Discrete Applied Mathematics 14 (1986) 93–100
North-Holland

ALGORITHMS AND COMPLEXITY FOR LEAST MEDIAN OF SQUARES REGRESSION

J.M. STEELE*
Applied and Computational Mathematics, Princeton University, Princeton, NJ 08544, USA

W.L. STEIGER
Department of Computer Science, Rutgers University, New Brunswick, NJ 08903, USA

Received 20 March 1985
Revised 15 August 1985

Given n points $\{(x_i, y_i)\}$ in the plane we study the problem of calculating the least median of squares regression line. This involves the study of the function $f(\alpha, \beta) = \text{median}(|y_i - (\alpha + \beta x_i)|)$; it is piecewise linear and can have a quadratic number of local minima. Several algorithms that locate a minimizer of f are presented. The best of these has time complexity $O(n^3)$ in the worst case. Our most practical algorithm appears to be one which has worst case behavior of $O(n^3 \log(n))$, but we provide a probabilistic speed-up of this algorithm which appears to have expected time complexity of $O((n \log(n))^2)$.

1. Introduction

Given n points (x_i, y_i) in the plane our computational objective is to find a line which is optimal with respect to the criterion of minimizing the median of the squared residuals. Equivalently, because $|t|^2$ is monotone in $|t|$, we wish to find values α^* and β^* which minimize the objective function

$$(1) \qquad f(\alpha, \beta) = \text{median}(|y_i - (\alpha + \beta x_i)|).$$

We follow Rousseeuw [5] in referring to the line $y = \alpha^* + \beta^* x$ a least median of squares fit because of the compelling analogy with the familiar term, least (sums of) squares.

One reason for the interest in the least median of squares regression procedure is that the method has high *breakdown point*. Roughly speaking, the breakdown point of a statistical estimator is the smallest percentage of contamination which may cause the estimator to take on arbitarily large values. This concept was first studied explicitly by Hampel [2] who extended an earlier concept of Hodges [3]. It now appears that breakdown point is emerging as one of the basic criteria for judging the robustness of an estimator (see e.g. Donoho and Huber [1]). One virtue of least median of squares regression is that it has a breakdown point of 50%, and this is obviously the highest possible breakdown point of any reasonable estimator (see

* Research supported in part by NSF grant DMS-8414069.

Rousseeuw [5], which also mentions an algorithm and cites Leroy and Rousseeuw [4]).

In the next section we characterize local minima of f and thus give a necessary condition for global minimizers. This converts the problem of minimizing f into a discrete optimization. In the course of studying the geometry of f, we will show that f can have $O(n^2)$ local minima. This high number of local minima makes any approach with gradient methods (or even naive line search) a risky proposition, but the characterization of the local minima which we exhibit gives an easy route to exact enumerative minimization.

Two algorithms exploiting our characterization are given. The first of these has time complexity $O(n^3)$ and the second has complexity $O(n^3 \log(n))$. The reason for putting forth the second algorithm is that in the last section it is modified to provide a probabilistic algorithm which may be the most practical one available. On the basis of modest simulation and a reasonably persuasive heuristic argument, it is conjectured to have complexity $O([n \log(n)]^2)$ on the average.

In all the considerations which follow, we suppose that the initial *data* (x_i, y_i), $1 \le i \le n$, *are in general position*. More explicitly, we assume no three points are colinear, and no two pairs of points determine parallel lines. We also need to disambiguate the notion of median in even sample sizes. If $u_1 \le \cdots \le u_n$, our convention, based on convenience, will be to take u_m as the median, where $m = 1 + \lfloor n/2 \rfloor$. This reduces to the middle value for n odd and to the *high median*, or larger of the two middle values, if n is even. One consequence of these conventions is that $f > 0$ when n is larger than 3.

2. Structure and algorithms

Given α and β, the line $l_{\alpha,\beta} = \{(x, y): y = \alpha + \beta x\}$ defines residuals $r_i(\alpha, \beta) \equiv y_i - (\alpha + \beta x_i)$. When there is no confusion we will simply write r_i. We say that $l_{\alpha,\beta}$ '*bisects*' the three distinct points (x_{i_j}, y_{i_j}), $j = 1, 2, 3$, if the residuals r_{i_j} are all the same size and not all have the same sign. If also $x_{i_1} < x_{i_2} < x_{i_3}$ and $r_{i_1} = -r_{i_2} = r_{i_3}$, we say that $l_{\alpha,\beta}$ *equioscillates* with respect to the points. It is easy to see that each triple of points is 'bisected' by three lines, one of which equioscillates.

We begin the study of the combinatorics of minimizing f by noting that any line $l_{\alpha,\beta}$ partitions $\{1, \ldots, n\}$ into three sets,

$$B_{\alpha,\beta} = \{i: |r_i(\alpha, \beta)| > f(\alpha, \beta)\},$$

$$M_{\alpha,\beta} = \{i: |r_i(\alpha, \beta)| = f(\alpha, \beta)\},$$

$$S_{\alpha,\beta} = \{i: |r_i(\alpha, \beta)| < f(\alpha, \beta)\}$$

of **b**ig, **m**edian, and **s**mall residuals, respectively. When there is no confusion we will drop the subscripts. The following simple result articulates a necessary and sufficient condition for local optimality.

Main Lemma. *The pair* (α^*, β^*) *is a local minimum of* f *if and only if the following conditions hold:*

(i) $|M_{\alpha^*,\beta^*}| = 3$,

(ii) l_{α^*,β^*} *equioscillates with respect to the points indexed by* M_{α^*,β^*},

(iii) $|B_{\alpha^*,\beta^*}| - |S_{\alpha^*,\beta^*}| \geq 1$.

Proof. Fix β and define $r_i(\alpha)$ by $r_i(\alpha) = y_i - (\alpha + \beta x_i)$. Suppose that $|M| = 1$ and that $|r_p(\alpha)|$ is the median-sized residual. If $r_p(\alpha)$ is positive, $|r_p(\alpha)|$ decreases as α increases; and, if it is negative, as α decreases. As α changes, $|r_p(\alpha)|$ remains the median until that point α' at which another residual, r_q, first attains equal size. We may assume $r_p(\alpha')r_q(\alpha') < 0$. Otherwise, we could change α' and decrease both $|r_p(\alpha')|$ and $|r_q(\alpha')|$. Hence, at a local optimum we always have two median sized residuals with opposite signs.

Now we show that (i) is necessary. Suppose (α, β) is a local optimum, and that there are points, say p and q, whose residuals from $l_{\alpha,\beta}$ are median sized and of opposite sign. Note that $l_{\alpha,\beta}$ contains the point $\boldsymbol{\mu} = [(x_p, y_p) + (x_q, y_q)]/2$. If no other data point has residual size $|r_p|$, $l_{\alpha,\beta}$ may be rotated about $\boldsymbol{\mu}$ so that both $|r_p|$ and $|r_q|$ decrease (equally). They will remain median sized residuals up to that point when another residual, say r_s, first attains equal size. This proves that (i) is necessary and also that a line defined by a local optimum will 'bisect' the points indexed by M.

In fact if (α, β) is a local optimum, $l_{\alpha,\beta}$ must equioscillate. If not, it already passes through the midpoint of two 'bisected' points that have opposite sign. By rotating about this midpoint, *all* three median sized residuals may be reduced in size. Since at least one of these residuals must remain median sized, (ii) is necessary for a local minimum.

To see that (iii) is necessary, suppose that (α, β) is a local minimizer of f. Then (i) and (ii) hold, so there are p, q, and $t \in M$ with $x_p < x_q < x_t$ and $r_p = -r_q = r_t$. The line $l_{\alpha,\beta}$ may be rotated about the midpoint of (x_p, y_p) and (x_q, y_q) so that both $|r_p|$ and $|r_q|$ decrease while $|r_t|$ increases. Let (α', β') denote the parameters of the rotated line. By the continuity of $y_i - (\alpha + \beta x_i)$, if (α', β') is close enough to (α, β), $|r_p(\alpha', \beta')|$, $|r_q(\alpha', \beta')|$, and $|r_t(\alpha', \beta')|$ are smaller than all $|r_i(\alpha', \beta')|$, $i \in B_{\alpha,\beta}$ and bigger than all $|r_i(\alpha', \beta')|$, $i \in S_{\alpha,\beta}$. Since $f(\alpha, \beta) < f(\alpha', \beta')$ we must have $M_{\alpha',\beta'} = t$, $S_{\alpha',\beta'} = S_{\alpha,\beta} \cup \{p, q\}$, and $B_{\alpha',\beta'} = B_{\alpha,\beta}$. The sizes of $B_{\alpha',\beta'}$ and $S_{\alpha',\beta'}$ can differ by at most 1, because $|r_t(\alpha', \beta')|$ is a unique median, and these facts now imply (iii).

Finally, (i)–(iii) are clearly sufficient. Otherwise there would exist (α', β') arbitrarily close to (α, β) for which $f(\alpha', \beta') < f(\alpha, \beta)$. But this is impossible because $l_{\alpha',\beta'}$ cannot equioscillate. □

Besides characterizing local minimizers of f, the lemma turns the problem into a discrete one. Only lines that equioscillate with respect to some triple of data points need be considered. The following algorithm checks $f(\alpha, \beta)$ on such a line for each triple. By the lemma, the equioscillating line with the smallest value of f will determine (α^*, β^*).

Crude Algorithm

- $d^* \leftarrow \infty$
- $(\alpha^*, \beta^*) \leftarrow (0, 0)$
- for each distinct triple i, j, k
  - renumber points so $x_i < x_j < x_k$
  - $\beta \leftarrow (y_i - y_k)/(x_i - x_k)$; $\alpha \leftarrow [y_j + y_k - \beta(x_j + x_k)]/2$
  - $d_{i,j,k} \leftarrow f(\alpha, \beta)$
  - if $d_{i,j,k} < d^*$, $d^* \leftarrow d_{i,j,k}$ and $(\alpha^*, \beta^*) \leftarrow (\alpha, \beta)$
  - restore points to original numbering

This may be the first finite, exact algorithm for least median of squares fits. Its crudeness is reflected in its complexity of $O(n^4)$, even if one uses a linear cost method to obtain median$(|y_i - (\alpha + \beta x_i)|)$ for each of the $n(n-1)(n-2)/6$ lines mentioned. It would not be practical for even as few as 50 points.

This crude algorithm nevertheless contains the germ of a useful method with a faster running time. The first step is to focus on lines through pairs of points. Let $\beta_{i,j} = (y_i - y_j)/(x_i - x_j)$. Because equioscillation is necessary, the optimal β^* must equal $\beta_{i,j}$ for some $i \neq j$. So regard β in (1) as fixed, and consider the reduced problem of finding a minimizer of

$$(2) \qquad g(\alpha) = \text{median}(|z_i - \alpha|)$$

where $z_i = y_i - \beta x_i$. If the z_i are in increasing order, then a little thought convinces one that the minimum value of $g(\alpha)$ is equal to the length of the smallest interval that contains half of the z_i. The minimizer α^*, is then the midpoint of this interval. Thus, writing $m(n)$ for the median of $\{1, \ldots, n\}$,

$$(3) \qquad \begin{aligned} &\text{if } z_{p+m(n)} - z_p = \min[(z_{j+m(n)} - z_j),\ j = 1, \ldots, m(n)], \text{ then} \\ &\min[g(a)] = z_{p+m(n)} - z_p, \qquad \alpha^* = (z_{p+m(n)} + z_p)/2. \end{aligned}$$

This formula can also be derived by noting that the graph of g is continuous and piecewise linear. Each piece has slope ± 1, the slopes change sign at each z_i, $g(z_1) = z_{m(n)} - z_1$, and $g' < 0$ for $z < z_1$.

For any β, let $l_{\alpha^*,\beta}$ denote the best possible line with slope β. This suggests the following

Algorithm 1

- $(\alpha^*, \beta^*) \leftarrow (0, 0)$
- $d^* \leftarrow \infty$
- for each distinct pair r, s
  - $\beta \leftarrow \beta_{r,s}$
  - $z_i \leftarrow y_i - \beta x_i$, $i = 1, \ldots, n$
  - sort the z_i
  - if $z_{p+m(n)} - z_p = \min(z_{j+m(n)} - z_j)$, $d \leftarrow z_{p+m(n)} - z_p$

- $\alpha \leftarrow (z_{p+m(n)} + z_p)/2$
- if $d < d^*$, $d^* \leftarrow d$ and $(\alpha^*, \beta^*) \leftarrow (\alpha, \beta)$

The complexity within the loop is $O(n \log(n))$ due to the sort, and therefore Algorithm 1 has complexity $O(n^3 \log(n))$ overall. This algorithm was used by Leroy and Rousseeuw [1] for small n although it was not known to be exact. For larger n they obtain an approximation to (α^*, β^*) by modifying Algorithm 1 to apply the loop to random choices of r and s, rather than to all pairs.

To go beyond Algorithm 1, it is necessary to explicate more of the geometry than was brought out in the Main Lemma. In condition (ii), let us weaken equioscillation to the requirement that $l_{\alpha,\beta}$ only 'bisect' points indexed by M. If (α, β) also satisfies conditions (i) and (iii), we say it is a '*possible local minimum*'. The next result identifies and counts the 'possible local minima'.

Little Lemma. *There are exactly* $n(n-1)/2$ *choices of* (α, β) *such that* $l_{\alpha,\beta}$ *'bisects' the points in* $M_{\alpha,\beta}$ *and the conditions* (i) *and* (iii) *hold.*

Proof. Consider the line $L_{j,k}$ given by

$$y = \beta_{j,k}(x - x_j) + y_j$$

and which contains points j and k. There are $n-2$ nonzero residuals from this line. Either at least $m = \lfloor (n-2)/2 \rfloor$ are positive, or m are negative. Assume there are m positive ones (the negative case can be handled similarly). The first step is to find p such that r_p is the m-th smallest positive residual from $L_{j,k}$.

Now consider the line $L^+_{j,k}$ parallel to $L_{j,k}$ and containing (x_p, y_p). There are three points **on** $L_{j,k}$ and $L^+_{j,k}$, $m-1$ points **between** the lines, and $q = n-m-2$ points **not between** them. Consider the line $L^*_{j,k}$ which passes halfway between $L_{j,k}$ and $L^+_{j,k}$. The construction assures that it 'bisects' points j, k, and p, that j, k, p are in M, and that (i) and (iii) hold. □

It is natural to ask how many of the 'possible local minima' are in fact, local minima. The answer depends on the particular configuration of data points. In fact for the 'bisecting' line $L^*_{j,k}$ to equioscillate, all that is required is that x_p lie between x_j and x_k. Using this observation, we exhibit a configuration where f has a quadratic number of local minima.

Let A denote the $\lfloor n/4 \rfloor$ points with the smallest x-coordinates and B, the $\lfloor n/4 \rfloor$ points with the largest x-coordinates. The remaining points have 'middle half' x-coordinates. They will be assigned y-coordinates larger than those of any of the points in A or B. Now, if point $j \in A$ and point $k \in B$ are used to define $L_{j,k}$, as in the proof of the little lemma, the point p will not be in A or B. The line $L^*_{j,k}$ will equioscillate and therefore define a local minimum of f. There are $n^2/16$ such lines, one for each pair j, k in A and B, respectively. This proves the interesting

Corollary. *f can have $O(n^2)$ local minima.*

The proof of the little lemma suggests another algorithm. Given j and k, compute the line $L^*_{j,k}$ given by

$$y = \beta_{j,k}x + [y_i + y_p - \beta_{j,k}(x_j + x_p)]/2.$$

The number p corresponds to the m-th least positive (or negative) residual from the line through points j and k, and may be obtained in linear time. For some pair j, k, one of these lines will define the minimizer in (1). It may be found in time $O(n^3)$.

Algorithm 2

- $(\alpha^*, \beta^*) \leftarrow (0, 0)$
- $d^* \leftarrow \infty$
- $m \leftarrow \lfloor (n-2)/2 \rfloor$
- for each pair r, s
  - $d \leftarrow \infty$
  - $\beta \leftarrow \beta_{r,s}$
  - $z_i \leftarrow y_i - y_r - \beta(x_i - x_r),\ i = 1, \ldots, n$
  - $\Pi \leftarrow \{i: z_i > 0\},\ N \leftarrow \{i: z_i < 0\}$
  - $z_p \leftarrow$ either m-th smallest $z_i,\ i \in \Pi$ or m-th largest $z_i,\ i \in N$, according to the smaller absolute value
  - $\alpha \leftarrow [y_r + y_p - \beta(x_r + x_p)]/2$
  - if $|z_p| < d^*,\ d^* \leftarrow |z_p|$ and $(\alpha^*, \beta^*) \leftarrow (\alpha, \beta)$

This algorithm checks all 'possible local minima'. In contrast, Algorithm 1 checks a sequence of lines, each of which is optimal amongst all lines with the same slope. In the next short section we will improve Algorithm 1, both deterministically and on the average.

3. Deterministic and random improvements

One can improve the practical performance of Algorithm 1 by applying what we call the **wedge trick**. Algorithm 1 loops over all pairs r and s but, as we shall see, the cost of many of these loops can be reduced from $n \log(n)$ to $\log(n)$. On the basis of our experience, this reduction is possible often enough to make a big difference in practice.

When examining r, s in the inner loop of Algorithm 1, we find the optimal intercept for lines of slope $\beta_{r,s}$. This defines α^* satisfying

$$(4) \qquad f(\alpha^*, \beta) \le f(\alpha, \beta), \quad \text{all } \alpha.$$

Equation (3) implies that $|M| \ge 2$ and that there are $p, q \in M$ such that $r_p r_q < 0$. If $l_{\alpha^*,\beta}$ doesn't satisfy (ii) of the Main Lemma, it may be rotated clockwise about $\mu = ((x_p, y_p) + (x_q, y_q))/2$ until a point, say s, first satisfies $|r_p| = |r_s|$. Let the slope

of this line be β_L. Now rotate $l_{\alpha^*,\beta}$ counter-clockwise about μ until a point, say t, first satisfies $|r_p| = |r_t|$ and denote the slope of this line by β_U. This defnes a *wedge* of lines through μ with slopes in the interval $W = [\beta_L, \beta_U]$ for which both p and q remain in M. Therefore for *any* $\beta \in W$, the optimal line with slope β is in the wedge. This observation allows one to eliminate from further consideration those r, s for which $\beta_{r,s} \in W$.

The modification to Algorithm 1 (call it **Algorithm W**) is to compute $W = [\beta_L, \beta_U]$, take (α, β) to be the parameters of the better of the two lines defining the wedge (e.g., if $|r_s| < |r_t|$, $\beta \leftarrow \beta_L$), and then eliminate all r, s for which $\beta_{r,s} \in W$. The wedge may be computed in linear time simply by solving $n-2$ simple equations that require $|r_i|$ to be equal to $|r_p|$, i different from p or q. To facilitate the elimination of some $\beta_{r,s}$, a preprocessing step could compute all the $\beta_{r,s}$ and sort then in time $O(n^2 \log(n))$. Then, given W, those $\beta_{r,s} \in W$ may be obtained (say by binary search) in time $O(\log(n))$ and eliminated from further consideration. In the sorted list of $\beta_{r,s}$, we could maintain flags for those slopes which have been eliminated. Given r and s, a search requiring $\log(n)$ time reveals whether $\beta_{r,s}$ is flagged. If not, we must process this slope. The optimal intercept for $\beta_{r,s}$ and the wedge generated by $l^*_\alpha, \beta_{r,s}$ may be obtained in time $O(n \log(n))$. Therefore the total cost of Algorithm W is

$$K\, O(n \log(n)) + (n^2 - K)\log(n) + O(n^2 \log(n)),$$

where K denotes the total number of r, s pairs that needed actual processing.

Monte-Carlo studies of Algorithm W were performed for various models governing random distributions of points and for various values of n ranging from 10 to 125, each combination replicated several times. A rough exploratory analysis of these experiments seems to suggest that $O(n \log(n))$ is a typical value for K. Although there is not thoroughly defensible model for a 'random' fitting problem, we make the following concrete statement.

Conjecture 1. *If* (X_i, Y_i), $1 \le i \le n$, *are independent observations from any bivariate distribution with continuous density, then Algorithm* W *has expected time complexity* $O((n \log(n))^2)$.

The probability theory involved in this conjecture appears to be nontrivial. On the practical side, two points are in order:

- Algorithm W has space complexity of $O(n^2)$. This means that examples of size 1000 become unrealistic without use of external storage.
- Our experience with data sets of size less than 120 shows that (α^*, β^*) can be computed rapidly. John Tukey suggested that a useful approximation to the least median of squares fit for 1000 points could be defined by averaging fits obtained on random subsets of size ~ 100.

The final algorithm, **Algorithm R**, randomizes the wedge trick. In preprocessing,

slopes $\beta_{r,s}$ are computed and sorted. Then, instead of checking the $\beta_{r,s}$ in order, one is selected at random. If it has not been eliminated already, the wedge $W=[\beta_L, \beta_U]$ generated by $\beta_{r,s}$ is computed. Then all $\beta \in W$ are eliminated and the process repeated until all $\beta_{r,s}$ have been eliminated or checked.

Algorithm R appears to perform well. We have no proveable complexity results to assert, e.g., that is superior to Algorithm 1. However we have put together several heuristic arguments that can be construed as support for the following statement.

Conjecture 2. *For Algorithm* R, *the expected value of K is* $O(n \log(n))$.

We know much less about the general problem. If the (x_i, y_i) are now n points in R^{k+1}, the MM^2R objective function is

$$\text{median}(|\, y_i - (\alpha_0 + \alpha_1 x_{i1} + \cdots + \alpha_k x_{ik})|).$$

Local minimizers must be the normal vectors to hyperplanes that bisect $k+2$ points with median sized residuals, a conjecture Jeff Salowe established by induction on k and using the main Lemma. An analogue of Algorithm 1 has complexity $O(n^{k+1} \log(n))$.

References

[1] D.L. Donoho and P.J. Huber, The notion of breakdown point, in: P.J. Bickel, K. Doksum and J.L. Hodges, eds., A Festschrift for Erich L. Lehman (Wadsworth, Belmont, CA, 1983) 157–184.

[2] F.R. Hampel, A general qualitative definition of robustness, Ann. Math. Statist. 42 (1971) 1887–1896.

[3] J.L. Hodges, Efficiency in normal samples and tolerance of extreme values of location, Proc. Fifth Berkeley Symposium on Mathematical Statistics and Probability (1967) 163–168.

[4] A. Leroy and P.J. Rousseeuw, PROGRES: A program for robust regression, Report 201, Centrum voor Statistiek en Operationeel Onderzoek, Univ. of Brussels, 1984.

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