Theta(n^{d+2})$. Our exact algorithms improve upon this approach by reducing

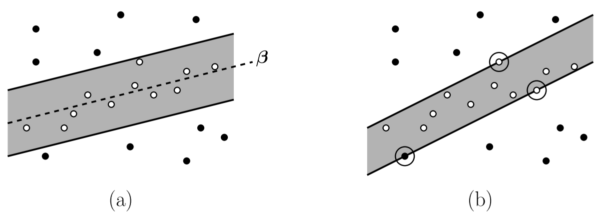

Fig. 1 The exact LTS algorithm: (**a**) the slab set consisting of the $h = 11$ points having the smallest squared residual with respect to $\boldsymbol{\beta}$, (**b**) an equivalent slab set generated by a critical slab

the enumeration of slab sets to the traversal of a dual hyperplane arrangement together with an efficient procedure for updating the LTS cost. In particular, given a point $p = (p_1, \dots, p_d)$, let the dual hyperplane in $\mathbb{R}^d$ be given by the equation $y = \sum_{i=1}^{d-1} p_i x_i - p_d$. It is easily verified that a slab of vertical height x is mapped through the dual transformation to a vertical segment of length x (see, e.g., [11]). The points lying within the slab correspond to dual hyperplanes that intersect this segment. The dual of a critical slab is a vertical line segment whose endpoints are incident to a total of $d+1$ dual hyperplanes of P. We call this a *critical segment*.

Our planar LTS algorithm is based on an adaptation of the topological plane sweep algorithm of Edelsbrunner and Souvaine. The principal new element is the need to efficiently maintain the LTS cost. 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 h, the linear LTS and LTA estimators can be computed in $O(n^2)$ time and $O(n)$ space.*

Proof We begin by describing the LTS solution. The algorithm is based on topological plane sweep of the dual line arrangement [10, 11]. For each i, where $1 \leq i \leq n-h+1$, consider any vertical line segment joining levels i and $i+h-1$ of the arrangement. (See [9] for terminology.) Let $\mathcal{L}_i$ and $\mathcal{L}_{i+h-1}$ denote these levels. Such a segment intersects at least h dual hyperplanes and so corresponds to a slab that contains at least h points of P. It is easy to show that there are $O(n^2)$ such critical segments [11]. These critical segments can be generated by topological plane sweep in $O(n^2)$ time and $O(n)$ space.

Consider a fixed pair of levels $\mathcal{L}_i$ and $\mathcal{L}_{i+h-1}$ and consider the left-to-right sequence of critical segments joining these two levels (shown as broken vertical segments of Fig. 2(a)). The LMS algorithm of [11] 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. 2(a)) all the vertical segments connecting $\mathcal{L}_i$ and $\mathcal{L}_{i+h-1}$ 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.

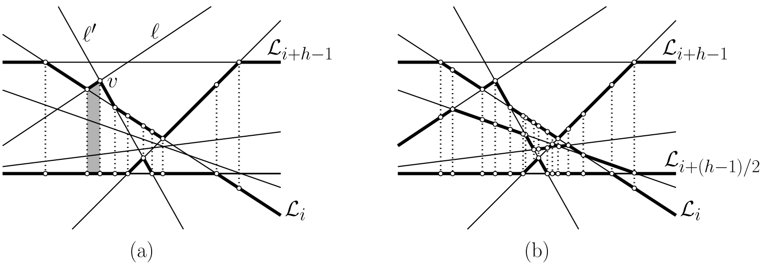

Fig. 2 Proof of Theorem 1: (**a**) updating a critical segment for LTS between levels $\mathcal{L}_i$ and $\mathcal{L}_{i+h-1}$, (**b**) the critical segments for LTA

Solving the LTS problem reduces therefore to solving the standard least squares problem for each of these slab sets.

It is well known that the solution to least squares for a set of points $p_i = (x_i, y_i)$ in the plane involves computing formulas whose terms are $\sum_i x_i$, $\sum_i y_i$, $\sum_i x_i^2$, $\sum_i y_i^2$, and $\sum_i x_i y_i$ (see, e.g., [7]). The algorithm maintains these sums for the current set of h inliers. As the plane sweep proceeds from one event to the next, only a constant number of points are inserted or deleted from the slab set. Therefore it is possible to incrementally update each of the above sums by adding or subtracting a constant number of quantities, and reevaluate the least squares solution in constant time. When the algorithm terminates, we have computed the cost of the standard least squares estimator for all $O(n^2)$ of the slab sets in $O(1)$ amortized time per set. The overall minimum is returned as the final answer. The space needed is $O(n)$.

The LTA algorithm has the same general structure. We will describe the case where h is odd, since the solution is more symmetric. It is well known (and easily proved by a variation argument) that, given a set of h numbers, the median of the set minimizes the sum of absolute distances to the other elements. As the topological plane sweep proceeds from left to right, for $1 \leq i \leq n - h + 1$, we maintain at each vertex a structure that consists of a vertical line segment joining levels i, $i + h - 1$, and $i + (h - 1)/2$. (See the dotted segments of Fig. 2(b).) As above, the first and second levels define the extremes of the critical segment and the third level is the median. Because each vertex of the arrangement participates in a constant number of such critical segments (playing the role of either the top, bottom, or center level), the total number of critical segments is $O(n^2)$. Between two critical segments, the same set of $(h - 1)/2$ segments pass above the median level and the same set of $(h - 1)/2$ below it. The LTA cost defined by these two sets can be expressed as the sum of linear functions. As we move from one critical segment to the next, a constant number of terms enter and leave this linear function. Therefore, in $O(1)$ time per critical segment, we can update the LTA cost. As above, after processing all $O(n^2)$ events, we return the overall minimum. The total space is $O(n)$. □

The planar algorithm makes use of the fact that, in the plane, there is an efficient way to enumerate the slab sets containing h points. Unfortunately, we know of no comparably efficient way to do this in higher dimensions. The following result shows that LTS can be solved in higher dimensions, but it is relatively less efficient for this reason. The general approach is to enumerate all $(d - 1)$-element subsets of P, and for each subset, to consider the restricted set of slabs that have the property that the points of this subset all lie on the slab's boundary. Since $d + 1$ points are needed to uniquely determine a slab, this restriction reduces the number of degrees of freedom to $(d + 1) - (d - 1) = 2$. The resulting restricted problem is thus reduced to a 2-dimensional problem, which we show can be solved by 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 h, the linear LTS and LTA estimators can be computed in $O(n^{d+1})$ time and $O(n)$ space.*

Proof We present only the LTS solution, and note that the LTA solution is a direct modification. 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 h points, and prune away those that contain an improper number of points. The algorithm enumerates all $(d-1)$-element subsets of points of P. For each subset R we define the *R-restricted slab set* to be those slabs such that the points of R all lie on the slab's boundary. There are at most 2^{d-1} ways of assigning the points of R to the upper and lower bounding hyperplanes of each slab (and, as mentioned earlier, we need only consider those with at least one point on each side). The resulting number of subsets along with upper-lower assignments is $O(2^{d-1}n^{d-1})$, which is $O(n^{d-1})$ given our assumption that d is fixed.

For each such subset R and each upper-lower assignment, we define the *R-restricted LTS problem* to be that of computing the least squares cost of the points lying within some R-restricted slab, subject to the constraint that the slab contains exactly h points of P. Our algorithm generates and solves all the restricted problems and returns the minimum among all of them. To complete the proof, it suffices to show how each restricted problem can be solved in $O(n^2)$ time and $O(n)$ space, which will yield the desired total time of $O(n^2 \cdot n^{d-1}) = O(n^{d+1})$ and space of $O(n)$.

Let us begin by defining a dual space of slabs. Consider a slab S, where $(\beta_1, \ldots, \beta_{d-1}, \beta_d^-)$ are the coefficients of the hyperplane bounding S's lower side, and $(\beta_1, \ldots, \beta_{d-1}, \beta_d^+)$ are the coefficients bounding the upper side. The condition that $p = (p_1, \ldots, p_d)$ lies on the boundary of S is a linear equality involving the coordinates of p, and thus the condition that p lies within the slab can be expressed as two linear inequalities involving the coordinates of p:

$$\beta_d^- \le p_d - \sum_{i=1}^{d-1} \beta_i p_i \le \beta_d^+.$$

We can therefore interpret a slab S as a point $\boldsymbol{\beta}_S = (\beta_1, \ldots, \beta_{d-1}, \beta_d^-, \beta_d^+)$ in a $(d+1)$-dimensional *slab space*. Given a point p, the set of slabs S that contain p "dualizes" to the set of points $\boldsymbol{\beta}_S$ in slab space such that

$$\beta_d^- + \sum_{i=1}^{d-1} \beta_i p_i \le p_d \le \beta_d^+ + \sum_{i=1}^{d-1} \beta_i p_i.$$

For a given point p, the set of points $\boldsymbol{\beta}$ satisfying these two linear inequalities is clearly the intersection of two (nonparallel) halfspaces in the $(d+1)$-dimensional slab space.

Consider any $(d-1)$-element subset of P and an assignment of these points to the upper and lower sides of a slab. Let R denote the associated restricted problem. It follows directly from the above remarks that any R-restricted slab S dualizes to a point $\boldsymbol{\beta}_S \in \mathbb{R}^{d+1}$ satisfying a system of $d-1$ linear equations, one for each point of R. By our general-position assumption, the set of points satisfying such a system is a 2-dimensional affine subspace (that is, a plane) in slab space, which we denote by F_R. Through Gaussian elimination, we may project this subspace to any coordinate plane spanned by two coordinate axes. This restricted problem is thus reduced to a search problem in $\mathbb{R}^2$, where each point of the plane corresponds uniquely to an R-restricted slab.

Consider any point $p \in P$. By our earlier remarks, a slab S contains p if and only if its dual slab $\boldsymbol{\beta}_S$ lies within a generalized wedge formed by the intersection of two halfspaces in $\mathbb{R}^{d+1}$. By intersecting this generalized wedge with F_R, the resulting 2-dimensional wedge contains the duals of the R-restricted slabs containing p. By the aforementioned projection, we can identify each point $p \in P$ with a pair of halfspaces in $\mathbb{R}^2$, such that the restricted slabs containing p correspond exactly to the points lying in the intersection of these two halfspaces.

If we apply this observation to every point of P, the result is an arrangement of $2n$ lines in $\mathbb{R}^2$, such that each face of this arrangement corresponds to the set of R-restricted slabs that contain the same subset of P. Such a face is *admissible* if this number of points is h. (Points lying on the boundary of a slab may be interpreted as lying inside or outside. We admit the face if it is possible to classify the boundary points so that the final count is h.) The vertices of this arrangement correspond to critical slabs. 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 in $O(n^2)$ time and $O(n)$ space. As we move from one face of the arrangement to the next, we incrementally update the solution of the associated least squares problem in $O(1)$ amortized time (see the proof of Theorem 1). Only those faces corresponding to admissible slabs are retained for final consideration. Thus, each restricted problem can be solved in $O(n^2)$ time and $O(n)$ space, which completes the proof. □

3 Residual Approximation

The results of the previous section do not provide a very efficient solution to LTS, unless the dimension is low. This raises the question of whether we can do better through approximation. In this section we present a residual approximation algorithm for LTS. The algorithm is randomized and runs in $\widetilde{O}((n^d/h)(1/\varepsilon)^d)$ time with high probability. The algorithm is based on the general framework for the LMS approximation presented by Erickson et al. [13]. 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 h points. The solution to this decision problem is then incorporated into a parametric search algorithm in order to determine the approximately optimal height.

The principal difficulty in adapting the algorithm of [13] 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 single slab of fixed height, as we could with the simpler LMS problem. This is because the cost depends on the residuals of *all* of the inliers. Our approach is to replace the single-slab method of [13] with a sequence of $O((1/\varepsilon)\log(n/\varepsilon))$ parallel slabs radiating outwards from the central hyperplane, which we call a slab system. The slab heights 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 aggregate and approximate their combined contributions to the LTS cost. The remainder of this section will be devoted to deriving this algorithm, whose performance is summarized in the following theorem.

Theorem 3 *Given an n-element set of points P in $\mathbb{R}^d$, a trimming parameter h, and an approximation parameter $0 < \varepsilon < 1$, it is possible to compute an ε-residual approximation to the linear LTS estimator, with high probability, in time $O((n^d/h)(1/\varepsilon^d)\log^{d+2} n)$ and space $O((n/\varepsilon)^{d-1}\log^{d-1} n)$.*

The presentation of the algorithm consists of three building blocks, which are given in the next few subsections. In Sect. 3.1 we introduce slab systems and present their basic properties. In Sect. 3.2 we present a restricted form of the problem and show that combinatorially distinct slab systems for the restricted problem can be represented as a hyperplane arrangement in $\mathbb{R}^d$. Finally, in Sect. 3.3 we present the complete algorithm.

3.1 Slab Systems

Consider an instance of the LTS problem consisting of a point set P, a trimming parameter h, and approximation parameter $\varepsilon < 1$. (It will be straightforward to modify our results to handle any fixed approximation parameter.) As stated earlier, we assume that $1/\varepsilon$ is bounded above by a polynomial in n, so $\log(1/\varepsilon)$ can be absorbed in the polylogarithmic term when stating complexity bounds. Suppose that we are given a real positive parameter Δ, called the *cost estimate*. Given such an instance and a hyperplane $\boldsymbol{\beta}$, we define a (Δ, ε)-*slab system* for $\boldsymbol{\beta}$ as follows. For an integer f (to be specified below), and for $-f \leq j \leq f$, define a sequence $\langle a_j \rangle$ of nonnegative reals whose squared values satisfy

$$a_j^2 = \left(1 + \frac{\varepsilon}{2}\right)^j \varepsilon \frac{\Delta}{h}.$$

(Because LTS is defined in terms of squared residuals, it will simplify matter to deal with squared distances.) We define $f(h,\varepsilon)$ to be the smallest integer such that $(1 + \varepsilon/2)^{f(h,\varepsilon)} \geq 2h/\varepsilon$, or equivalently $f(h,\varepsilon) = \lceil \ln(2h/\varepsilon)/\ln(1+\varepsilon/2) \rceil$. When h and ε are clear, we refer to this quantity as f. It is easy to verify that, for $0 < \varepsilon < 1$, $\ln(1 + \varepsilon/2) \geq \varepsilon/4$. Thus, it follows that $f(h,\varepsilon) = O((1/\varepsilon)\log(h/\varepsilon))$. By our assumption that $1/\varepsilon$ is bounded by a polynomial function in n, this is $O((1/\varepsilon)\log n)$. Clearly,

$$a_f^2 \geq 2\Delta \quad \text{and} \quad a_{-f}^2 \leq \frac{\varepsilon^2}{2h^2}\Delta.$$

Next, we define a set of regions S_j for $-f \leq j \leq f$, each of which is the union of two slabs that are parallel to $\boldsymbol{\beta}$. First, define S_{-f} to be the set of points of $\mathbb{R}^d$ whose squared residuals with respect to $\boldsymbol{\beta}$ are at most a_{-f}^2. Next, for $-f+1 \leq j \leq f$, define S_j to be the set of points of $\mathbb{R}^d$ whose squared residuals with respect to $\boldsymbol{\beta}$ lie in the half-open interval $(a_{j-1}^2, a_j^2]$ (see Fig. 3). For $j > -f$, S_j is the disjoint union of two slabs, one on each side of $\boldsymbol{\beta}$. Observe that, for fixed values of ε and h, the squared distances of the slabs to the central hyperplane vary linearly as a function of Δ.

Given the point set P, for $-f \leq j \leq f$, define the *weight*, denoted 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

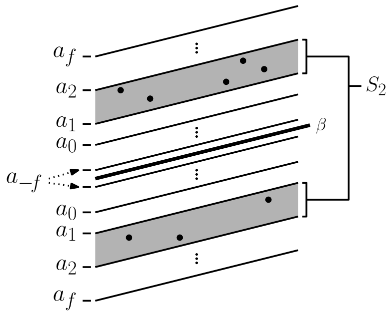

Fig. 3 A slab system and region S_2 with associated weight $w_2 = 5 + 3 = 8$ (not drawn to scale)

interested only in the h closest points to $\boldsymbol{\beta}$, we define the jth *trimmed weight* to be region weights to the first h points, that is,

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

Clearly, $\sum_{j=-f}^{f} \widehat{w}_j = h$, and the sum includes the regions containing the h closest points to $\boldsymbol{\beta}$. We define the approximate LTS cost to be the trimmed weighted sum of squared distances over all the groups, that is,

$$\widehat{\Delta}_{\boldsymbol{\beta}}(P, h) = \sum_{j=-f}^{f} \widehat{w}_j a_j^2. \tag{1}$$

We refer to the collection of regions and weights defined above as a (Δ, ε)-*slab system* for $\boldsymbol{\beta}$. The next lemma shows that, for suitably chosen Δ, this quantity provides an approximation to the LTS cost.

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

(i) $W_0 \geq h \Rightarrow \Delta_{\boldsymbol{\beta}}^{(\text{LTS})}(P, h) < \Delta$
(ii) $W_f < h \Rightarrow \Delta_{\boldsymbol{\beta}}^{(\text{LTS})}(P, h) > \Delta$
(iii) $W_0 < h \leq W_f \Rightarrow \Delta_{\boldsymbol{\beta}}^{(\text{LTS})}(P, h) \leq \widehat{\Delta}_{\boldsymbol{\beta}}(P, h) < (1 + \varepsilon)\Delta_{\boldsymbol{\beta}}^{(\text{LTS})}(P, h)$.

Proof To establish (i), observe that, if $W_0 \geq h$, there are at least h points of P each of whose squared residual with respect to $\boldsymbol{\beta}$ is at most a_0^2. This implies that the LTS cost with respect to $\boldsymbol{\beta}$ is at most $ha_0^2 = \varepsilon\Delta$, which is less than Δ under our assumption that $\varepsilon < 1$. To establish (ii), observe that if $W_f < h$, then at least one point among the h closest is at squared distance greater than $a_f^2 \geq 2\Delta \geq \Delta$, and hence the total cost is at least this large.

Henceforth, let us assume that $W_0 < h \leq W_f$. Because $h \leq W_f$, the trimmed weighted sum of the squared distance values, a_j^2, for j ranging over all f groups is an upper bound on the sum of the h smallest squared residuals, that is,

$$\widehat{\Delta}_{\boldsymbol{\beta}}(P,h) = \sum_{-f\le j\le f} \widehat{w}_j a_j^2 \ge \sum_{1\le i\le h} r_{[i]}^2(\boldsymbol{\beta}) = \Delta_{\boldsymbol{\beta}}^{(\mathrm{LTS})}(P,h).$$

Because $W_0 < h$, at least one inlier lies within a squared distance of at least $a_0^2 = \varepsilon\Delta/h$, and therefore $\Delta_{\boldsymbol{\beta}}^{(\mathrm{LTS})}(P,h) \ge \varepsilon\Delta/h$. Each point of S_{-f} has a squared residual of at most $a_{-f}^2 \le \varepsilon^2\Delta/(2h^2)$, and therefore

$$\widehat{w}_{-f} a_{-f}^2 \le h\frac{\varepsilon^2\Delta}{2h^2} = \frac{\varepsilon}{2}\cdot\frac{\varepsilon\Delta}{h} \le \frac{\varepsilon}{2}\Delta_{\boldsymbol{\beta}}^{(\mathrm{LTS})}(P,h).$$

Thus, we have

$$\widehat{\Delta}_{\boldsymbol{\beta}}(P,h) = \sum_{-f\le j\le f} \widehat{w}_j a_j^2 \le \frac{\varepsilon}{2}\Delta_{\boldsymbol{\beta}}^{(\mathrm{LTS})}(P,h) + \sum_{-f< j\le f} \widehat{w}_j a_j^2.$$

Observe that $a_j^2/a_{j-1}^2 = (1+\varepsilon/2)$, for $-f < j \le f$. Observe that if the point with the ith smallest residual lies within slab j, then $r_{[i]}^2(\boldsymbol{\beta}) > a_{j-1}^2$. Therefore, we have

$$\begin{aligned}
\widehat{\Delta}_{\boldsymbol{\beta}}(P,h) &\le \frac{\varepsilon}{2}\Delta_{\boldsymbol{\beta}}^{(\mathrm{LTS})}(P,h) + \sum_{-f<j\le f} \widehat{w}_j a_{j-1}^2 \frac{a_j^2}{a_{j-1}^2} \\
&= \frac{\varepsilon}{2}\Delta_{\boldsymbol{\beta}}^{(\mathrm{LTS})}(P,h) + \left(1+\frac{\varepsilon}{2}\right)\sum_{-f<j\le f} \widehat{w}_j a_{j-1}^2 \\
&< \frac{\varepsilon}{2}\Delta_{\boldsymbol{\beta}}^{(\mathrm{LTS})}(P,h) + \left(1+\frac{\varepsilon}{2}\right)\sum_{1\le i\le h} r_{[i]}^2(\boldsymbol{\beta}) \\
&= \frac{\varepsilon}{2}\Delta_{\boldsymbol{\beta}}^{(\mathrm{LTS})}(P,h) + \left(1+\frac{\varepsilon}{2}\right)\Delta_{\boldsymbol{\beta}}^{(\mathrm{LTS})}(P,h) \\
&= (1+\varepsilon)\Delta_{\boldsymbol{\beta}}^{(\mathrm{LTS})}(P,h),
\end{aligned}$$

which completes the proof of (iii). □

3.2 The q-Restricted Problem

In this section we present an approximate solution for a restricted form of the LTS problem. Given a point q (which need not be in P) the *q-restricted LTS problem* involves computing the hyperplane passing through q that minimizes the LTS cost. Our 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. By approximating the solution of the restricted problem for this point, we obtain the desired approximation. (Note that the notion of "restriction" presented here is not related to the notion of R-restriction presented in Sect. 2.)

Throughout the remainder of this section, let us assume that a restriction point $q \in \mathbb{R}^d$ is given. We begin by defining a dual space in which each point is associated with a slab system, whose the central hyperplane passes through the restriction point q. (This representation is unrelated to the dual representation given in Sect. 2.) Recall that a slab system in $\mathbb{R}^d$ is determined by two quantities, a central hyperplane $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_d)$ and an LTS cost estimate $\Delta > 0$.

The constraint that the central hyperplane $y = \sum_{i=1}^{d-1} \beta_i x_i + \beta_d$ passes through q implies that $q_d = \sum_{i=1}^{d-1} \beta_i q_i + \beta_d$, or equivalently $\beta_d = q_d - \sum_{i=1}^{d-1} \beta_i q_i$. By performing a single stage of Gaussian elimination, we may eliminate the coordinate β_d and represent each central hyperplane of a q-restricted system as a $(d-1)$-element vector $(\beta_1, \ldots, \beta_{d-1})$. Based on this observation, we define the dual representation of any (Δ, ε)-slab system to be the d-vector $(\beta_1, \ldots, \beta_{d-1}, \delta) \in \mathbb{R}^d$, where we define $\delta = \sqrt{\Delta}$. We refer to this dual representation as *slab space*. (This is different from the notion of slab space that was introduced in Sect. 2.) We say that two slab systems are *combinatorially equivalent* if corresponding slabs contain the same subset of P. The following lemma shows how to identify the combinatorially distinct slab systems with the cells of a hyperplane arrangement.

Lemma 2 *Given an n-element point set P in $\mathbb{R}^d$, trimming parameter h, $\varepsilon > 0$, and a restriction point $q \in \mathbb{R}^d$, there exists a hyperplane arrangement, $\mathcal{A}_{h,\varepsilon}(P, q)$, in the associated d-dimensional slab space consisting $O((n/\varepsilon)\log n)$ hyperplanes, such that the points lying within any given cell of this arrangement correspond to combinatorially equivalent q-restricted slab systems.*

Proof Recall that the slab boundaries of a slab system are defined by a collection of $2f+1$ values a_j, for $-f \le j \le f$ and $f = O((1/\varepsilon)\log n)$. Given P and ε, we define a collection of $2n(2f+1) = O((n/\varepsilon)\log n)$ triples

$$\Psi = \big\{(p, j, b) : p \in P, -f \le j \le f, b \in \{0, 1\}\big\}.$$

Intuitively, the triple $(p, j, b) \in \Psi$ corresponds to the set of q-restricted slab systems such that point p lies on the jth hyperplane (that is, at vertical distance a_j) above the central hyperplane if $b = 0$ and below the central hyperplane if $b = 1$ (see Fig. 4).

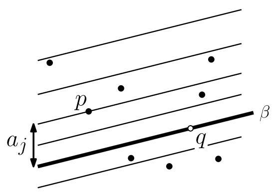

Fig. 4 An event corresponding to the triple $(p, j, 0)$, where q is the restriction point

Let $p=(x_1,\dots,x_d,y)$. Recall that $a_j=\sqrt{(1+\varepsilon/2)^j\varepsilon\Delta/h}$. The point p lies on the desired hyperplane if and only if

$$y=\sum_{i=1}^{d-1}\beta_i x_i+\beta_d+(-1)^b a_j=\sum_{i=1}^{d-1}\beta_i x_i+\left(q_d-\sum_{i=1}^{d-1}\beta_i q_i\right)+(-1)^b a_j$$
$$=q_d+\sum_{i=1}^{d-1}\beta_i(x_i-q_i)+(-1)^b a_j.$$

(Note that, if $b=0$, the point lies on the jth hyperplane above the central hyperplane, if $b=1$, it is on the jth hyperplane below.) By defining $c_{j,\varepsilon,h}=\sqrt{(1+\varepsilon/2)^j\varepsilon/h}$, it follows that p lies on the desired hyperplane of the slab system if and only if the associated point in slab space for this slab system is in the set $\boldsymbol{\beta}(p,j,b)$, which is defined to be

$$\boldsymbol{\beta}(p,j,b)=\left\{(\beta_1,\dots,\beta_{d-1},\delta):y=q_d+\sum_{i=1}^{d-1}\beta_i(x_i-q_i)+(-1)^b c_{j,\varepsilon,h}\delta\right\}.$$

Observe that this is a linear function of $(\beta_1,\dots,\beta_{d-1},\delta)$, and so $\boldsymbol{\beta}(p,j,b)$ is a hyperplane in slab space.

Let $\mathcal{A}_{h,\varepsilon}(P,q)$ denote the arrangement of $O((n/\varepsilon)\log n)$ hyperplanes in our d-dimensional q-restricted slab space defined by the set of hyperplanes $\boldsymbol{\beta}(p,j,b)$, for all $(p,j,b)\in\Psi$. To complete the proof, we assert that the points within any single cell of this arrangement correspond to combinatorially equivalent slab systems. Suppose to the contrary that the points in slab space of two slab systems Σ and Σ' are combinatorially distinct but lie within the same cell. Because they are combinatorially distinct, there exists j such that the corresponding slabs S_j and S'_j of these two systems contain different subsets of P. That is, there is a point $p\in P$ that lies in one of these slabs but not the other. Therefore, a straight-line path between the points in slab space dual to Σ and Σ' must pass through an intermediate slab system in which p lies on the boundary of the corresponding slab. This implies that the points of Σ and Σ' lie on opposite sides of a hyperplane of $\mathcal{A}_{h,\varepsilon}(P,q)$ corresponding to this incidence, which contradicts the hypothesis that the points in slab space for Σ and Σ' are in the same cell of the arrangement (See Fig. 5(a).) □

By standard results on hyperplane arrangements [8], the total combinatorial complexity of the arrangement $\mathcal{A}_{h,\varepsilon}(P,q)$ described in the above lemma is $O((n/\varepsilon)^d\log^d n)$. Observe that fixing the value of $\Delta\ge 0$ (and thus fixing the value of δ) defines a $(d-1)$-dimensional "slice" of this arrangement, denoted $\mathcal{A}_{h,\varepsilon,\Delta}(P,q)$, whose total combinatorial complexity is $O((n/\varepsilon)^{d-1}\log^{d-1} n)$. When P, q, h, and ε are clear from context, we refer to this arrangement simply as $\mathcal{A}_\Delta$. (An example is shown in Fig. 5(b).)

The principal advantage of expressing matters in term of the arrangement in slab space is that, since one of the coordinates of the configuration space varies monotonically with Δ, we can solve the problem by performing a parametric binary search based on this coordinate. Each probe of the parametric search is given a value Δ, and

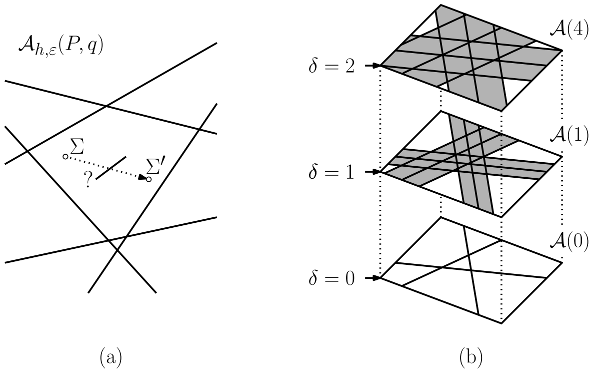

Fig. 5 Slab-space transformation: (**a**) the proof that points of the same cell correspond to combinatorially equivalent slab systems, (**b**) examples of slices the arrangement of slap space for a case involving three points (*three dual lines*) and a slab system consisting of a single slab

it searches the arrangement $\mathcal{A}_\Delta$ to determine whether there exists a feasible solution of cost approximately Δ or less. (The LMS algorithm given by Erickson et al. [13] can be viewed as a special case of this approach where there is only one slab, that is, $f(h,\varepsilon) = 1$.)

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 h, a target cost Δ and an approximation parameter ε, we say that an algorithm solves the *ε-approximate LTS decision problem* if it satisfies the following conditions. If the algorithm accepts Δ, then $\Delta \geq \Delta^{(\text{LTS})}(P,h)$. If the algorithm rejects Δ, then $\Delta < (1+\varepsilon)\Delta^{(\text{LTS})}(P,h)$. Note that, if $\Delta^{(\text{LTS})}(P,h) \leq \Delta < (1+\varepsilon)\Delta^{(\text{LTS})}(P,h)$, the algorithm may either accept or reject Δ. The q-restricted decision problem is defined analogously, subject to the constraint that the hyperplane passes through q.

Lemma 3 *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 h, an approximation parameter $0 < \varepsilon \leq 1$, and a restriction point $q \in \mathbb{R}^d$ (not necessarily in P). Given a target cost $\Delta > 0$, it is possible to solve the q-restricted ε-approximate LTS decision problem in time $O((n/\varepsilon)^{d-1} \log^d n)$ and space $O((n/\varepsilon)^{d-1} \log^{d-1} n)$.*

Proof Given P, q, h, ε, and Δ, we begin by constructing the $(d-1)$-dimensional arrangement $\mathcal{A}_{h,\varepsilon,\Delta}(P,q) = \mathcal{A}_\Delta$, defined earlier. Recall that each cell of this arrangement corresponds to a set of combinatorially equivalent partitions of the point set P into $O(f(h,\varepsilon))$ slab sets of the form $S_j \cap P$. For each cell, we maintain a $(2f+1)$-element vector of weights $(w_{-f}, \ldots, w_f)$, where w_j denotes the number of points of P in the jth slab set. We also maintain the smallest index k, $-f \leq k \leq f$, such that the accumulated sum of weights satisfies $W_k \geq h$, and the values of the accumulated weights W_0 and W_k. If no such k exists, we set $k = f+1$.

By our assumption that the point set 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 (as a point transitions out of one slab and into another). Thus, by traversing this arrangement, it is a simple exercise to update the above values in constant time as we enter each new cell of the arrangement.

Given the above quantities, we claim that in $O(1)$ time we can also update the value of $\widehat{\Delta}_{\boldsymbol{\beta}}(P,h)$, as defined in Eq. (1), for each new cell visited. First observe that it is possible to compute each trimmed weight $\widehat{w}_j$ in $O(1)$ time by noting that, if $j < k$, then $\widehat{w}_j = w_j$, if $j > k$, then $\widehat{w}_j = 0$, and if $j = k$, then $\widehat{w}_j = w_j - (W_k - h)$. Thus, as we move from one cell to another, at most a constant number of trimmed weight values change, and so, the LTS cost estimate $\widehat{\Delta}_{\boldsymbol{\beta}}(P, h)$ (recall Eq. (1) of Sect. 3.1) can be updated in constant time.

Assuming that we know the value of $\widehat{\Delta}_{\boldsymbol{\beta}}(P, h)$ for each cell visited, we proceed as follows. If, for any cell of $\mathcal{A}_\Delta$, we find that either (1) $W_0 \geq h$ or (2) $W_0 < h \leq W_f$ and $\widehat{\Delta}_{\boldsymbol{\beta}}(P, h) \leq \Delta$, then we accept Δ. If, after traversing all the cells, we have not encountered either of these two conditions, we reject Δ.

To establish the correctness of this procedure, it suffices to show that, if the algorithm accepts, then $\Delta^{(\mathrm{LTS})}(P, h) \leq \Delta$, and, if the algorithm rejects, then $\Delta < (1 + \varepsilon)\Delta^{(\mathrm{LTS})}(P, h)$. Observe that if the algorithm accepts, then there are two possibilities. The first is that there exists a cell such that $W_0 \geq h$, and therefore, by Lemma 1(i), for every $\boldsymbol{\beta}$ in this cell, we have $\Delta^{(\mathrm{LTS})}_{\boldsymbol{\beta}}(P, h) < \Delta$. Since the optimum is at least this small, we have $\Delta^{(\mathrm{LTS})}(P, h) < \Delta$. The other reason for accepting is that there is a cell such that $W_0 < h \leq W_f$ and $\widehat{\Delta}_{\boldsymbol{\beta}}(P, h) \leq \Delta$. If so, by Lemma 1(iii) we have $\Delta^{(\mathrm{LTS})}_{\boldsymbol{\beta}}(P, h) \leq \widehat{\Delta}_{\boldsymbol{\beta}}(P, h)$, which implies that $\Delta^{(\mathrm{LTS})}(P, h) \leq \Delta$.

On the other hand, if the algorithm rejects, there are two possibilities for every cell visited. In the first case, $W_f < h$, which by Lemma 1(ii) implies that for every $\boldsymbol{\beta}$ in this cell $\Delta < \Delta^{(\mathrm{LTS})}_{\boldsymbol{\beta}}(P, h)$. In the second case, we have $W_0 < h \leq W_f$ and $\widehat{\Delta}_{\boldsymbol{\beta}}(P, h) > \Delta$. By Lemma 1(iii), it follows that, for all $\boldsymbol{\beta}$ in the cell, we have $\widehat{\Delta}_{\boldsymbol{\beta}}(P, h) \leq (1 + \varepsilon)\Delta^{(\mathrm{LTS})}_{\boldsymbol{\beta}}(P, h)$. Therefore, for all $\boldsymbol{\beta}$ in the cell, we have $\Delta < (1 + \varepsilon)\Delta^{(\mathrm{LTS})}_{\boldsymbol{\beta}}(P, h)$. Since these inequalities hold for every cell of the arrangement, we have $\Delta < (1 + \varepsilon)\Delta^{(\mathrm{LTS})}(P, h)$, as desired.

To derive the running time, observe that each cell is processed in constant time (after initializations taking $O(n)$ time). If $d \geq 3$, the algorithm's running time is dominated by the total size of the arrangement $\mathcal{A}_\Delta$. By the comments made earlier, the combinatorial complexity of $\mathcal{A}_\Delta$ is $O((n/\varepsilon)^{d-1} \log^{d-1} n)$. If $d = 2$, then the arrangement degenerates to a set of $O(f(h, \varepsilon))$ points, and so sorting is required. The running time in this case is $O(f(h, \varepsilon) \log f(h, \varepsilon))$, which under our assumption that $1/\varepsilon$ is bounded by a polynomial in n, is $O((n/\varepsilon) \log^2 n)$. In either case, the space is dominated by the arrangement size, which is $O((n/\varepsilon)^{d-1} \log^{d-1} n)$. Clearly, these satisfy the bounds in the lemma's statement. □

Next, we show how to convert this approximate decision algorithm into an approximation algorithm for the restricted problem. We apply straightforward adaptation of the randomized sample-and-sweep approach given in [13].

Lemma 4 *Given the same setup as Lemma 3, it is possible to solve the q-restricted ε-approximate LTS problem with high probability in time $O((n/\varepsilon)^{d-1} \log^{d+1} n)$ and space $O((n/\varepsilon)^{d-1} \log^{d-1} n)$.*

Proof Recall the d-dimensional slab-space hyperplane arrangement $\mathcal{A}_{h,\varepsilon}(P, q) = \mathcal{A}$ described in Lemma 2. For a parameter r to be determined below, we first compute a random sample of r arrangement vertices. Each is computed by sampling d-element subsets of P at random and constructing the associated vertex in the hyperplane arrangement of slab space. We then sort these vertices according to their Δ values and apply Lemma 3 in concert with a binary search to find a pair of consecutive cost values, $\Delta^- \le \Delta^+$, such that the decision algorithm rejects Δ^- and accepts Δ^+.

By the conditions of the decision problem, $\Delta^- < (1+\varepsilon)\Delta^{(\mathrm{LTS})}(P,h)$ and $\Delta^+ \ge \Delta^{(\mathrm{LTS})}(P,h)$. Clearly, there exists $\Delta \in [\Delta^-, \Delta^+]$ such that

$$\Delta^{(\mathrm{LTS})}(P,h) \le \Delta < (1+\varepsilon)\Delta^{(\mathrm{LTS})}(P,h),$$

and therefore, an approximate solution to the restricted problem lies somewhere between these values, that is, somewhere between the two arrangement slices $\mathcal{A}_{\Delta^-}$ and $\mathcal{A}_{\Delta^+}$. By standard results on ε-nets [1], assuming that r is $\Omega(\log n)$, with high probability the number of arrangement vertices of $\mathcal{A}$ lying between these two slices is $O(N/r)$, where N is the number of arrangement vertices.

To determine the final solution we first construct the arrangement $\mathcal{A}_{\Delta^-}$ and then perform a sweep, updating the arrangement A_Δ as Δ varies from Δ^- to Δ^+. As observed earlier, it is possible to update the approximate LTS cost associated with each cell of the arrangement in $O(1)$ time, and the other data structures required by the sweep can be updated in time $O(\log n)$ per arrangement vertex. We determine the cell of lowest approximate LTS cost and return any hyperplane associated with this cell.

The running time of the algorithm involves the following components. First, computing and sorting the sample of arrangement vertices can be done in $O(r \log r)$ time. Second, the time to perform the binary search is $O(\log r)$ times the running time of the decision procedure, which by Lemma 3 is $O((n/\varepsilon)^{d-1} \log^d n)$. The time to compute $\mathcal{A}_{\Delta^-}$ is dominated by this latter quantity (since both involve computing a single slice of $\mathcal{A}$). Finally, the time to sweep the arrangement is $O((N/r)\log n)$. Given that $N = O((n/\varepsilon)^d \log^d n)$, we obtain the best running time by setting $r = cn/\varepsilon$, for an appropriate constant c. By our assumption that $1/\varepsilon$ is bounded by a polynomial in n, we have a total running time of

$$\begin{aligned} T(n) &= O(r \log r) + O\left((\log r)\left(\frac{n}{\varepsilon}\right)^{d-1} \log^d n\right) + O\left(\frac{N}{r}\log n\right) \\ &= O\left(\frac{n}{\varepsilon}\log n + (\log n)\left(\frac{n}{\varepsilon}\right)^{d-1} \log^d n + \frac{(n/\varepsilon)^d \log^d n}{n/\varepsilon}\log n\right) \\ &= O\left(\left(\frac{n}{\varepsilon}\right)^{d-1} \log^{d+1} n\right). \end{aligned}$$

The total space is bounded by the complexity of a single slice of the arrangement, which is $O((n/\varepsilon)^{d-1} \log^{d-1} n)$. □

3.3 The Approximation Algorithm

We are now in a position to present the algorithm that establishes Theorem 3. Before this, we provide one more technical lemma, which relates the LTS costs of two parallel hyperplanes as a function of the vertical distance between them.

Lemma 5 *Consider a set of points P in $\mathbb{R}^d$ and a trimming parameter h. Let $\boldsymbol{\beta}$ and $\boldsymbol{\beta}'$ be two parallel hyperplanes, and let Δ and Δ' denote their respective LTS costs. Suppose that these hyperplanes are separated by a vertical distance of $\alpha\sqrt{\Delta/h}$, for some $\alpha \geq 0$. Then $\Delta' \leq (1+\alpha)^2\Delta$.*

Proof Let $r[i]^2$ denote the ith smallest squared residual of the points of P to $\boldsymbol{\beta}$. We have $\Delta = \sum_{i=1}^{h} r_{[i]}^2$. Let $y = \alpha\sqrt{\Delta/h}$ denote the vertical distance between $\boldsymbol{\beta}$ and $\boldsymbol{\beta}'$. It follows that the LTS cost of $\boldsymbol{\beta}'$ satisfies

$$\Delta' \leq \sum_{i=1}^{h} \bigl(|r_{[i]}| + y\bigr)^2.$$

For any sequence of reals $\langle a_1, \ldots, a_h\rangle$, it is a direct consequence of the Cauchy-Schwarz inequality that $\sum_{i=1}^{h} |a_i| \leq (h\sum_{i=1}^{h} a_i^2)^{1/2}$. By combining this with the above inequality we have

$$\begin{aligned}
\Delta' &\leq \sum_{i=1}^{h}\bigl(r_{[i]}^2 + 2|r_{[i]}|y + y^2\bigr) = \sum_{i=1}^{h} r_{[i]}^2 + 2y\sum_{i=1}^{h}|r_{[i]}| + hy^2 \\
&\leq \Delta + 2y\left(h\sum_{i=1}^{h} r_{[i]}^2\right)^{1/2} + hy^2 = \Delta + 2y\sqrt{h\Delta} + hy^2 \\
&= (\sqrt{\Delta} + \sqrt{h}y)^2 = \left(1 + \sqrt{\frac{h}{\Delta}}y\right)^2\Delta.
\end{aligned}$$

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

Here now is the final approximation algorithm. Let $\Delta = \Delta^{(\mathrm{LTS})}(P, h)$ denote the optimum LTS cost of P, and let $\boldsymbol{\beta}$ denote the optimum LTS hyperplane. We start by computing a coarse estimate Δ. For an appropriately chosen constant c (defined below), let P' be a random sample of $c((n/h)\log n)$ points of P (see Fig. 6(a)). Apply the restricted approximate LTS algorithm of Lemma 4 for every restriction point $q \in P'$ and with ε set to 1. Let Δ' be the minimum LTS cost resulting from all of these runs. Clearly $\Delta' \geq \Delta$.

We claim that, with high probability, $\Delta' \leq 6\Delta$. To see this, consider a slab S consisting of the points whose squared distance from $\boldsymbol{\beta}$ is at most $2\Delta/h$ (see Fig. 6(a)). Observe that at least $h/2$ points of P must lie within S, for otherwise the LTS cost of $\boldsymbol{\beta}$ would be strictly greater than $(h/2)(2\Delta/h) = \Delta$, contradicting $\boldsymbol{\beta}$'s optimality. The range space of d-dimensional slabs (for fixed d) has constant VC-dimension, and therefore, by standard results on ε-nets [1], there exists a constant c so that with high probability at least one of the points of P' lies within S. Let us assume that c has been so chosen. Suppose that q is a point of $P' \cap S$, and let $\boldsymbol{\beta}'$ be a hyperplane parallel to $\boldsymbol{\beta}$ passing through q. By applying Lemma 5 with $\alpha = \sqrt{2}$, it follows that $\Delta' \leq (1+\sqrt{2})^2\Delta \leq 6\Delta$, as desired.

To compute the final ε-approximation we compute a refined set of restriction points, one of which is guaranteed to provide the desired approximation. Let $\delta =$

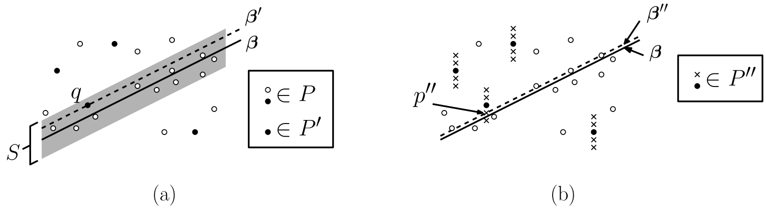

Fig. 6 The final approximation algorithm: (**a**) showing that a sample point q lies within squared distance $2\Delta/h$ of the LTS optimal hyperplane, (**b**) the refined set of restriction points

$(\varepsilon/3)\sqrt{\Delta'/(6h)}$, and consider a collection of values $\{c_{-k},\ldots,c_k\}$, where $c_i = i\delta$ and $k = \lceil\sqrt{2\Delta'/h}/\delta\rceil$. Clearly, $k = O(1/\varepsilon)$. For each point $q \in P'$, create $2k+1$ copies of this point by translating it vertically by a distance c_i, for $-k \le i \le k$. Let P'' be the resulting set of $(2k+1)|P'| = O((1/\varepsilon)(n/h)\log n)$ points. (The copies are shown as $\times$'s in Fig. 6(b).) We then invoke the algorithm of Lemma 4 using each point $q \in P''$ as a restriction point and with the approximation parameter set to $\varepsilon' = \varepsilon/3$. Finally, we return the result of all these runs that produces the minimum LTS cost.

To see that this achieves the desired approximation bound, recall that (with high probability) there exists a point $q \in P'$ within distance $\sqrt{2\Delta/h}$ of $\boldsymbol{\beta}$. Since

$$k\delta \ge \sqrt{\frac{2\Delta'}{h}} \ge \sqrt{\frac{2\Delta}{h}},$$

it follows that $\boldsymbol{\beta}$ passes between two of the copies of q in P''. Since the copies are separated by a vertical distance δ, there is a point of P'' whose vertical distance from $\boldsymbol{\beta}$ is at most $\delta/2$. Let p'' denote this point. By definition of δ, the distance from p'' to $\boldsymbol{\beta}$ is at most

$$\frac{\delta}{2} \le \frac{\varepsilon}{6}\sqrt{\frac{\Delta'}{6h}} \le \frac{\varepsilon}{6}\sqrt{\frac{\Delta}{h}}.$$

Let $\boldsymbol{\beta}''$ denote the hyperplane parallel to $\boldsymbol{\beta}$ passing through p'' (see Fig. 6(b)). By applying Lemma 5 with $\alpha = \varepsilon/6$, it follows that the optimum LTS cost of the restricted problem at p'', which we denote by Δ'', 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 ε', the result has LTS cost at most

$$(1+\varepsilon')\Delta'' = \left(1+\frac{\varepsilon}{3}\right)\Delta'' \le \left(1+\frac{\varepsilon}{3}\right)\left(1+\frac{\varepsilon}{6}\right)^2\Delta.$$

Under our assumption that $\varepsilon \le 1$, it is easy to verify that this is at most $(1+\varepsilon)\Delta$, as desired.

The running time is dominated by the $O((n/(h\varepsilon))\log n)$ invocations of the algorithm of Lemma 4. Thus, by Lemma 4, the running time is

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

and the space is $O((n/\varepsilon)^{d-1}\log^{d-1} n)$. This completes the proof of Theorem 3.

Note that for the typical case in which $h = \Theta(n)$ and ε is a fixed constant, the running time is $\widetilde{O}(n^{d-1})$ with high probability. In closing, we add that the methods applied in this section, computing a (Δ, ε)-slab system, the hyperplane arrangement in slab space, 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 the following well known conjecture. A set of $d+1$ points in $\mathbb{R}^d$ is *affinely dependent* if any one point of the set can be expressed as an affine combination of the others. This is equivalent to saying that the points lie on a common $(d-1)$-dimensional hyperplane. A set of n points is said to be *affinely degenerate* if any $(d+1)$-element subset is affinely dependent.

Conjecture (Hardness of Affine Degeneracy) *The problem of determining whether an n-element subset of the d-dimensional integer grid is affinely degenerate requires $\Omega(n^d)$ time to solve in the worst case.*

Erickson and Seidel [12, 14] proved an $\Omega(n^d)$ lower bound on the number of sidedness queries required to solve the affine degeneracy problem. (Note, however, that the model of computation in which their lower bound holds is not strong enough to solve our problems.) Affine degeneracy in the plane is related to the 3SUM problem [16]. Even though subquadratic algorithms are known for special cases of 3SUM [2], no such algorithm is known for the planar version of affine degeneracy.

The following technical lemma will be useful in a number of our proofs. It bounds various quantities related to points and hyperplanes on the grid.

Lemma 6 *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 α (depending on d but not on M) such that the following hold. Let $P \subseteq \mathbb{Z}^d(M)$ be a set of at least $d+1$ affinely nondegenerate points from the grid.*

(i) *The vertical height of any slab containing P is at least α/M^d.*
(ii) *Let $h = |P|$, let $\boldsymbol{\beta}$ by any nonvertical hyperplane, and let $\Delta^{(\mathrm{LTA})}$ and $\Delta^{(\mathrm{LTS})}$ denote, respectively, the sum of the absolute and squared residuals of P with respect to $\boldsymbol{\beta}$. Then*

$$\Delta^{(\mathrm{LTA})}_{\boldsymbol{\beta}}(P,h) \geq \frac{(h-d)\alpha}{M^d} \quad \text{and} \quad \Delta^{(\mathrm{LTS})}_{\boldsymbol{\beta}}(P,h) \geq \frac{(h-d)\alpha^2}{M^{2d}}.$$

(iii) *Let $\boldsymbol{\beta}$ be the $(d-1)$-dimensional hyperplane that passes through any d points of P. Then the vertical distance from any remaining point of P to $\boldsymbol{\beta}$ is at most αM^{d+1}.*

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 [13] states that if P is affinely independent then any critical slab has vertical height at least α/M^d, for some α depending only on dimension. Therefore, any slab containing P must have at least this height.

To prove (ii), consider the slab of minimum vertical height centered about $\boldsymbol{\beta}$ that contains at least $d+1$ points of P. Let v denote this slab's vertical height. By assertion (i), $v \geq \alpha'/M^d$, for some α'. By minimality at least one of these points is at distance at least $v/2$ from $\boldsymbol{\beta}$ as are the remaining $h-(d+1)$ points of P. Therefore each of these $h-d$ 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.

To prove (iii), let $\boldsymbol{\beta}$ be given by the equation $y = \beta_1 x_1 + \cdots + \beta_{d-1} x_{d-1} + \beta_d$. Let $\{p_1, \ldots, p_d\}$ be the points that define $\boldsymbol{\beta}$. The coefficients of $\boldsymbol{\beta}$ are the solution of a $d \times d$ linear equation whose coefficients are the coordinates of these points. By Cramer's rule [19], each of the coefficients of $\boldsymbol{\beta}$ can be expressed as the ratio of two $d \times d$ determinants, each of whose elements are of absolute value at most M. Therefore, each coefficient of $\boldsymbol{\beta}$ is of absolute value at most $d! \cdot M^d$. Given any point $p \in P$, the vertical distance between its y-coordinate and $\boldsymbol{\beta}$ is of the form $y - \sum_{i=1}^{d} \beta_i x_i$, which is of magnitude $O(d! \cdot M^d \cdot M) = O(M^{d+1})$. □

4.1 Hardness of Exact LTA

The main result of this section is the following theorem regarding the LTA estimator.

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* $h \geq d+1$ *inliers requires* $\Omega(\min(h, n-h)^d)$ *time in the worst case.*

Our proof holds under the assumption that $h-d$ is odd, but since h is an asymptotic quantity in our results, our lower bound holds infinitely often in h. This is commonly allowed in algorithmic lower bounds.

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 7 *Let* $p \in \mathbb{R}^d$, *let* $\boldsymbol{\beta}$ *be a nonvertical* $(d-1)$*-dimensional hyperplane in* $\mathbb{R}^d$, *and let t be a nonnegative real. Let* v_t *denote the vertical vector* $(0, \ldots, 0, t)$, *and let* $p - \boldsymbol{\beta}$ *denote the signed vertical distance from* $\boldsymbol{\beta}$ *to p (positive if p is above* $\boldsymbol{\beta}$*). Then*

(i) $|p - \boldsymbol{\beta}| + |(p + v_t) - \boldsymbol{\beta}| \geq t$, *and equality holds if p and* $p + v_t$ *lie on opposite sides of* $\boldsymbol{\beta}$.

(ii) $|p - \boldsymbol{\beta}| + |(p - v_t) - \boldsymbol{\beta}| \geq t$, *and equality holds if p and* $p - v_t$ *lie on opposite sides of* $\boldsymbol{\beta}$.

(iii) $|p - \boldsymbol{\beta}| + |(p + v_t) - \boldsymbol{\beta}| + |(p - v_t) - \boldsymbol{\beta}| \geq 2t + |p - \boldsymbol{\beta}|$, *and equality holds if* $p - v_t$ *and* $p + v_t$ *lie on opposite sides of* $\boldsymbol{\beta}$.

Proof Clearly, for any $a, b \in \mathbb{R}$, $|a| + |b| \geq |a + b|$, and equality is achieved if both a and b are of the same sign. Since p, $p + v_t$, and $p - v_t$ are vertically aligned, we may treat their vertical distances to any fixed hyperplane $\boldsymbol{\beta}$ as numbers on the real line.

To prove (i), observe that $|-(p - \boldsymbol{\beta})| + |(p + v_t) - \boldsymbol{\beta}| \geq |-(p - \boldsymbol{\beta}) + (p + v_t - \boldsymbol{\beta})| = t$. If p and $p + v_t$ lie on opposite sides of $\boldsymbol{\beta}$, then both $-(p - \boldsymbol{\beta})$ and $(p + v_t) - \boldsymbol{\beta}$ are nonnegative, and so equality holds. Assertion (ii) follows from (i) by setting p to $p - v_t$.

To prove (iii) we consider two cases. If p lies on or below $\boldsymbol{\beta}$, then by (i) we have

$$\big(|p - \boldsymbol{\beta}| + \big|(p + v_t) - \boldsymbol{\beta}\big|\big) + \big|(p - v_t) - \boldsymbol{\beta}\big| \geq t + \big|(p - v_t) - \boldsymbol{\beta}\big|.$$

Since p is below $\boldsymbol{\beta}$, this is equal to $t + (|p - \boldsymbol{\beta}| + t) = 2t + |p - \boldsymbol{\beta}|$. Equality holds if $p + v_t$ lies above $\boldsymbol{\beta}$. A symmetrical argument applies when p lies above $\boldsymbol{\beta}$. □

Our second technical lemma is a simple counting utility.

Lemma 8 *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* $n_2 + 2n_3 \geq n + (d + 1)$ *and* $n_3 - n_1 \geq d + 1$.

Proof If we subtract the first equation from the second equation, we have $-n_0 + n_2 + 2n_3 = n + (d + 1)$, which implies that $n_2 + 2n_3 \geq n + (d + 1)$. If we multiply the first equation by 2 and subtract it from the second equation we have $-2n_0 - n_1 + n_3 = d + 1$, implying that $n_3 - n_1 \geq d + 1$. □

Before giving the proof of Theorem 4, we present a special case, where $h = 2n/3 + (d + 1)$. The proof is similar in spirit to Theorem 5.3 of [13] for LMS, but there are a number of novel elements. The construction of [13] 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 h 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 9 *Let* n *and* h *be positive integers*, *where* n *is divisible by* 3 *and* $h = (2n/3) + (d + 1)$. *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* h *inliers requires* $\Omega(n^d)$ *time in the worst case.*

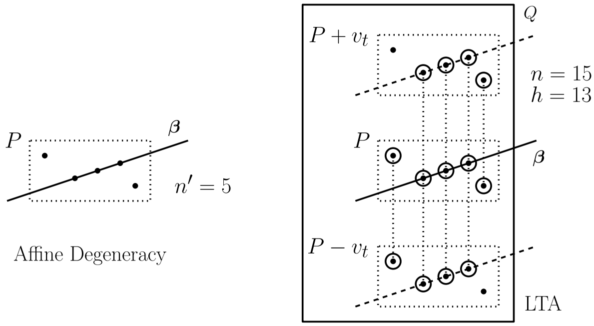

Fig. 7 Proof of Lemma 9 (not drawn to scale). *The circled points* are in Q'

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 $h = 2n' + (d+1)$. Clearly, $h = (2n/3) + (d+1)$, as desired. Given a real t, recall that $v_t = (0, \ldots, 0, t)$ denotes a vertical vector of length t, and let $P + v_t$ denote the vertical translation of P by distance t. We fix the value t to be a positive real value that is sufficiently large so that, if $\boldsymbol{\beta}$ is any nonvertical hyperplane passing through d or more points of P, then every point of P lies strictly within vertical distance $t/5$ of $\boldsymbol{\beta}$. If the points of P lie on an integer grid $[-M, +M]^d$ for some M, then by Lemma 6(iii), $t = O(M^{d+1})$. (The value of t is larger than what is needed for this proof, but it will be reused in the proof of Theorem 4.)

We create a point set Q of size n by taking the union of three translated copies of P, called *clones*, P, $P - v_t$, and $P + v_t$. (See Fig. 7.) We will establish the claim that P contains $d+1$ points lying on a common hyperplane $\boldsymbol{\beta}$ if and only if $\Delta_{\boldsymbol{\beta}}^{(\mathrm{LTA})}(Q, h) \le \Delta$, where $\Delta = (n' + (d+1))t$. Assuming the hardness of affine degeneracy, we will therefore have the desired lower bound of $\Omega((n')^d) = \Omega(n^d)$ on the hardness of this special case of LTA.

To prove the "only if" part of the claim, let us assume that there exists a hyperplane $\boldsymbol{\beta}$ containing at least $d+1$ points of P. Let P^0 denote any $d+1$ points of P lying on $\boldsymbol{\beta}$, and among the remaining points of P, let P^+ be the points of P that lie on or above $\boldsymbol{\beta}$ and let P^- be the points that lie strictly below $\boldsymbol{\beta}$. Thus, $|P^-| + |P^+| + |P^0| = n'$. By definition of t, all the points of $P + v_t$ lie above $\boldsymbol{\beta}$. Similarly, all the points of $P - v_t$ lie below $\boldsymbol{\beta}$.

Define a subset $Q' \subseteq Q$ of size h as follows. For each $p \in P^+$, add p and $p - v_t$ to Q'. For each $p \in P^-$, add p and $p + v_t$ to Q'. Finally, for each $p \in P^0$, add p, $p - v_t$, and $p + v_t$ to Q'. Clearly,

$$\left|Q'\right| = 2\left(\left|P^+\right| + \left|P^-\right|\right) + 3\left|P^0\right| = 2\left(\left|P^+\right| + \left|P^-\right| + \left|P^0\right|\right) + \left|P^0\right| = 2n' + (d+1) = h.$$

Recall that $p - \boldsymbol{\beta}$ denotes the signed vertical distance from $\boldsymbol{\beta}$ to p. If $p \in P^+$ then the combined contribution of p and $p - v_t$ to $\Delta_{\boldsymbol{\beta}}^{\mathrm{LTA}}(Q, h)$ is $|p - \boldsymbol{\beta}| + |(p - v_t) - \boldsymbol{\beta}|$ which by Lemma 7(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 must strictly exceed Δ. Let $\boldsymbol{\beta}$ be any nonvertical hyperplane. Let $\Delta_{\boldsymbol{\beta}}$ denote the sum of the $d+1$ smallest absolute residuals of P with respect to $\boldsymbol{\beta}$. By hypothesis, no $d+1$ points of P lie on any hyperplane, and therefore $\Delta_{\boldsymbol{\beta}} > 0$. To complete the proof, it suffices to show that $\Delta_{\boldsymbol{\beta}}^{\mathrm{LTA}}(Q,h) \geq \Delta + \Delta_{\boldsymbol{\beta}}$, since it will then follow that $\Delta_{\boldsymbol{\beta}}^{\mathrm{LTA}}(Q,h) > \Delta$.

Let $Q' \subseteq Q$ denote the h points having the smallest absolute residuals with respect to $\boldsymbol{\beta}$. 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 $|\{p - v_t, p, p + v_t\} \cap Q'| = i$. 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 8 hold (using n' in the role of n).

Let us consider the contribution of each of these subsets to the LTA cost for $\boldsymbol{\beta}$.

- 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. If p lies below $\boldsymbol{\beta}$, both p and $p + v_t$ will be closer to $\boldsymbol{\beta}$ than is $p - v_t$, and so the contribution of this group will be $|p - \boldsymbol{\beta}| + |(p + v_t) - \boldsymbol{\beta}|$. By Lemma 7(i) this is at least t. On the other hand, if p lies above $\boldsymbol{\beta}$, p and $p - v_t$ will be the closer of the triple, and they will contribute a total of $|p - \boldsymbol{\beta}| + |(p - v_t) - \boldsymbol{\beta}|$. By Lemma 7(ii) this is also at least t.
- If $p \in P_3$, then p and both of its copies contribute, and so the total is

  $$|p - \boldsymbol{\beta}| + \big|(p + v_t) - \boldsymbol{\beta}\big| + \big|(p - v_t) - \boldsymbol{\beta}\big|,$$

  which by Lemma 7(iii) is at least $2t + |p - \boldsymbol{\beta}|$.

Summing the contributions for all four cases and applying Lemma 8(i), it follows that the LTA cost of $\boldsymbol{\beta}$ satisfies

$$\begin{aligned}\Delta_{\boldsymbol{\beta}}^{\mathrm{LTA}}(Q,h) &\geq \sum_{p \in P_2} t + \sum_{p \in P_3} \big(2t + |p - \boldsymbol{\beta}|\big) \geq n_2 t + 2n_3 t + \sum_{p \in P_3} |p - \boldsymbol{\beta}| \\ &\geq \big(n' + (d+1)\big)t + \sum_{p \in P_3} |p - \boldsymbol{\beta}| = \Delta + \sum_{p \in P_3} |p - \boldsymbol{\beta}|. \end{aligned} \quad (2)$$

From Lemma 8(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 $\Delta_{\boldsymbol{\beta}}$. □

We can now provide the proof of the main theorem.

Proof of Theorem 4 We assume that $h - d$ is odd, and therefore our lower bound holds infinitely often in the asymptotic parameters n and h. Our proof is based on two cases, depending on the relationship between h and n. For the first case, let us assume that $h \leq (2n/3) + (d+1)$. We show that solving LTA for n points and h inliers requires $\Omega(h^d)$ time under the assumption of the hardness of affine degeneracy. Let $n' = (h - (d+1))/2$. Since $h - d$ is odd and $h \geq d+1$, 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 $\hat{h} = 2n' + (d+1)$, $\hat{n} = 3n'$, and $k = n - \hat{n}$. Observe that $\hat{h}$ and $\hat{n}$ satisfy the conditions on h and n, respectively, in the statement of Lemma 9, and therefore we may apply the construction of that lemma to P. The resulting number of inliers is

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

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

$$\hat{n} = 3n' = \frac{3(h-(d+1))}{2} \le \frac{3((\frac{2n}{3}+(d+1))-(d+1))}{2} = n.$$

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 h inliers, and by Lemma 9 the time needed to solve such an instance in the worst case is $\Omega(\hat{n}^d) = \Omega((n')^d) = \Omega(h^d)$. This completes the first case.

For the second case, let us assume that $h > (2n/3) + (d+1)$. We show that solving LTA for n points and h inliers requires $\Omega((n-h)^d)$ time under the assumption of the hardness of affine degeneracy. Let P be a point set of size $n' = n - h + (d+1)$ for which we want to solve the affine degeneracy problem. Let $\hat{h} = 2n' + (d+1)$, $\hat{n} = 3n'$, and $k = 3h - 2n - 3(d+1)$. Note that by the lower bound on h and the fact that $h - d$ is odd, it follows that $k \ge 0$, and k is even. As before, $\hat{h}$ and $\hat{n}$ satisfy the conditions of Lemma 9 in the roles of h 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 the origin. We add two copies of R to the construction, one translated vertically up by distance $t/2$ above P's centroid and the other down by distance $t/2$ below P's centroid. (It is not important that the centroid be used. Any point lying within P's convex hull suffices.) Recall that every point of P lies within distance $t/5$ of any hyperplane that passes through d or more points of P, and therefore the points of R lie within vertical distance $t/2 + t/5 < 3t/4$ of such a hyperplane. The points of $P + v_t$ and $P - v_t$ all lie at distance at least $t - t/5 > 3t/4$. Therefore, the points of R have smaller residuals than any point of the translated sets $P + t$ or $P - t$, and hence all the points of R will be counted among the inliers.

The total number of points in the resulting construction is

$$\hat{n} + k = 3n' + \bigl(3h - 2n - 3(d+1)\bigr) = 3\bigl(n - h + (d+1)\bigr) + \bigl(3h - 2n - 3(d+1)\bigr) = n.$$

Also, the total number of inliers is

$$\begin{aligned}\hat{h} + k &= \bigl(2n' + (d+1)\bigr) + \bigl(3h - 2n - 3(d+1)\bigr)\\ &= \bigl(2\bigl(n - h + (d+1)\bigr) + (d+1)\bigr) + \bigl(3h - 2n - 3(d+1)\bigr) = h.\end{aligned}$$

Thus, by Lemma 9 the time needed to solve such an instance in the worst case is $\Omega(\hat{n}^d) = \Omega((n')^d) = \Omega((n-h)^d)$. This completes the proof. □

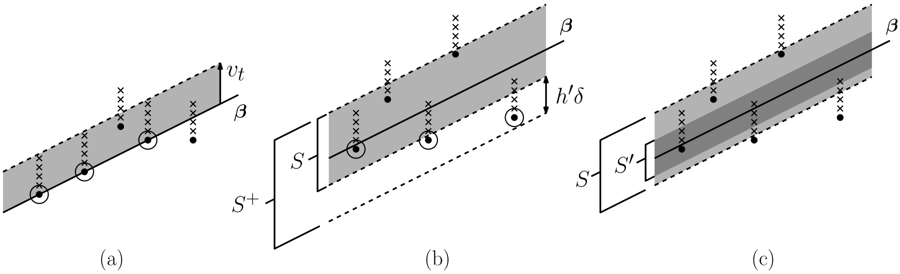

Fig. 8 Proof of Theorem 5 for $d = 2$, $h = 15$, and $h' = 5$: (**a**) the original point set (*black circles*) and copies ($\times$'s), (**b**) the slabs S and S^+, and (**c**) lower bounding the LTS cost

It is natural to wonder whether an analogous result to Theorem 4 holds for LTS. Unfortunately, the proof of Theorem 4 makes critical use of the linearity of absolute residuals in the definition of the LTA cost, which does not generalize to squared distances. We conjecture that such a lower bound holds, however.

The above theorem does not provide an interesting lower bound if h is very small. The following result shows that both the exact LTS and LTA problems are hard even for small values of h. The proof is a relatively straightforward adaptation of the analogous result in [13].

Theorem 5 *Under the assumption of the hardness of affine degeneracy, both the linear LTA and LTS estimators for a set of n points in $\mathbb{Z}^d$ with h inliers require $\Omega((n/h)^d)$ time to compute in the worst case.*

Proof We prove the result for LTS only, and note that the LTA version comes about by a direct modification. Suppose we are given a set P of $n' = n(d+1)/h$ 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 $\delta = \delta'(d+1)/(hM^d)$ for some sufficiently small constant $\delta' > 0$ (whose value will be given later). Let $h' = h/(d+1)$. We construct a new point set Q consisting of h' copies of P, where, for $1 \le i \le h'$, the ith copy is shifted upward a distance of $(i-1)\delta$ (see Fig. 8). The set consisting of the h' copies of each point of P is called a *group*.

If P is affinely degenerate, then there exists a hyperplane $\boldsymbol{\beta}$ that contains at least $d+1$ points of P (see Fig. 8(a)). Let $t = h'\delta$, and consider the slab bounded by $\boldsymbol{\beta}$ and $\boldsymbol{\beta} + v_t$ (the vertical translation of $\boldsymbol{\beta}$ by distance t). Observe that this slab is high enough to contain the union of the groups associated with these $d+1$ points, and so it contains at least $h'(d+1) = h$ of the points of Q. The optimal LTS cost is not greater than the LTS cost of $\boldsymbol{\beta}$, which (by considering just the points within these $d+1$ groups) is at most

$$(d+1)\sum_{i=1}^{h'}\bigl((i-1)\delta\bigr)^2 < h'(d+1)\bigl(h'\delta\bigr)^2 \le h'(d+1)\left(\frac{\delta'}{M^d}\right)^2.$$

Let Δ_0 denote this bound.

For the remainder of the proof, let us consider the converse, where P is affinely nondegenerate. That is, every $(d+1)$-element subset of P is affinely independent. Let $\boldsymbol{\beta}$ be the optimal LTS hyperplane for Q, and let S be the narrowest slab centered at $\boldsymbol{\beta}$ that contains h points of Q (see Fig. 8(b)). Each group that contributes at least one point to S can contribute at most h' points in all, and thus, the total number of distinct groups that contribute a point to S is at least $h/h' = d+1$. Let S^+ be the expanded slab that results by lowering S's lower bounding hyperplane by a distance of $h'\delta$. Since the vertical extent of each group does not exceed $h'\delta$, it follows that S^+ contains the lowest point of each of $d+1$ groups, which implies that S^+ contains at least $d+1$ points of P. (Three such points are highlighted in Fig. 8(b).)

Since P is affinely nondegenerate, by Lemma 6(i) any slab containing $d+1$ points of P has height at least α/M^d for some constant α. Thus, the height of S^+ is at least α/M^d, which implies that the height of S, denoted $\mathrm{ht}(S)$, satisfies

$$\mathrm{ht}(S) \geq \frac{\alpha}{M^d} - h'\delta = \frac{\alpha}{M^d} - \frac{\delta'}{M^d} = \frac{\alpha - \delta'}{M^d}.$$

Next, consider a slab S', centered about $\boldsymbol{\beta}$ whose height is half that of S (see Fig. 8(c)). We assert that at least h' points of Q lie within $S \setminus S'$. If this were not so, then strictly more than $h - h' = dh'$ points of P would lie within $S \cap S'$. Since each group has h' points, strictly more d groups would contribute at least one point to S'. As we did above, if we expand S' by lowering its lower bounding hyperplane by $h'\delta$, it follows that the resulting slab contains at least $d+1$ points of P. The height of this expanded slab is $((\alpha - \delta')/2M^d) + h'\delta$. If we make δ' smaller than α, it is easily verified that this expanded slab is of height less than α/M^d. However, the fact that it contains at least $d+1$ points of P contradicts Lemma 6(i). Therefore, at least h' points of Q lie within $S \setminus S'$.

All the points of $S \setminus S'$ are at distance at least $\mathrm{ht}(S)/4$ from $\boldsymbol{\beta}$, and since there are h' of them, it follows that the LTS cost of $\boldsymbol{\beta}$ is at least

$$h'\left(\frac{\mathrm{ht}(S)}{4}\right)^2 \geq h'\left(\frac{\alpha - \delta'}{4M^d}\right)^2.$$

Let Δ_1 denote this quantity. If we set δ' smaller than $4\alpha/(5\sqrt{d+1})$, it is easily verified that $\Delta_0 < \Delta_1$. (Note that this will also satisfy our previous constraint that $\delta' < \alpha$.)

In summary, if P is affinely degenerate, the optimum LTS cost of Q is at most Δ_0, and if P is affinely nondegenerate, the optimum LTS cost is at least Δ_1. Since $\Delta_0 < \Delta_1$, we have reduced affine degeneracy for a set of $n' = \Omega(n/h)$ points to solving LTS on n points with h inliers. Therefore, under the assumption of the hardness of affine degeneracy, this requires $\Omega((n/h)^d)$ time in the worst case. □

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

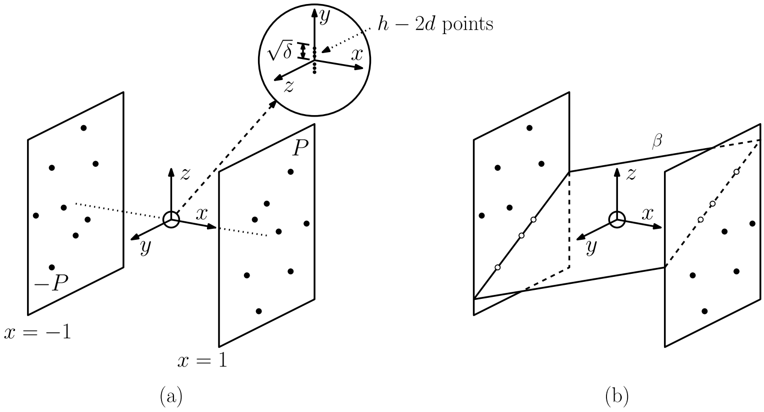

Fig. 9 Proof of Lemma 6: (**a**) the construction, (**b**) the affinely degenerate case

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. [13] for approximating LMS. Recall that ε_r denotes the allowed error in the residual.

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

Proof As before, we prove the result for LTS only. Suppose we are given a set P of $n' = n/2 - h/2 - d$ 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 (3) a set of $h - 2d$ 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 at most δ (see Fig. 9(a)).

If any d points of P lie on a common $(d-2)$-flat, then by symmetry there is a nonvertical $(d-1)$ hyperplane $\boldsymbol{\beta}$ that passes through this flat on the hyperplane $x_1 = 1$, its reflection on $x_1 = -1$, and the origin (see Fig. 9(b)). The total contribution to the LTS cost of the $2d$ points lying on the two vertical hyperplanes is zero, and the contribution of the $h - 2d$ points near the origin is roughly $(h-2d)\delta$. Let Δ_0 denote this quantity. Thus the total LTS cost in this case is at most Δ_0.

If, on the other hand, the points of P are affinely nondegenerate, let $\boldsymbol{\beta}$ denote the LTS optimal hyperplane. By applying Lemma 6(ii) in dimension $d-1$ to the smallest d squared residuals with respect to the intersection of $\boldsymbol{\beta}$ with $x_1 = 1$, it follows that (even ignoring all the other points) the LTS cost of $\boldsymbol{\beta}$ is at least $(d-(d-1))\alpha^2/M^{2(d-1)} = \alpha^2/M^{2(d-1)}$ for some positive α. Let Δ_1 denote this value. By selecting $\delta < \alpha^2/((1+\varepsilon_r)M^{2(d-1)}(h-2d))$, it follows that

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

Thus, the ratio of LTS costs in the cases where P is and is not affinely degenerate exceeds $1+\varepsilon_r$. Therefore, if we could approximate $\Delta^{(\mathrm{LTS})}(Q,h)$ to within a factor of $1+\varepsilon_r$, it would be possible to determine whether the original set P contains a degeneracy. Under the assumption of the hardness of affine degeneracy, this implies a lower bound of $\Omega((n')^{d-1}) = \Omega((n-h)^{d-1})$ on the worst-case running time of LTS. □

We can establish a similar result for the quantile approximation. Recall that ε_q denotes the allowed error in the quantile.

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

Proof As before, we prove the result for LTS only. The construction of the set Q is essentially the same as the construction of Theorem 6, except the value of δ (given below) will differ and the number of points on the vertical line through the origin is h, rather than $h-2d$. Let $h^- = h - \lfloor n\varepsilon_q \rfloor$, denote the number of inliers in the quantile approximation.

Let $(1/h)\Delta^{(\mathrm{LTS})}(Q,h)$ denote the minimum *average* squared residual for the points of Q assuming h inliers. Consider first that some d points of P lie on a common $(d-2)$-flat, and let $\boldsymbol{\beta}$ 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 $h-2d$ points near the origin is roughly $(h-2d)\delta$. Thus we have

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

On the other hand, if the points of P are affinely nondegenerate, let us consider the minimum average LTS cost assuming h^- inliers, $(1/h^-)\Delta^{(\mathrm{LTS})}(Q,h^-)$. Let $\boldsymbol{\beta}$ denote the hyperplane generating this minimum cost. As in Theorem 6, we may apply Lemma 6(ii), and by making δ sufficiently small, we may assume that all but $2(d-1)$ of the h^- closest squared residuals to $\boldsymbol{\beta}$ arise from the cluster of points that are close to the origin. The contribution of all of these points to the sum of squared residuals is at least $(h^- - 2(d-1))\delta$, and so we have

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

Let us assume that n and h are sufficiently large so that the effect of the ceiling in the definition of h^- is negligible. By our hypothesis that $\varepsilon_q < h/(dn)$, it follows that $h - n\varepsilon_q > h(d-1)/d$. Thus, for all sufficiently large n and h we have

$$\begin{aligned}\frac{1}{h^-}\Delta^{(\mathrm{LTS})}\big(Q,h^-\big) &\geq \delta\left(1-\frac{2(d-1)}{h^-}\right) > \delta\left(1-\frac{2(d-1)}{h-n\varepsilon_q}\right) \\ &> \delta\left(1-\frac{2(d-1)}{h(d-1)/d}\right) = \delta\left(1-\frac{2d}{h}\right) \geq \frac{1}{h}\Delta^{(\mathrm{LTS})}(Q,h).\end{aligned}$$

Thus, by computing a ε_q-quantile approximation to $\Delta^{(\mathrm{LTS})}(Q,h)$, we can determine whether the original set P contains an affine degeneracy. Under the assumption of the hardness of affine degeneracy, this implies a lower bound of $\Omega((n')^{d-1}) = \Omega((n-h)^{d-1})$ on the worst-case running time. □

5 Hybrid Approximation

The hardness results for residual and quantile approximations suggest that in dimensions three and higher it is unlikely that approximation schemes exist that run in linear or near linear time in n. It is natural to consider therefore what sort of approximation can be achieved in roughly linear time. In this section we show that it is possible to compute a hybrid approximation for LTS, that is, an approximation in which we relax both the requirements on the optimality of the sum of squared residuals and on the exact number of inliers to be used.

Let P be a set of n points in $\mathbb{R}^d$, and let h be the trimming parameter. Recall that in a hybrid approximation we are given two approximation parameters: an allowed residual error ε_r, and an allowed quantile error ε_q, where $0 < \varepsilon_r$ and $0 < \varepsilon_q < h/n$. Let $h^- = h - \lfloor n\varepsilon_q \rfloor$. An $(\varepsilon_r, \varepsilon_q)$-*hybrid approximation* is any hyperplane $\boldsymbol{\beta}$ such that

$$\frac{1}{h^-}\Delta_{\boldsymbol{\beta}}^{(\mathrm{LTS})}\big(P,h^-\big) \leq (1+\varepsilon_r)\frac{1}{h}\Delta^{(\mathrm{LTS})}(P,h).$$

It will simplify the presentation to describe the number of inliers in terms of quantiles and eliminate the normalizing factor. Let $\varphi = h/n$ and $\varphi^- = h^-/n$, and define $\overline{\Delta}_{\boldsymbol{\beta}}(P,\varphi)$ to be $(1/h)\Delta^{(\mathrm{LTS})}(P,h)$. Thus, our objective is to compute a hyperplane $\boldsymbol{\beta}$ such that

$$\overline{\Delta}_{\boldsymbol{\beta}}\big(P,\varphi^-\big) \leq (1+\varepsilon_r)\overline{\Delta}_{\boldsymbol{\beta}}(P,\varphi).$$

We begin by recalling the notion of an ε-approximation. Given a finite point set $X \subseteq \mathbb{R}^d$, a range space $(X, \mathcal{Q})$ of finite VC-dimension, and a parameter $\varepsilon > 0$ (not to be confused with ε_r), a point set Y is an ε-*approximation* for X if for any $S \in \mathcal{Q}$,

$$\left|\frac{|X\cap S|}{|X|} - \frac{|Y\cap S|}{|Y|}\right| \leq \varepsilon. \tag{3}$$

By standard results on ε-approximation, if $(X, \mathcal{Q})$ is of constant VC-dimension, then, for a suitable constant c (depending on the VC-dimension), a random sample of size $(c/\varepsilon^2)\log|X|$ is an ε-approximation with high probability [1, 22].

Let P be a set of points in $\mathbb{R}^d$, let $\varphi = h/n$ be the trimming quantile, and let $0 < \varepsilon < \varphi/2$ be a given parameter. Consider the range space $(P, \mathcal{Q})$ whose ranges are hyperplane slabs, that is, the region bounded between two parallel hyperplanes. Under the assumption that the dimension d is a constant, this range space has constant VC-dimension. (This follows directly from the fact that the range space of halfspaces has constant VC-dimension, and any range space consisting of the pairwise intersections of ranges of a space of VC-dimension d has VC-dimension at most $O(d \log d)$ [22].) In $O(n)$ time we draw a random sample of P of size $m = (c/\varepsilon^2)\log n$, for a suitable constant c. Let A denote this sample. With high probability, A is an ε-approximation for P. Let $h' = \lceil(\varphi - \varepsilon)m\rceil$, and solve the resulting LTS problem by the algorithm given in Theorem 2. Combining the sample and solving phases, the algorithm's total running time is $O(n + |A|^{d+1}) = O(n + ((1/\varepsilon^2)\log n)^{d+1}) = \widetilde{O}(n + 1/\varepsilon^{2(d+1)})$.

The main result of this section is that this algorithm yields a hybrid approximation with high probability. Before showing this, we present the following utility lemma, which relates the average LTS costs of any two sets that satisfy Eq. (3) above.

Lemma 10 *Consider two parameters* $0 < \varepsilon, \varphi \le 1$, *where* $\varphi > 2\varepsilon$, *and two sufficiently large finite point sets* $X, Y \subset \mathbb{R}^d$ *that satisfy Eq.* (3) *for the range space of hyperplane slabs. Then for any hyperplane* $\boldsymbol{\beta}$, $\overline{\Delta}_{\boldsymbol{\beta}}(Y, \varphi - \varepsilon) \le (1 + (2\varepsilon/\varphi)) \cdot \overline{\Delta}_{\boldsymbol{\beta}}(X, \varphi)$.

Proof Let $m = |X|$ and $n = |Y|$. For $1 \le i \le m$, let x_i denote the ith smallest absolute residual of X with respect to $\boldsymbol{\beta}$, and for $1 \le j \le n$, let y_j denote the jth smallest absolute residual of Y with respect to $\boldsymbol{\beta}$. Let X' be the multi-set formed by taking n copies of each element of X, and define Y' analogously by taking m copies of each element of Y. Letting $k = mn$, we have $|X'| = |Y'| = k$. Define x'_i and y'_j analogously for these two sets, respectively.

We are interested in quantiles of the above sets, and so for $0 < \gamma \le 1$, define $n_\gamma = \gamma \cdot n$, $m_\gamma = \gamma \cdot m$, and $k_\gamma = \gamma \cdot k$. It will simplify the presentation below to assume that, for $\gamma \in \{\varepsilon, \varphi\}$, these quantities are all integers. (If ε is sufficiently small relative to n and m, the slackness in our later inequalities will be sufficient to compensate for this bit of sloppiness.)

We begin by asserting that,

$$x'_{i+k_\varepsilon} \ge y'_i, \quad \text{for } 1 \le i \le k - k_\varepsilon. \tag{4}$$

To see this, suppose to the contrary that there existed i such that $x'_{i+k_\varepsilon} < y'_i$. Consider the slab S centered about $\boldsymbol{\beta}$ whose height is equal to $2x'_{i+k_\varepsilon}$. We have

$$\left|X' \cap S\right| \ge i + k_\varepsilon \quad \text{and} \quad \left|Y' \cap S\right| < i.$$

Since the points of X' come in multiplicities of size n and the points of Y' come in multiplicities of m, we have $|X \cap S| \ge (i + k_\varepsilon)/n = (i + \varepsilon mn)/n$ and $|Y \cap S| < i/m$. Since $k = mn$, this implies that

$$\frac{|X \cap S|}{|X|} \ge \frac{(i + \varepsilon nm)/n}{m} = \frac{i}{k} + \varepsilon \quad \text{and} \quad \frac{|Y \cap S|}{|Y|} < \frac{i}{k}.$$

Thus, we have

$$\left|\frac{|X\cap S|}{|X|}-\frac{|Y\cap S|}{|Y|}\right|>\varepsilon,$$

which contradicts our hypothesis regarding X and Y, and so establishes Eq. (4).

Returning to the proof of the lemma, we can express $\overline{\Delta}(X,\varphi)$ in terms of the residuals of the n-fold replicated points as

$$\overline{\Delta}_{\boldsymbol{\beta}}(X,\varphi)=\frac{1}{m_\varphi}\sum_{i=1}^{m_\varphi}x_i^2=\frac{1}{\varphi m}\sum_{i=1}^{\varphi m}x_i^2=\frac{1}{\varphi mn}\sum_{i=1}^{\varphi mn}\left(x_i'\right)^2=\frac{1}{k_\varphi}\sum_{i=1}^{k_\varphi}\left(x_i'\right)^2.$$

If we ignore the first k_ε terms of the sum and apply Eq. (4), we have

$$\overline{\Delta}_{\boldsymbol{\beta}}(X,\varphi)\geq\frac{1}{k_\varphi}\sum_{i=k_\varepsilon+1}^{k_\varphi}\left(x_i'\right)^2=\frac{1}{k_\varphi}\sum_{i=1}^{k_\varphi-k_\varepsilon}\left(x_{i+k_\varepsilon}'\right)^2\geq\frac{1}{k_\varphi}\sum_{i=1}^{k_\varphi-k_\varepsilon}\left(y_i'\right)^2.$$

In order to relate this to $\overline{\Delta}(Y,\varphi-\varepsilon)$, we first observe that $k_\varphi-k_\varepsilon=(\varphi-\varepsilon)k$, yielding

$$\begin{aligned}\overline{\Delta}_{\boldsymbol{\beta}}(X,\varphi)&\geq\frac{1}{k_\varphi}\sum_{i=1}^{k_\varphi-k_\varepsilon}\left(y_i'\right)^2=\left(\frac{k_\varphi-k_\varepsilon}{k_\varphi}\right)\frac{1}{k_\varphi-k_\varepsilon}\sum_{i=1}^{k_\varphi-k_\varepsilon}\left(y_i'\right)^2\\&=\left(1-\frac{\varepsilon}{\varphi}\right)\frac{1}{(\varphi-\varepsilon)k}\sum_{i=1}^{(\varphi-\varepsilon)k}\left(y_i'\right)^2.\end{aligned}$$

Recalling that $k=mn$, we then combine the copies into groups of size m to obtain

$$\begin{aligned}\overline{\Delta}_{\boldsymbol{\beta}}(X,\varphi)&\geq\left(1-\frac{\varepsilon}{\varphi}\right)\frac{1}{(\varphi-\varepsilon)k}\sum_{i=1}^{(\varphi-\varepsilon)n}my_i^2=\left(1-\frac{\varepsilon}{\varphi}\right)\frac{1}{(\varphi-\varepsilon)n}\sum_{i=1}^{(\varphi-\varepsilon)n}y_i^2\\&=\left(1-\frac{\varepsilon}{\varphi}\right)\frac{1}{n_{\varphi-\varepsilon}}\sum_{i=1}^{n_{\varphi-\varepsilon}}y_i^2=\left(1-\frac{\varepsilon}{\varphi}\right)\overline{\Delta}_{\boldsymbol{\beta}}(Y,\varphi-\varepsilon).\end{aligned}$$

In conclusion, we have $\overline{\Delta}_{\boldsymbol{\beta}}(Y,\varphi-\varepsilon)\leq\overline{\Delta}_{\boldsymbol{\beta}}(X,\varphi)/(1-(\varepsilon/\varphi))$. To complete the proof, we observe that if $0<\gamma<1/2$, then $1/(1-\gamma)<(1+2\gamma)$. Since $\varepsilon/\varphi<1/2$, we obtain $\overline{\Delta}_{\boldsymbol{\beta}}(Y,\varphi-\varepsilon)\leq(1+2\varepsilon/\varphi)\overline{\Delta}_{\boldsymbol{\beta}}(X,\varphi)$, as desired. □

It is interesting to note that the above lemma holds whether Y is an ε-approximation of X or vice versa. We will exploit this fact in our next result, which establishes that the above algorithm is an $(\varepsilon_r,\varepsilon_q)$-hybrid approximation, for suitable choices of ε_r and ε_q. We state the result for LTS, but the generalization to LTA is straightforward.

Theorem 8 *Given an n-element point set in $\mathbb{R}^d$ and parameters $0<\varepsilon_q<h/n$ and $0<\varepsilon_r$, let $\varepsilon_m=\min(\varepsilon_q,\varepsilon_r h/n)$. It is possible to compute an $(\varepsilon_r,\varepsilon_q)$-hybrid approximation to LTS and LTA in time $O(n+((1/\varepsilon_m^2)\log n)^{d+1})$ with high probability.*

Proof As before, we present the proof for LTS only. Recall that $\varphi = h/n$, and define $\varepsilon = \min(\varepsilon_r\varphi/12, \varepsilon_q/3)$. Run the aforementioned algorithm with this value of ε. The running time is $O(n + ((1/\varepsilon^2)\log n)^{d+1}) = O(n + ((1/\varepsilon_m^2)\log n)^{d+1})$. With high probability, the random sample chosen by the algorithm is an ε-approximation (for the value of ε given above) for the range space of hyperplane slabs. Under this assumption, let $\boldsymbol{\beta}$ denote the hyperplane returned by the algorithm, and let $\boldsymbol{\beta}^*$ denote the optimal LTS hyperplane for the point set P and quantile φ.

Since $\varepsilon \le \varepsilon_q/3 < \varphi/3$, we have $\varphi - \varepsilon > 2\varepsilon$. We may therefore apply Lemma 10 with $X \leftarrow A$, $Y \leftarrow P$, and $\varphi \leftarrow \varphi - \varepsilon$, to obtain

$$\overline{\Delta}_{\boldsymbol{\beta}}(P, \varphi - 2\varepsilon) = \overline{\Delta}_{\boldsymbol{\beta}}\bigl(P, (\varphi - \varepsilon) - \varepsilon\bigr) \le \left(1 + \frac{2\varepsilon}{\varphi - \varepsilon}\right)\overline{\Delta}_{\boldsymbol{\beta}}(A, \varphi - \varepsilon).$$

Since $\boldsymbol{\beta}$ is the optimal solution for A for quantile $\varphi - \varepsilon$, it follows that no other hyperplane has a smaller LTS cost, and in particular, $\overline{\Delta}_{\boldsymbol{\beta}}(A, \varphi - \varepsilon) \le \overline{\Delta}_{\boldsymbol{\beta}^*}(A, \varphi - \varepsilon)$. Applying Lemma 10 again, but with $X \leftarrow P$, $Y \leftarrow A$, and $\varphi \leftarrow \varphi$, we obtain

$$\overline{\Delta}_{\boldsymbol{\beta}^*}(A, \varphi - \varepsilon) \le \left(1 + \frac{2\varepsilon}{\varphi}\right)\overline{\Delta}_{\boldsymbol{\beta}^*}(P, \varphi).$$

Thus, we have

$$\overline{\Delta}_{\boldsymbol{\beta}}(P, \varphi - 2\varepsilon) \le \left(1 + \frac{2\varepsilon}{\varphi - \varepsilon}\right)\left(1 + \frac{2\varepsilon}{\varphi}\right)\overline{\Delta}_{\boldsymbol{\beta}^*}(P, \varphi).$$

Since $\varepsilon < \varphi/3$, we have $\varepsilon/(\varphi - \varepsilon) < 1$ and $\varphi - \varepsilon > 2\varphi/3$. Thus, the approximation factor is at most

$$\begin{aligned}\left(1 + \frac{2\varepsilon}{\varphi - \varepsilon}\right)\left(1 + \frac{2\varepsilon}{\varphi}\right) &\le \left(1 + \frac{2\varepsilon}{\varphi - \varepsilon}\right)^2 = \left(1 + \frac{4\varepsilon}{\varphi - \varepsilon} + \frac{4\varepsilon^2}{(\varphi - \varepsilon)^2}\right) \\ &< \left(1 + \frac{4\varepsilon}{\varphi - \varepsilon} + \frac{4\varepsilon}{\varphi - \varepsilon}\right) = \left(1 + \frac{8\varepsilon}{\varphi - \varepsilon}\right).\end{aligned}$$

Since $\varphi - \varepsilon > 2\varphi/3$ and $\varepsilon \le \varepsilon_r\varphi/12$, we have $8\varepsilon/(\varphi - \varepsilon) < 12\varepsilon/\varphi \le \varepsilon_r$. Therefore, the approximation factor is at most $(1 + \varepsilon_r)$, which completes the proof. □

6 Conclusions

In this paper we have established the first nontrivial bounds on the computational complexity of the LTS and LTA linear estimators, both exact and approximate, in spaces of constant dimension. As mentioned in the introduction, these estimators (especially LTS) are of great interest in the area of robust statistics. Overall, our results suggest that, except for hybrid approximations, these problems require significant time to solve both exactly and approximately, except small dimensions.

In particular, our exact algorithm for LTS runs in $O(n^{d+1})$ time, and, assuming that h is $\Theta(n)$ and ε is fixed, our residual approximation has a running time of

$\widetilde{O}(n^{d-1})$. Our hardness results suggest that significant improvements in these execution times may not be easy. Under the assumptions of the hardness of the affine degeneracy problem and $h = \Theta(n)$, we presented a lower bound of $\Omega(n^d)$ for the exact LTA problem (and it is natural to conjecture a similar lower bound for LTS). Under the same assumptions, we provided an $\Omega(n^{d-1})$ lower bound for residual and quantile approximations of both LTS and LTA.

The hardness of LTS and LTA stems from the need to satisfy hard bounds on both the cost and the allowed number of inliers. We have shown that, by relaxing both of these constraints, much more efficient solutions exist. In particular, we presented a randomized hybrid approximation to LTS and LTA which runs in time $\widetilde{O}(n + 1/\varepsilon^{2(d+1)})$ with high probability.

The obvious open problems that remain involve the computational complexity of the exact LTS and LTA problems. Can we reduce the upper bound from $O(n^{d+1})$ to $O(n^d)$? Failing this, can we establish a lower bound of $\Omega(n^{d+1})$ for the LTS and LTA problems? Another interesting question is whether our lower bound for the LTA problem can be adapted to hold for the LTS problem as well.

The results of this paper suggest that any algorithm, either exact or approximate, for LTS or LTA will have a worst-case running time that grows exponentially as a function of the dimension. Our lower bounds are based on rather pathological input sets, however. There do exist practical heuristics, such as the Fast-LTS heuristic of Rousseeuw and Van Driessen [30], which run efficiently and accurately on typical input sets. Unfortunately, this heuristic does not provide guarantees on the accuracy of the final result. This raises the question of whether there exists an algorithm that can provide guarantees the quality of the final results and also runs efficiently on "typical" input sets. In a companion paper [25], we consider this question and present a practical approximation algorithm for the LTS problem.

References

1. Alon, N., Spencer, J.: The Probabilistic Method. Wiley, New York (2000)
2. Baran, I., Demaine, E.D., Patrascu, M.: Subquadratic algorithms for 3SUM. Algorithmica **50**, 584–596 (2008)
3. Bernholt, T.: Computing the least median of squares estimator in time $O(n^d)$. In: Proc. Intl. Conf. on Computational Science and Its Applications. Springer LNCS, vol. 3480, pp. 697–706. Springer, Berlin (2005)
4. Bremner, D., Chen, D., Iacono, J., Langerman, S., Morin, P.: Output-sensitive algorithms for Tukey depth and related problems. Stat. Comput. **18**, 259–266 (2008)
5. Chan, T.M.: An optimal randomized algorithm for maximum Tukey depth. In: Proc. 15th Annu. ACM-SIAM Sympos. Discrete Algorithms, pp. 430–436 (2004)
6. Coakley, C.W., Hettmansperger, T.P.: A bounded influence, high breakdown, efficient regression estimator. J. Am. Stat. Assoc. **88**, 872–880 (1993)
7. Conte, S.D., De Boor, C.W.: Elementary Numerical Analysis: an Algorithmic Approach, 3rd edn. McGraw-Hill, New York (1980)
8. de Berg, M., Cheong, O., van Kreveld, M., Overmars, M.: Computational Geometry: Algorithms and Applications, 3rd edn. Springer, Berlin (2008)
9. Edelsbrunner, H.: Algorithms in Combinatorial Geometry. EATCS Monographs on Theoretical Computer Science, vol. 10. Springer, Heidelberg (1987)
10. Edelsbrunner, H., Guibas, L.J.: Topologically sweeping an arrangement. J. Comput. Syst. Sci. **38**, 165–194 (1989). Corrigendum in **42**, 249–251 (1991)

11. Edelsbrunner, H., Souvaine, D.L.: Computing median-of-squares regression lines and guided topological sweep. J. Am. Stat. Assoc. **85**, 115–119 (1990)
12. Erickson, J.: Lower bounds for linear satisfiability problems. Chic. J. Theor. Comput. Sci. **1999**(8) (1999). http://compgeom.cs.uiuc.edu/~jeffe/pubs/linsat.html
13. Erickson, J., Har-Peled, S., Mount, D.M.: On the least median square problem. Discrete Comput. Geom. **36**, 593–607 (2006)
14. Erickson, J., Seidel, R.: Better lower bounds on detecting affine and spherical degeneracies. Discrete Comput. Geom. **13**, 41–57 (1995)
15. Da Fonseca, G.D.: Fitting flats to points with outliers. Unpublished manuscript (2010)
16. Gajentaan, A., Overmars, M.H.: On a class of $O(n^2)$ problems in computational geometry. Comput. Geom. Theory Appl. **5**, 165–185 (1995)
17. Har-Peled, S.: How to get close to the median shape. Comput. Geom. Theory Appl. **36**, 39–51 (2007)
18. Hawkins, D.M.: The feasible solution algorithm for least trimmed squares regression. Comput. Stat. Data Anal. **17**, 185–196 (1994)
19. Herstein, I.N.: Topics in Algebra, 2nd edn. Wiley, New York (1975)
20. Hössjer, O.: Exact computation of the least trimmed squares estimate in simple linear regression. Comput. Stat. Data Anal. **19**, 265–282 (1995)
21. Langerman, S., Steiger, W.: The complexity of hyperplane depth in the plane. Discrete Comput. Geom. **30**, 299–309 (2003)
22. Matoušek, J.: Lectures on Discrete Geometry. Springer, Berlin (2002)
23. Miller, K., Ramaswami, S., Rousseeuw, P., Sellares, T., Souvaine, D., Streinu, I., Struyf, A.: Efficient computation of location depth contours by methods of combinatorial geometry. Stat. Comput. **13**, 153–162 (2003)
24. Mount, D.M., Netanyahu, N.S., Piatko, C., Silverman, R., Wu, A.Y.: Quantile approximation for robust statistical estimation and k-enclosing problems. Int. J. Comput. Geom. Appl. **10**, 593–608 (2000)
25. Mount, D.M., Netanyahu, N.S., Piatko, C., Silverman, R., Wu, A.Y.: A practical approximation algorithm for the LTS estimator. Technical report CS-TR-5000, University of Maryland (2011)
26. Mount, D.M., Netanyahu, N.S., Romanik, K.R., Silverman, R., Yu, A.Y.: A practical approximation algorithm for the LMS line estimator. Comput. Stat. Data Anal. **51**, 2461–2486 (2007)
27. Mount, D.M., Netanyahu, N.S., Zuck, E.: Analyzing the number of samples required for an approximate Monte-Carlo LMS line estimator. In: Hubert, M., Pison, G., Struyf, A., Van Aelst, S. (eds.) Theory and Applications of Recent Robust Methods. Statistics for Industry and Technology, pp. 207–219. Birkhäuser, Basel (2004)
28. Rousseeuw, P.J.: Least median-of-squares regression. J. Am. Stat. Assoc. **79**, 871–880 (1984)
29. Rousseeuw, P.J., Leroy, A.M.: Robust Regression and Outlier Detection. Wiley, New York (1987)
30. Rousseeuw, P.J., van Driessen, K.: Computing LTS regression for large data sets. Data Min. Knowl. Discov. **12**, 29–45 (2006)
31. Simpson, D.G., Ruppert, D., Carroll, R.J.: On one-step GM estimates and stability of influences in linear regression. J. Am. Stat. Assoc. **87**, 439–450 (1992)
32. Souvaine, D.L., Steele, J.M.: Time- and space-efficient algorithms for least median of squares regression. J. Am. Stat. Assoc. **82**, 794–801 (1987)
33. Yohai, V.J.: High breakdown-point and high efficiency robust estimates for regression. Ann. Stat. **15**, 642–656 (1987)