ELSEVIER

Information Processing Letters 63 (1997) 237-241

Information Processing Letters

An approximation algorithm for least median of squares regression

Clark F. Olson [1]

Jet Propulsion Laboratory, California Institute of Technology, 4800 Oak Grove Drive, MS 107-102, Pasadena, CA 91109, USA

Received 11 February 1997; revised 3 June 1997
Communicated by D. Gries

Abstract

Least median of squares (LMS) regression is a robust method to fit equations to observed data (typically in a linear model). This paper describes an approximation algorithm for LMS regression. The algorithm generates a regression solution with median residual no more than twice the optimal median residual. Random sampling is used to provide a simple $O(n \log^2 n)$ expected time algorithm in the two-dimensional case that is successful with high probability. This algorithm is also extended to arbitrary dimension d with $O(n^{d-1} \log n)$ worst-case complexity for fixed $d > 2$. © 1997 Elsevier Science B.V.

Keywords: Least median of squares regression; Approximation algorithms; Computational geometry; Design of algorithms

1. Introduction

A common problem in many fields is to fit an equation to a set of observed data points. Often a linear model can be used. Let us say that we have n data points $\{p_1, \ldots, p_n\}$ in a d-dimensional space, so $p_i = [x_{i1}, \ldots, x_{id}]^T$. In the classical linear model, the points are fit by a hyperplane as follows:

$$x_{id} = \left(\sum_{j=1}^{d-1} \alpha_i x_{ij}\right) + \alpha_d.$$

A robust method of fitting this hyperplane to the set of points is the least median of squares (LMS) estimator [12], which exhibits the best possible breakdown point with respect to outliers [11]. The LMS estimator minimizes the median of the squared residuals, where the residuals are defined as

$$r_i = x_{id} - \left(\sum_{j=1}^{d-1} \alpha_i x_{ij}\right) - \alpha_d.$$

A disadvantage to the LMS estimator is that it is expensive to compute. Currently known exact solutions to this problem require $O(n^{d+1} \log n)$ time for fixed $d > 2$. [11,8]. For $d = 2$, Edelsbrunner and Souvaine [3] supply the best known exact algorithm, which requires $O(n^2)$ time and $O(n)$ space.

Due to the complexity of these algorithms, approximate methods are used for most problems. The method that is commonly used in practice is given by Leroy and Rousseeuw [6], who describe the PROGRESS system to approximate the LMS estimator. They use a random sampling approach that yields an $O(n \log n)$ time algorithm for problems with fixed dimension. The

[1] This work was performed while the author was with the Cornell University Department of Computer Science, Ithaca, NY.

PII S0020-0190(97)00132-4

hidden constant is exponential in the number of dimensions. The drawback to this method is that even if a "good" sample is taken, where all of the points in the sample have less than the optimal median residual, the error in the fit may be arbitrarily large.

We describe an approximation algorithm for LMS regression in two dimensions that has an approximation factor of 2, that succeeds with high probability, and that has $O(n \log^2 n)$ expected complexity. This algorithm is also extended to higher dimensions with the same approximation factor and $O(n^{d-1} \log n)$ worst-case complexity for fixed $d > 2$.

2. An approximate LMS problem

For now, we restrict our attention to the problem of determining the least-median-of-squares (LMS) regression line for n points in the plane. Let $p_i = (x_i, y_i)$ for $1 \leqslant i \leqslant n$. We define the median to be the mth largest value, where $m = \lfloor n/2 \rfloor + 1$, although we could, in fact, choose any rank or quantile without changing the algorithm significantly.

Consider the set of lines that have residuals with respect to some pair of points that are no greater than some arbitrary ε. For points p_i and p_j, we denote this set by $L_\varepsilon(p_i, p_j)$.

$$L_\varepsilon(p_i, p_j) = \{(\alpha, \beta) \mid y_i - \alpha x_i - \beta \leqslant \varepsilon \text{ and } y_j - \alpha x_j - \beta \leqslant \varepsilon\}.$$

The smallest ε for which there exists a line l_{opt} such that $l_{opt} \in L_\varepsilon(p_i, p_j)$ for $\binom{m}{2}$ distinct pairs of points in the set is the optimal median residual. In addition, l_{opt} is the optimal LMS regression line, since it yields the line with minimum residual to m of the points.

The LMS regression problem can be decomposed into n subproblems, each of which considers a distinct *distinguished point* p_k. Each pair of points that includes the distinguished point, (p_k, p_i), $k \neq i$, is called a *distinguished pair*. The subproblems now determine the smallest ε for which there exists a line $l_{opt}^{p_k}$ such that $l_{opt}^{p_k} \in L_\varepsilon(p_k, p_i)$ for $m - 1$ distinct distinguished pairs, and also determine the line, $l_{opt}^{p_k}$, that solves the problem. The m points that are the closest to the optimal LMS line yield subproblems that produce the optimal solution. Random sampling of the distinguished points can now be used to reduce the number of subproblems that must be examined, while still achieving a high probability that the correct solution is found. Call a subproblem examining a particular distinguished point a *constrained LMS problem*. Note that the complexity of solving such subproblems is not necessarily lower than the original problem.

We consider simpler subproblems that allow approximate solutions. For a particular distinguished point, p_k, we determine the smallest ε (denoted by $\varepsilon_{rel}^{p_k}$) for which there exist $m - 1$ (not necessarily distinct) parallel lines $\{l_1, \ldots, l_{m-1}\}$ such that $l_i \in L_\varepsilon(p_{\pi(k)}, p_{\pi(i)})$, $1 \leqslant i \leqslant m - 1$, for some permutation of the points π where $\pi(k) = m$, and also determine the line $l_{app}^{p_k}$ that passes through p_k and is parallel to l_1. The median residual with respect to $l_{app}^{p_k}$ is denoted by $\varepsilon_{app}^{p_k}$.

Such subproblems relax the constraints on the LMS problem by allowing a series of parallel lines, all of which lie within ε of the distinguished point (rather than a single line), that optimize the median residual. We call these *constrained approximate LMS problems*.

3. Is the solution accurate?

We now consider the quality of the solutions to the constrained approximate LMS problems. For simplicity, we assume that there are no ties in the solutions to the problems and that no two points have the same x-coordinate. However, these restrictions can be easily removed.

Let l_{opt} to be the optimal LMS regression solution, yielding median residual ε_{opt}. We call any point that has a residual, with respect to the optimal LMS solution, that is no greater than the mth largest such residual, a *good point*, where m is the rank we are minimizing. The m closest points to the LMS line are thus the good points. All other points are bad points. A *good sample* is one that contains no bad points.

Proposition 1. *If the distinguished point, p_k, is a good point, then $\varepsilon_{app}^{p_k} \leqslant 2\varepsilon_{opt}$.*

Proof. The proposition follows from the following two lemmas. □

Lemma 2. *If the distinguished point, p_k, is a good point, then $\varepsilon_{rel}^{p_k} \leqslant \varepsilon_{opt}$.*

Proof. Consider the optimal LMS regression line, l_{opt}. At least m points, including p_k, have a residual no greater than ε_{opt} with respect to l_{opt}, by definition of the median residual. Permute the points by π such that the distinguished point is p_m and the remaining points are $p_1, \ldots, p_{m-1}$. We have $l_{opt} \in L_{\varepsilon_{opt}}(p_i, p_m)$ for $1 \leqslant i \leqslant m-1$ and this places an upper bound of ε_{opt} on ε_{rel}. □

Lemma 3. *For any distinguished point,* p_k, $\varepsilon_{app}^{p_k} \leqslant 2\varepsilon_{rel}^{p_k}$.

Proof. Consider the line, $l_{app}^{p_k}$, that is the solution to the constrained approximate LMS problem. From the problem definition, we must have m points (including the distinguished point), each of which has a residual no more than $\varepsilon_{rel}^{p_k}$ with respect to some line that is parallel to $l_{app}^{p_k}$. The residual of each point from $l_{app}^{p_k}$ is thus no greater than $2\varepsilon_{rel}^{p_k}$, since $l_{app}^{p_k}$ passes through the distinguished point. □

The solution to a constrained approximate LMS problem is thus guaranteed to yield a median residual no greater than $2\varepsilon_{opt}$ for any problem where the distinguished point is a good point.

The bounds in Lemmas 2 and 3 are tight, but the bound in Lemma 2 can only be achieved when the distinguished point has distance ε_{opt} from the optimal LMS line. If the distance of the distinguished point from the optimal LMS line is ε_{dis}, then $\varepsilon_{rel} \leqslant (\varepsilon_{dis} + \varepsilon_{opt})/2$ and a regression line is found with median residual no greater than $\varepsilon_{dis} + \varepsilon_{opt}$. We are thus likely to get a better result when the distinguished point is close to the optimal LMS line, and, if the distinguished point lies on the true LMS regression line, then we have $\varepsilon_{app} = \varepsilon_{opt}$. (The optimal LMS solution is found whenever the distinguished point lies on l_{opt}.)

4. Solving constrained approximate LMS problems

Consider a particular pair of points, $p_i = (x_i, y_i)$ and $p_j = (x_j, y_j)$, and a particular residual ε. If $x_i \neq x_j$, then the lines belonging to $L_\varepsilon(p_i, p_j)$ have slopes in the following range:

$$\alpha \in \left[\frac{y_j - y_i - 2\varepsilon}{x_j - x_i}, \frac{y_j - y_i + 2\varepsilon}{x_j - x_i}\right]. \quad (1)$$

To solve a constrained approximate LMS problem, we must find the minimum ε such that $m-1$ of the $n-1$ slope ranges yielded by the distinguished point overlap at some point. From (1), each distinguished pair yields a vertical cone (a V shape) in the α–ε plane with its base on the $\varepsilon = 0$ axis. The problem can thus be transformed into finding the lowest point in the $(m-1)$-level of the $n-1$ cones.

Note, first, that any particular ε can be tested in $O(n \log n)$ time to determine if it is correct, below the correct value, or above the correct value. This is performed by computing the $2n-2$ cone boundaries at ε, sorting them, and scanning them in order while maintaining a count on the number of the slope ranges that overlap in each region (i.e. the count is incremented when we reach the beginning of a slope range and decremented when we reach the end of a slope range). If a finite region is found where $m-1$ or more regions overlap, then the correct ε is below the tested value. If no points are found where $m-1$ regions overlap, then the correct ε is above the tested value. If a single point (assuming no ties) is found at which $m-1$ or more regions overlap, then the tested value is the solution and the slope of $l_{app}^{p_k}$ is given by this point of overlap.

This formulation of the problem is amenable to parametric search techniques, since the solution must occur at the intersection of two cone boundaries. For example, we can use Megiddo's techniques [9] to solve this problem in $O(n \log^3 n)$ time, since the testing step requires $O(n \log n)$ time. Cole [1] describes techniques by which this can be improved to $O(n \log^2 n)$ using either a randomized algorithm or a complex deterministic algorithm. We describe a variation on the inversion sampling and contraction methods given by both Matoušek [7] and Dillencourt et al. [2] that yields a simple $O(n \log^2 n)$ randomized method for solving this problem.

The approach that we follow is to randomly sample n of the $(n-1)(n-2)/2$ cone intersections (we can neglect the intersections that occur between two right cone boundaries or two left cone boundaries). These intersections are sorted according to ε and the two that bracket the correct value are determined in $O(n \log^2 n)$ time using $O(\log n)$ steps of binary search. Each step tests the median ε in the remaining interval and determines whether the correct value is above, below, or at the tested value, as described above. The interval is then contracted appropriately.

Now, the remaining intersections that lie between the determined brackets must be found. The expected number of such intersections, I, is $O(n)$ (this is a variant of well known results, see [10]). These intersections can be enumerated in $O(n \log n + I)$ time using inversion counting techniques [5,7,2]. A final binary search is then performed on these remaining intersections to find the solution to the constrained approximate LMS problem. The expected time required for a single trial is thus $O(n \log^2 n)$.

Let us consider how many constrained approximate LMS problems must be examined such that at least one is a good sample with high probability. The probability of examining at least one good sample in t trials, if sampling with replacement is used, is

$$P_{good} = 1 - \left(\frac{n-m}{n}\right)^t \geqslant 1 - \left(\frac{1}{2}\right)^t.$$

To achieve $P_{good} \geqslant 1-\delta$, we may use $t = \lceil -\log_2 \delta \rceil$ trials. The number of trials is thus independent of the number of data points, and the overall approximate LMS algorithm requires $O(n \log^2 n)$ expected time for fixed δ (with a constant that has a logarithmic dependence on δ).

5. Higher dimensions

In higher dimensions, the LMS problem is that of finding the $(d-1)$-dimensional hyperplane that best fits n points in d-dimensions:

$$x_d = \sum_{j=1}^{d-1} \alpha_j x_j + \alpha_d.$$

To extend our algorithm for solving constrained approximate LMS problems to this case, $d-1$ distinguished points are sampled, rather than a single one, and a d-dimensional cone is constructed in the space spanned by $(\varepsilon, \alpha_1, \ldots, \alpha_{d-1})$ for each set of d points that includes the $d-1$ distinguished points. These cones are searched in a manner similar to the two-dimensional case, although the two step search procedure is no longer necessary. This section sketches this method.

We must first specify how to construct the cone of slopes that are consistent with some hyperplane fitting each of d points up to an error of ε. Let the coordinates of the points be given by $p_i = (x_{i1}, \ldots, x_{id})$ for $1 \leqslant i \leqslant d$. The hyperplane that fits the set of points exactly (assuming non-degeneracy) is given by:

$$\begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_d \end{bmatrix} = \begin{bmatrix} x_{11} & \cdots & x_{1(d-1)} & 1 \\ \vdots & & \vdots & \vdots \\ x_{d1} & \cdots & x_{d(d-1)} & 1 \end{bmatrix}^{-1} \begin{bmatrix} y_1 \\ \vdots \\ y_d \end{bmatrix}.$$

Allowing an error of ε yields a volume in the α-space that is bounded by the 2^d hyperplanes given by all possible combinations of adding ε or $-\varepsilon$ to each of the y_i's in the above equation:

$$\begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_d \end{bmatrix} = \begin{bmatrix} x_{11} & \cdots & x_{1(d-1)} & 1 \\ \vdots & & \vdots & \vdots \\ x_{d1} & \cdots & x_{d(d-1)} & 1 \end{bmatrix}^{-1} \begin{bmatrix} y_1 \pm \varepsilon \\ \vdots \\ y_d \pm \varepsilon \end{bmatrix}.$$

If the points are in general position (i.e. the matrix is invertible), this yields $2^d - 2$ hyperplanes in the $(\varepsilon, \alpha_1, \ldots, \alpha_{d-1})$ space. Note that we drop α_d, since we are examining only the slopes and not the intercept parameter. This also allows us to discard the two hyperplanes where ε either added to each point or subtracted from each point, since these only change the intercept of the hyperplane from the exact fit and not the slopes.

The cone for the set of points can now be constructed from the intersection of the half-spaces above (with respect to ε) each of these hyperplanes. The cones have their bases at $\varepsilon = 0$, axes perpendicular to the $\varepsilon = 0$ hyperplane, and a cross-section that is a convex polyhedron of linearly increasing size in ε.

In order to solve a constrained approximate LMS problem in a d-dimensional space, the minimum ε at which there exists a point in the $(m-1)$-level of the $n-1$ cones is sought. This solution must occur at the minimum ε at which some pair of the cones intersect. Note that determining the minimum ε at which an arbitrary pair of cones intersect is a linear programming problem with d dimensions and $2^{d+1}-4$ constraints. It is thus possible to enumerate the $O(n^2)$ possible solutions (corresponding to each pair of cones) in $O(n^2)$ time for any fixed d.

To determine which of these possible solutions is correct, it suffices to sort them and perform binary search using a decision procedure that is able to determine which side of any particular ε the correct solution lies on. In order to test a particular ε, we search

the arrangement of the $(d-1)$-dimensional convex polyhedra given by slicing each of the cones at ε and determine if any point lies within $m-1$ of the polyhedra. The sorting step requires $O(n^2 \log n)$ time and the decision procedure for each of the $O(\log n)$ binary search steps requires $O(n^{d-1})$ time for any fixed $d > 2$ [4]. The algorithm thus requires $O(n^{d-1} \log n)$ time for fixed $d > 2$ and is guaranteed to find a solution with residual no greater than twice the optimal residual, if each of the $d-1$ distinguished points is a good point.

The probability that any particular trial examines a set of $d-1$ good points (sampling without replacement) is

$$P_1 = \prod_{i=0}^{d-2} \frac{m-i}{n} > \left(\frac{m-d+2}{n}\right)^{d-1}.$$

Taking t trials yields the following probability of success (sampling with replacement):

$$P_t = 1 - (1 - P_1)^t.$$

We can achieve $P_t \geqslant 1 - \delta$ by using

$$t = \frac{\log \delta}{\log(1 - (1/4)^{d-1})},$$

if $n \geqslant 4d - 8$.

6. Summary

This work has considered an approximation algorithm to perform LMS regression. We have formalized an approximate LMS problem and shown that solutions to this problem have bounded error in terms of the median residual that they yield. A two-dimensional constrained approximate LMS problem can be solved in $O(n \log^2 n)$ expected time using a variety of randomized algorithms or a complex deterministic algorithm. Random sampling is used to determine a good solution to unconstrained approximate LMS problems with high probability by solving a constant number of constrained approximate LMS problems. The algorithm is simple to implement and yields a method that is fast and accurate in practice. The techniques have been extended to higher dimensions with an $O(n^{d-1} \log n)$ worst-case complexity for fixed $d > 2$, although the hidden constant is exponential in d.

References

[1] R. Cole, Slowing down sorting networks to obtain faster sorting algorithms, *J. ACM* 34 (1) (1987) 200–208.

[2] M.B. Dillencourt, D.M. Mount and N.S. Netanyahu, A randomized algorithm for slope selection, *Internat. J. Comput. Geom. Appl.* 2 (1) (1992) 1–27.

[3] H. Edelsbrunner and D.L. Souvaine, Computing least median of squares regression lines and guided topological sweep, *J. Amer. Statist. Assoc.* 85 (409) (1990) 115–119.

[4] H. Edelsbrunner, J. O'Rourke and R. Seidel, Constructing arrangements of lines and hyperplanes with applications, *SIAM J. Comput.* 15 (1986) 341–363.

[5] D.E. Knuth, *The Art of Computer Programming, Vol. 3: Sorting and Searching* (Addison-Wesley, Reading, MA, 1973).

[6] A. Leroy and P.J. Rousseeuw, PROGRESS: A program for robust regression, Research Rept. 201, Centrum voor Statistiek en Operationeel Onderzoek, University of Brussels, 1984.

[7] J. Matoušek, Randomized optimal algorithm for slope selection, *Inform. Process. Lett.* 39 (1991) 183–187.

[8] P. Meer, D. Mintz, A. Rosenfeld and D.Y. Kim, Robust regression methods for computer vision: A review, *Internat. J. Comput. Vision* 6 (1) (1991) 59–70.

[9] N. Megiddo, Applying parallel computation algorithms in the design of serial algorithms, *J. ACM* 30 (4) (1983) 852–865.

[10] K. Mulmuley, *Computational Geometry: An Introduction Through Randomized Algorithms* (Prentice-Hall, Englewood Cliffs, NJ, 1994).

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

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