COMPUTATIONALLY EFFICIENT ALGORITHMS FOR HIGH-DIMENSIONAL ROBUST ESTIMATORS

David M. Mount[1]

Department of Computer Science and
Institute for Advanced Computer Studies
University of Maryland
College Park, MD 20742

Nathan S. Netanyahu[2]

Space Data and Computing Division
NASA Goddard Space Flight Center
Greenbelt, MD 20771
and
Center for Automation Research
University of Maryland
College Park, MD 20742

Abstract

Given a set of n points in a d-dimensional space that are hypothesized to lie on a hyperplane, robust statistical estimators have been recently proposed for the parameters of the model of best fit for the points. This paper presents computationally efficient algorithms for high-dimensional, median-based robust estimators (e.g., d-dimensional Theil-Sen and repeated median (RM) estimators). Basic techniques for achieving efficient algorithms in the 2-D case are generalized to yield efficient algorithms in higher dimensions. Specifically, the (randomized) algorithms presented are of (expected) running times $O(n^{d-1}\log n)$ and $O(n^{d-1}\log^2 n)$, respectively, for the d-dimensional Theil-Sen and RM estimators considered. Both algorithms are space optimal, i.e., they require $O(n)$ storage, for fixed d. We also briefly discuss an extension to nonlinear domain(s) of the methodology introduced.

1. Introduction

Consider a set of n distinct points in d-dimensional Euclidean space. A fundamental problem in the plane is that of finding a good line of fit for these points, and more generally in d-dimensional space to find a good $(d-1)$-dimensional hyperplane fitting these points. Classical approaches such as ordinary least squares, L_1, and L_∞ estimators have been well studied (see, e.g., [13], for an overview), but suffer from a sensitivity to outlying points. Recently there has been a growing interest in the use of ***robust estimators*** that alleviate this problem by considering medians rather than mean values. These estimators include the Theil [18] and Sen estimator [14], the repeated median (RM) estimator [15], and the least median squared (LMS) estimator [12]. Robustness is measured by the ***breakdown point*** of an estimator, which is roughly defined to be the fraction of outlying data points that may cause the estimator to take on an arbitrarily large aberrant value. (See [3], [13], for an exact definition.) For example, ordinary least squares, L_1, and L_∞ have a breakdown point of $1/n$, i.e., an asymptotic breakdown of zero. A number of papers have been written about the problem of computing various robust estimators in the plane. These address the Theil-Sen estimator (or more generally, the slope selection problem) [1], [6], [2], the RM estimator [8], [17], and the LMS estimator [16], [4].

In this paper we present algorithms for the computation of robust estimators in arbitrary fixed dimensional Euclidean space. In particular, we consider the computation of the Theil-Sen estimator and the RM estimator, as well as natural generalizations of these estimators to nonlinear domains. We show that the Theil-Sen estimator (a $1-0.5^{(1/d)}$ breakdown point estimator) can be computed in $O(n^{d-1}\log n)$ expected time and $O(n)$ space, and that the RM estimator (a 50% breakdown estimator) can be computed in $O(n^{d-1}\log^2 n)$ expected time and $O(n)$ space. These improve the naive $O(n^d)$ time algorithms for both estimators. More importantly, the naive algorithm for the Theil-Sen estimator requires $O(n^d)$ space, which is infeasible for all but small instances of this problem.

To the best of our knowledge, these are the first reported algorithms for robust estimators in dimensions greater than two. Throughout the discussion, we make the general position assumption that no $d+1$ points are coplanar, and that no d points form a hyperplane that is perpendicular to any of the axes.

---

[1] The support of the National Science Foundation under Grant CCR-89-08901 is gratefully acknowledged.

[2] This work was completed while the author held a National Research Council-NASA Goddard Research Associateship. The support of the Air Force Office of Scientific Research under Grant AFOSR-91-0239 (while a doctoral candidate at the University of Maryland) is gratefully acknowledged.

2. The High-Dimensional Theil-Sen Estimator

For clarity of presentation, we focus mainly on 3-D space. It should be noted, however, that all of our results generalize to the d-dimensional case. A three-dimensional version of the Theil-Sen estimator, otherwise known as the Oja-Niinimaa (1984) estimator, is defined as follows: Given a set of points $p_i(x_i, y_i, z_i)$ in E^3 $(i = 1, \ldots, n)$, we want to efficiently compute the plane equation $z = \hat{\theta}_1 x + \hat{\theta}_2 y + \hat{\theta}_3$, such that

$$\hat{\theta}_1 = \underset{1=i<j<k=n}{\text{med}} \frac{\Delta_1}{\Delta}, \quad \hat{\theta}_2 = \underset{1=i<j<k=n}{\text{med}} \frac{\Delta_2}{\Delta}, \quad \hat{\theta}_3 = \underset{1=i<j<k=n}{\text{med}} \frac{\Delta_3}{\Delta},$$

where

$$\Delta_1 = \begin{vmatrix} z_i & y_i & 1 \\ z_j & y_j & 1 \\ z_k & y_k & 1 \end{vmatrix}, \quad \Delta_2 = \begin{vmatrix} x_i & z_i & 1 \\ x_j & z_j & 1 \\ x_k & z_k & 1 \end{vmatrix}, \quad \Delta_3 = \begin{vmatrix} x_i & y_i & z_i \\ x_j & y_j & z_j \\ x_k & y_k & z_k \end{vmatrix}, \quad \Delta = \begin{vmatrix} x_i & y_i & 1 \\ x_j & y_j & 1 \\ x_k & y_k & 1 \end{vmatrix}.$$

Choosing the plane representation $z = \theta_1 x + \theta_2 y + \theta_3$, the above defined estimator amounts to taking the median values of θ_1, θ_2, and θ_3 over all $\binom{n}{3}$ planes defined by the n points.

A brute force computation requires $O(n^3)$ time and space, and is obviously infeasible. Instead, we propose to extend our methodology to 3-D according to the guidelines below, so that it results in an $O(n^2 \log n)$ time and $O(n)$ space algorithm.

Apply a dual transformation such that each point $p_i(x_i, y_i, z_i)$ is mapped to the plane $D(p_i) : z = x_i x + y_i y + z_i$, and vice versa. It can be easily shown that the problem of finding $\hat{\theta}_1$, $\hat{\theta}_2$, and $\hat{\theta}_3$ is now reduced to the problem of finding, up to a sign, the median x-coordinate, y-coordinate, and z-coordinate, respectively, among the $\binom{n}{3}$ intersection points of the planes $D(p_i)$. (Our general position assumption guarantees a *simple* arrangement in dual space, i.e., no degeneracies take place.)

Without loss of generality, we demonstrate below an efficient computation of the median x-coordinate. (The computation of the other median coordinates is very similar.) Analogously to our derivation for the slope selection problem [2], we maintain two x-values, x_{lo} and x_{hi}, $-\infty \leq x_{lo} \leq x_{hi} \leq +\infty$. Let $(x_{lo}, x_{hi}]$ denote the half-open, half-closed interval of points x, $x_{lo} < x \leq x_{hi}$, and let $I(x_{lo}, x_{hi}]$ denote the set of intersection ordinates (i.e., the x-coordinates of the intersection points between triplets of planes) in this interval. As in the two-dimensional case, we will maintain the invariant that the median intersection ordinate is in $I(x_{lo}, x_{hi}]$. Initially $x_{lo} = -\infty$, $x_{hi} = +\infty$, and $I(x_{lo}, x_{hi}]$ induces all $\binom{n}{3}$ intersection ordinates. (The assumptions/conventions made here are analogous to the ones made in [2].)

In principle, each stage of the algorithm should operate by contracting the interval $(x_{lo}, x_{hi}]$ to a subinterval $(x_{lo}', x_{hi}']$ which contains the median intersection ordinate. This could be performed by random sampling of intersection ordinates, analogously to the derivation in [2]. Also, analogously to the probability theory of our slope selection algorithm, the subinterval $(x_{lo}', x_{hi}']$ would be chosen so that, with high probability, the number of intersection ordinates in the subinterval, C', is expected to decrease by a factor of $O(1/\sqrt{n})$ with respect to the original number of ordinates, C. Since the initial interval contains $O(n^3)$ intersection ordinates, it follows, with high probability, that after *four* stages the number of remaining intersection ordinates in the interval would drop to $O(n)$.

To verify that the interval contraction is successful (i.e., that the ordinate of interest is, in fact, contained in the contracted subinterval), we apply an ordinate counting procedure to the subinterval $(x_{lo}, x_{lo}']$. Let L denote the number of intersections in this subinterval. The verification is done based on the values of C' and L. Should $(x_{lo}, x_{hi}']$ not contain the desired ordinate, the algorithm either makes a recursive call to $(x_{lo}, x_{lo}']$ (the left adjacent subinterval), or to $(x_{hi}', x_{hi}]$ (the right adjacent subinterval).

Once the number of ordinate intersections in the current interval is smaller than $c \cdot n$, for some constant $c \geq 1$, an $O(n \log n)$ enumeration procedure followed by a fast selection algorithm are invoked to determine the desired intersection ordinate.

A detailed description of the algorithm can be found in [9], [10]. The basic subproblem left to be solved is how to count/sample intersection ordinates. In Section 3 we show that this can be done in $O(n^2 \log n)$ time and $O(n)$ space in 3-D (and $O(n^{d-1} \log n)$ time and $O(n)$ space in d-dimensional space, for fixed d).

3. Intersection Counting and Sampling in Higher Dimensions

Extending the notion of intersection/inversion counting (and sampling) in the plane to efficient counting of intersection ordinates in higher dimensions requires special clarification. Inherent to such an extension is the notion of ***orientations*** [5], which are defined as follows: A sequence $(p_0, ..., p_d)$ of $d+1$ points in E^d, with $p_i = (x_{i1}, ..., x_{id})$ for all i, is said to have a ***positive orientation*** (denoted by $p_0 \cdots p_d > 0$) if $\det(x_{ij}) > 0$, where $x_{i0} = 1$ for each i. (Similarly, ***negative orientation*** and ***zero orientation*** are defined.) We embed the concept of (positive/negative) orientations in counting hyperplane intersections in d dimensions.

We first consider the 3-D case. Let H_i, H_j, and H_k denote three planes in general position in E^3, i.e., assume that they intersect at a finite (unique) point $(x_{ijk}, y_{ijk}, z_{ijk})$. Also assume their intersection with any plane $x = x'$ is not parallel to the z axis. For some x' let $l_i(x')$, $l_j(x')$, and $l_k(x')$ denote, respectively, the intersection lines of the above planes with the plane $x = x'$. (Obviously, each intersection line can be expressed as $z = z(y)$.) Note that for every two planes $x = x_{lo}$ and $x = x_{hi}$ $(x_{lo} < x_{hi})$,

$$l_i(x_{lo}) \parallel l_i(x_{hi}), \quad l_j(x_{lo}) \parallel l_j(x_{hi}), \text{ and } l_k(x_{lo}) \parallel l_k(x_{hi}).$$

To detect whether or not H_i, H_j, and H_k intersect in the interval $(x_{lo}, x_{hi}]$, we make the observations below (similar observations can be made with respect to the y and z axes). Detailed proofs can be found in [9], [10].

Observation 1: The three planes intersect in the given interval ***if and only if*** the ***orientation*** of the sequence of dual points (in the y-z plane) $D(l_i(x_{lo}))$, $D(l_j(x_{lo}))$, and $D(l_k(x_{lo}))$ is of ***opposite*** sign to the orientation of the sequence $D(l_i(x_{hi}))$, $D(l_j(x_{hi}))$, $D(l_k(x_{hi}))$.

Observation 2: $x_{lo} < x_{ijk} \le x_{hi}$ ***if and only if*** the order of which an oriented $l_i(x_{lo})$, say, intersects $l_j(x_{lo})$ and $l_k(x_{lo})$ is the ***reverse*** order of which a "consistently oriented" $l_i(x_{hi})$ intersects $l_j(x_{hi})$ and $l_k(x_{hi})$.

Observation 2 gives rise to the following procedure which counts the number of intersection ordinates in a given interval. Without loss of generality, reconsider the interval $(x_{lo}, x_{hi}]$. For each plane H_i, perform inversion counting on the list determined by the order of which an oriented $l_i(x_{hi})$ intersects the set of lines $\{l_j(x_{hi})\}$ relatively to the ***sorted*** order of which a "consistently oriented" $l_i(x_{lo})$ intersects the set of lines $\{l_j(x_{lo})\}$, for all $j > i$. Since inversion counting for k lines in the plane requires $O(k\log k)$ time, the total running time of the procedure proposed is $O(\sum_{k=1}^{n-1} k\log k) = O(n^2\log n)$ time. Also, since $O(k)$ reusable storage is required at each step of the above proposed procedure, the total space complexity is $O(n)$.

Notice that the above described computation is equivalent to the summation, over all H_i $(i = 1, 2, ..., n)$, of the number of intersections, $I(H_i)$, in $(x_{lo}, x_{hi}]$ of the line arrangement $\{H_i \cap H_j,\ i < j \le n\}$ ***projected*** onto the x-y plane.

For general fixed dimension d, let $\mathbf{H}$ denote the set of hyperplanes $\{H_i,\ i = 1, 2, ..., n\}$ in E^d. We describe a recursive function Ord_Count($\mathbf{H}$, n, d, C, x_{lo}, x_{hi}) which returns the number of intersection ordinates in the given interval, $(x_{lo}, x_{hi}]$. For each hyperplane H_i, Ord_Count invokes itself and returns the number of intersections associated with H_i in the given interval. At each recursive step the dimensionality d of the space considered is decremented and the problem is essentially reduced to finding the number of $n-1$ hyperplane intersections in E^{d-1}. Once $d = 2$, an efficient 2-D inversion counting procedure [2], is invoked. Analyzing the complexity of Ord_Count, we obtain the result below.

```
function Ord_Count(H, n, d, C, x_lo, x_hi);
begin
    C := 0;
    H' := ∅;
    if d = 2 then begin
        Invoke an efficient 2-D inversion counting procedure;
        return(C)
    end;
    for i := 1 to n do begin
        for j := i + 1 to n do begin
            H_j' := H_i ∩ H_j;
            H' := H' ∪ {H_j'}
        end;
        Ord_Count(H', n - 1, d - 1, C_i, x_lo, x_hi);
        C := C + C_i
    end;
    return(C)
end;
```

Lemma 1: The number of intersections, in a given interval, of n hyperplanes in E^d can be computed in $O(n^{d-1}\log n)$ time and $O(n)$ space, for fixed d.

Proof: By induction on d, and based on the fact that $T(n, 2) = O(n\log n)$ (see [9], [10], for a detailed proof). ∎

Once the number of (plane) intersections, C, in the given interval is known, the task of sampling $n < C$ intersections can be embedded in the above counting procedure in a very similar manner to that of the two-dimensional case [2]. Again, we generate a set IS of n random integers distributed uniformly in the range from 1 to C (allowing duplicates), and sort these integers. Next, we rerun Ord_Count, but in addition to updating an intersection counter, for each $IS\_Indx \in IS$, we sample the IS_Indx-th x-coordinate. (Actual sampling of the ordinate of interest of an intersection takes place only at the recursive level where $d = 2$.) The only difference with respect to the 2-D case is that here, the inversion count associated with H_i should be offset by the current intersection count.

Based on Lemma 1 and the discussion above, we obtain the following main result (see [9], [10], for a detailed proof):

Theorem 2: The d-dimensional Theil-Sen estimator can be computed in expected $O(n^{d-1}\log n)$ time and $O(n)$ space, for fixed d.

4. The High-Dimensional Repeated Median (RM) Estimator

In this section we present an algorithm for computing the d-dimensional RM estimator, a 50% breakdown point estimator. Again, we first consider the 3-D case. The three-dimensional RM estimator fits a plane equation $z = \hat{\theta}_1 x + \hat{\theta}_2 y + \hat{\theta}_3$ to a given set of n points in E^3, such that

$$\hat{\theta}_p = \underset{i}{\text{med}}\ \underset{j \neq i}{\text{med}}\ \underset{k \neq i,j}{\text{med}}\ \frac{\Delta_p}{\Delta},$$

where Δ_p ($p = 1, 2, 3$) and Δ are defined in Section 2. The meaning of the above definition is the following: Fix a point i and a point $j \neq i$. Compute the plane coefficients determined by all triplets of points i, j, k, such that $k \neq i, j$ and store their median values. Repeat the process for all points $j \neq i$, and (for each coefficient) obtain the median value of the (median) values stored in the previous stage. Finally, repeat the process over all points i to obtain the RM estimate.

A brute force approach results in an $O(n^3)$ time algorithm. A straightforward improvement of the above bound results from the following approach:

Apply a three-dimensional dual transformation as in the previous subsection. The problem is now reduced to finding

$$\underset{i}{\text{med}}\ \underset{j \neq i}{\text{med}}\ \underset{k \neq i,j}{\text{med}}\ (H_i \cap H_j \cap H_k)_{ordinate}$$

in dual space, where $(H_i \cap H_j \cap H_k)_{ordinate}$ denotes the ordinate of interest (i.e., x, y, or z) of the intersection point of H_i, H_j, and H_k.

For each (dual) plane $D(p_i) \equiv H_i$, invoke an efficient two-dimensional RM procedure to compute, for example,

$$\underset{j \neq i}{\text{med}}\ \underset{k \neq i,j}{\text{med}}\ (H_i \cap H_j \cap H_k)_x.$$

Notice that this step can be carried out by considering, for example, the projection of $\{H_i \cap H_j, j \neq i\}$ onto the x-y plane and invoking an efficient two-dimensional RM algorithm with respect to the x-coordinate. The computation of the y and z coordinates can be performed similarly.

Take the median value of the n estimates computed in the previous step to obtain the corresponding $\hat{\theta}_p$.

The above outlined procedure draws strongly on the availability of an efficient 2-D RM algorithm. Based on the algorithms presented in [8], [17], we state the following result:

Lemma 3: The 3-D RM estimator can be computed in expected $O(n^2\log^2 n)$ time and $O(n)$ space.

By definition, a recursive procedure, similar to Ord_Count, can be derived for the computation of the d-dimensional RM estimator. See [9], [10], for a detailed discussion, which leads to the following result:

Theorem 4: The d-dimensional RM estimator can be computed in expected $O(n^{d-1}\log^2 n)$ time and $O(n)$ space, for fixed d.

5. An Extension to Nonlinear Domain(s)

Intuitively assuming that (statistical) properties of the estimators proposed in the previous sections are retained, to some extent, in their "nonlinear counterparts", the question of efficient computation remains a valid issue of interest. We briefly discuss *circular arc fitting*, and argue the applicability of our methods to a nonlinear domain.

Consider a given set of points (x_i, y_i), $i = 1, ..., n$, that are hypothesized to lie on a circular arc. We want to find the circle equation $(x - \hat{a})^2 + (y - \hat{b})^2 = \hat{r}^2$, that fits the data according to the following definitions:

(1) A nonlinear (3-D) Theil-Sen variant (for circular arc fitting) computes for each triplet (i, j, k), $1 \leq i < j < k \leq n$, the parameters of the corresponding circle (i.e., $a_{i,j,k}$, $b_{i,j,k}$, and $r_{i,j,k}$). The ultimate parameter estimates are given by the median values, over all $\binom{n}{3}$ elements, of each of the above sets; $\hat{a} = \text{med}\, a_{i,j,k}$, $\hat{b} = \text{med}\, b_{i,j,k}$, and $\hat{r} = \text{med}\, r_{i,j,k}$.

(2) An RM circular arc estimator would compute $a_{i,j,k}$, $b_{i,j,k}$, and $r_{i,j,k}$ for all triplets (i, j, k), such that $i \neq j \neq k$. The center's estimated coordinates would be given by $\hat{a} = \underset{i}{\text{med}}\, \underset{j \neq i}{\text{med}}\, \underset{k \neq i,j}{\text{med}}\, a_{i,j,k}$, $\hat{b} = \underset{i}{\text{med}}\, \underset{j \neq i}{\text{med}}\, \underset{k \neq i,j}{\text{med}}\, b_{i,j,k}$, and the estimated radius would be given by $\hat{r} = \underset{i}{\text{med}}\, \underset{j \neq i}{\text{med}}\, \underset{k \neq i,j}{\text{med}}\, r_{i,j,k}$.

Extending our methodology to a nonlinear domain may not always be possible. One difficulty, for example, lies in the fact that a nonlinear dual transformation may not always help determine the number of intersection ordinates in a given interval, as in the linear case. (See [9], [10], for a more detailed explanation.) In general, counting intersections (without explicitly enumerating them) becomes much less straightforward.

Let us first clarify the notion of a "nonlinear dual transformation" in the context of our model fitting methodology. For some m and n, let Φ denote a functional mapping a vector of m parameters in E^m to a function in E^n. For example, the circle functional $\Phi[a, b, r]$ maps the 3-vector (a, b, r) in E^3 to the function $\Phi[a, b, r](x, y) = (x - a)^2 + (y - b)^2 - r^2$ acting on E^2, whose zeroes are the points on the circle of radius r centered at (a, b). Now the dual defined by Φ is a mapping Φ^* from E^n to E^m, where for the circle functional

$$\Phi^*(x, y) = \{(a, b, r) \mid \Phi[a, b, r](x, y) = 0 \text{ and } r \geq 0\},$$

i.e., $\Phi^*(x, y)$ is the set of all points (a, b, r) in E^3 such that (x, y) lies on the circle whose center is (a, b) and whose radius is r.

We now argue that our methodology in dual space (i.e., efficient intersection ordinate counting and sampling, interval contraction, etc.) is applicable to circular arc fitting as well. The reason for this is that by definition of the dual transformation introduced, there is a one to one correspondence between a circle in primal space, uniquely determined by three (distinct) points, and a point in dual space, uniquely determined by three dual circular cones (of the points). This property enables us to count intersection ordinates (of triplets of circular cones) in a given interval, and to compute $\hat{a}$, $\hat{b}$ in a similar manner to the linear three-dimensional case (Section 2). To estimate the radius, we may apply at this point an alternative *hierarchical* computation: Select the median over all r_i values given by $r_i^2 = (x_i - \hat{a})^2 + (y_i - \hat{b})^2$. Due to space limitation, we omit the details of the algorithm, which can be found in [9], [10].

Based on our 2-D results, and in view of the above hierarchical scheme, we arrive at the following:

Theorem 5: The (hierarchical) Theil-Sen and RM circular arc estimators can be computed in (expected) $O(n^2 \log n)$ and $O(n^2 \log^2 n)$ time, respectively, and $O(n)$ space.

6. Conclusions

Several computationally efficient algorithms for high-dimensional robust estimators were presented in this abstract. Specifically, we extended the algorithmic methodology pursued with respect to the Theil-Sen and the RM line estimators to their high-dimensional counterparts and demonstrated how to apply such extensions to nonlinear domains (e.g., circular arc fitting).

The main characteristics of our extended methodology are highlighted by the following points:

- Our algorithms always terminate and return correct computational results, within machine's precision.
- Space complexity for all cases considered is optimal (i.e., $O(n)$).
- (Expected) time complexity results are better than those currently known. In fact, the results claimed for the RM estimators suggest, for the first time, algorithms better than $O(n^d)$, $d \geq 3$, for estimators having 50% breakdown points. Moreover, in view of ongoing work [7], the time bounds for the RM estimator could be further improved by a log$n$ factor.

- The algorithms described are simple, implementable, and practical.
- In principle, our proposed methodology is applicable to nonlinear domains, although the conditions for this are not fully characterized. It appears that circular arc fitting, a problem having significant industrial applications, can be efficiently dealt with by the methodology introduced here.

Acknowledgements

We would like to thank Michael Dillencourt and Azriel Rosenfeld for their collaboration and helpful comments.

References

[1] R. Cole, J.S. Salowe, W.L. Steiger, and E. Szemerédi (1989), An Optimal-Time Algorithm for Slope Selection, *SIAM Journal on Computing*, **18**, pp. 792–810.

[2] M.B. Dillencourt, D.M. Mount, and N.S. Netanyahu (1992), A Randomized Algorithm for Slope Selection, *International Journal of Computational Geometry and Applications*, **2**, pp. 1–27.

[3] D.L. Donoho and P.J. Huber (1983), The notion of breakdown point, in *A Festschrift for Erich L. Lehman*, eds. P.J. Bickel, K. Doksun, and J.L. Hodges, Jr., Wadsworth International Group, Belmont, California, pp. 157–184.

[4] H. Edelsbrunner and D.L. Souvaine (1990), Computing Median-of-Squares Regression Lines and Guided Topological Sweep, *Journal of the American Statistical Association*, **85**, pp. 115–119.

[5] J.E. Goodman and R. Pollack (1983), Multidimensional Sorting, *SIAM Journal on Computing*, **12**, pp. 484–507.

[6] J. Matoušek (1991), Randomized Optimal Algorithm for Slope Selection, *Information Processing Letters*, **39**, pp. 183–187.

[7] J. Matoušek, D.M. Mount, and N.S. Netanyahu (1992), Efficient Randomized Algorithms for the Repeated Median Line Estimator, in preparation.

[8] D.M. Mount and N.S. Netanyahu (1991), Computationally Efficient Algorithms for a Highly Robust Line Estimator, CS-TR-2816, Center for Automation Research, University of Maryland, December 1991.

[9] D.M. Mount and N.S. Netanyahu (1992), Computationally Efficient Algorithms for High-Dimensional Robust Estimators, Technical Report in preparation, Center for Automation Research, University of Maryland.

[10] N.S. Netanyahu (1991), *Computationally Efficient Algorithms for Robust Estimation*, Ph.D. Thesis, Department of Computer Science, University of Maryland.

[11] H. Oja and Niinimaa (1984), On Robust Estimation of Regression Coefficients, Research Report, Department of Applied Mathematics and Statistics, University of Oulu, Finland.

[12] P.J. Rousseeuw (1984), Least Median of Squares Regression, *Journal of the American Statistical Association*, **79**, pp. 871–880.

[13] P.J. Rousseeuw and A.M. Leroy (1987), *Robust Regression and Outlier Detection*, Wiley, New York.

[14] P.K. Sen (1968), Estimates of the Regression Coefficient Based on Kendall's Tau, *Journal of the American Statistical Association*, **63**, pp. 1379–1389.

[15] A.F. Siegel (1982), Robust Regression Using Repeated Medians, *Biometrika*, **69**, pp. 242–244.

[16] D.L. Souvaine and J.M. Steele (1987), Time- and Space-Efficient Algorithms for Least Median of Squares Regression, *Journal of the American Statistical Association*, **82**, pp. 794–801.

[17] A. Stein and M. Werman (1992), Finding the Repeated Median Regression Line, in *Proceedings of the Third Annual Symposium on Discrete Algorithms*, Orlando, Florida, January 1992, pp. 409–413.

[18] H. Theil (1950), A Rank-Invariant Method of Linear and Polynomial Regression Analysis (Parts 1–3), *Nederlandse Akademie Wetenschappen Series A*, **53**, pp. 386–392, 521–525, 1397–1412.