

UNCLASSIFIED

Defense Technical Information Center
Compilation Part Notice

ADP013759

TITLE: A Robust Algorithm for Least Absolute Deviations Curve Fitting

DISTRIBUTION: Approved for public release, distribution unlimited

This paper is part of the following report:

TITLE: Algorithms For Approximation IV. Proceedings of the 2001
International Symposium

To order the complete compilation report, use: ADA412833

The component part is provided here to allow users access to individually authored sections of proceedings, annals, symposia, etc. However, the component should be considered within the context of the overall compilation report and not as a stand-alone technical report.

The following component part numbers comprise the compilation report:

ADP013708 thru ADP013761

UNCLASSIFIED

A robust algorithm for least absolute deviations curve fitting

Dongdong Lei, Iain J Anderson

University of Huddersfield, Huddersfield, UK.
d.lei@hud.ac.uk

Maurice G Cox

National Physical Laboratory, Teddington, UK.
Maurice.Cox@npl.co.uk

Abstract

The least absolute deviations criterion, or the ℓ_1 norm, is frequently used for approximation where the data may contain outliers or 'wild points'. One of the most popular methods for solving the least absolute deviations data fitting problem is the Barrodale and Roberts (BR) algorithm (1973), which is based on linear programming techniques and the use of a modified simplex method [1]. This algorithm is particularly efficient. However, since it is based upon the simplex method it can be susceptible to the accumulation of unrecoverable rounding errors caused by using an inappropriate pivot. In this paper we shall show how we can extend a numerically stable form of the simplex method to the special case of ℓ_1 approximation whilst still maintaining the efficiency of the Barrodale and Roberts algorithm. This extension is achieved by using the ℓ_1 characterization to rebuild the relevant parts of the simplex tableau at each iteration. The advantage of this approach is demonstrated most effectively when the observation matrix of the approximation problem is sparse, as in the case when using compactly supported basis functions such as B-splines. Under these circumstances the new method is considerably more efficient than the Barrodale and Roberts algorithm as well as being more robust.

1 Introduction

Given a set of m data points $\{(x_i, y_i)\}_{i=1}^m$, the ℓ_1 , or least absolute deviations curve-fitting problem seeks $c \in \mathbb{R}^n$ to solve the optimization problem

$$\min_c \|y - Ac\|_1 = \sum_{i=1}^m \left| y_i - \sum_{j=1}^n a_{i,j} c_j \right| = \sum_{i=1}^m |r_i|, \quad (1.1)$$

where A is an $m \times n$ observation matrix, and r_i denotes the residual of the i th point.

Another way of stating the ℓ_1 , or least absolute deviations curve-fitting problem, is by the characterization theory of an ℓ_1 solution [8], which may be given in different forms. The following is perhaps the most commonly used.

A vector $c \in \mathbb{R}^n$ solves the minimization problem (1.1) if and only if there exist $\lambda \in \mathbb{R}^m$ such that

$$A^T \lambda = 0 \quad \text{with} \quad \begin{cases} |\lambda_i| \leq 1, & \text{for } i \in \mathcal{Z}, \\ \lambda_i = \text{sign}(r_i), & \text{for } i \notin \mathcal{Z}, \end{cases} \quad (1.2)$$

where \mathcal{Z} represents the set of indices for which $r_i = 0$.

One of the popular methods designed for solving the ℓ_1 approximation problem is the Barrodale and Roberts (BR) algorithm. It replaces the unconstrained variables c and r in (1.1) by nonnegative variables c^+ , c^- , u and v , and considers the linear programming problem

$$\begin{aligned} \min_c \quad & e^T u + e^T v \\ \text{subject to} \quad & A c^+ - A c^- + u - v = y, \\ & c^+, c^-, u, v \geq 0. \end{aligned} \quad (1.3)$$

Much of the reason for the popularity of the BR algorithm is that it exploits the characteristics of the ℓ_1 approximation in order to solve the problem in a more efficient manner than the general simplex approach. However, it is a simplex based method, and so it is susceptible to numerical instabilities caused by using inappropriate pivots. The new method presented here uses matrix factorization instead of simplex pivoting. This approach allows numerically stable updates to be made, thus avoiding the unnecessary build-up of rounding errors. This method is particularly efficient when the observation matrix is large and sparse [5].

Bartels [2] and Gill and Murray [4] presented methods that concentrate on avoiding the inherent instability of the simplex method. However, these methods are designed for a general linear programming problem and if we were to employ these techniques for the special case of the ℓ_1 problem, the storage requirements and computational workload of the method would be unnecessarily large compared to those of the highly efficient BR algorithm.

The ℓ_1 problem is, in essence, an interpolation problem. The aim of any iterative procedure for the ℓ_1 problem is to find an optimal set of interpolation points. Indeed, this is how the BR algorithm solves the ℓ_1 problem. It begins with all coefficients, c , set to zero (being non-basic variables), and during each iteration of stage one, one of the residuals, r_i , becomes non-basic by making the corresponding point an interpolation point (i.e., the coefficients are altered so that $r_i = 0$). At the end of stage one, the current estimate interpolates n distinct points. During stage two, the interpolation points are exchanged one at a time with a non-interpolation point until an optimal solution is achieved.

In fact, the new algorithm is effectively identical to the BR algorithm in the sense that we use exactly the same pivoting strategy. However, we start with a predetermined set of interpolation points and do not store the simplex tableau directly. In each iteration, we only reconstruct the parts of the simplex tableau that are needed by the more stable approach employed.

2 A more stable computational approach

The linear programming presentation of a least absolute deviations curve-fitting problem is given in (1.3). It is a standard linear programming problem of dimension $m \times (2m+2n)$. The robust approaches of Bartels and Gill and Murray can be applied to solve it. They involve the factorization of an $m \times m$ matrix. On the other hand, the BR algorithm only deals with an $m \times n$ matrix in each iteration, if $m \gg n$, the direct usage of these stable approaches is less efficient. We shall show next that the factorization of an $n \times n$ matrix is all that is required at each iteration.

We split the data points based on the set interpolation \mathcal{Z} , and let $A_{\mathcal{Z}}, y_{\mathcal{Z}}, u_{\mathcal{Z}}$ and $v_{\mathcal{Z}}$ be the counterparts of A, y, u and v in (1.3) corresponding to the set \mathcal{Z} . Their complementary matrix and vectors are denoted by $\tilde{A}_{\mathcal{Z}}$ and $\tilde{y}_{\mathcal{Z}}, \tilde{u}_{\mathcal{Z}}$ and $\tilde{v}_{\mathcal{Z}}$, so that $A_{\mathcal{Z}}$ and $\tilde{A}_{\mathcal{Z}}$ comprise A , etc., problem (1.3) can be expressed as

$$\begin{aligned} \min_c \quad & e^T(u_{\mathcal{Z}} + \tilde{u}_{\mathcal{Z}}) + e^T(v_{\mathcal{Z}} + \tilde{v}_{\mathcal{Z}}) \\ \text{subject to} \quad & A_{\mathcal{Z}}c^+ - A_{\mathcal{Z}}c^- + u_{\mathcal{Z}} - v_{\mathcal{Z}} = y_{\mathcal{Z}}, \\ & \tilde{A}_{\mathcal{Z}}c^+ - \tilde{A}_{\mathcal{Z}}c^- + \tilde{u}_{\mathcal{Z}} - \tilde{v}_{\mathcal{Z}} = \tilde{y}_{\mathcal{Z}}, \\ & c^+, c^-, u_{\mathcal{Z}}, \tilde{u}_{\mathcal{Z}}, v_{\mathcal{Z}}, \tilde{v}_{\mathcal{Z}} \geq 0. \end{aligned} \quad (2.1)$$

Since the coefficients for c_j^- are just the negative of the coefficients for c_j^+ , $j = 1, 2, \dots, n$, it is possible to suppress c_j^- and let c represent the unconstrained variable. The initial simplex tableau associated with problem (2.1) can be constructed in matrix form by Table 1, where e_k , $k = m, n, m-n$, are $k \times 1$ vectors with all components equal to one.

BV	c	$u_{\mathcal{Z}}$	$\tilde{u}_{\mathcal{Z}}$	$v_{\mathcal{Z}}$	$\tilde{v}_{\mathcal{Z}}$	r
$u_{\mathcal{Z}}$	$A_{\mathcal{Z}}$	I	0	$-I$	0	$y_{\mathcal{Z}}$
$\tilde{u}_{\mathcal{Z}}$	$\tilde{A}_{\mathcal{Z}}$	0	I	0	$-I$	$\tilde{y}_{\mathcal{Z}}$
Z	$e_m^T \begin{pmatrix} A_{\mathcal{Z}} \\ \tilde{A}_{\mathcal{Z}} \end{pmatrix}$	0	0	$-2e_n^T$	$-2e_{m-n}^T$	$e_m^T \begin{pmatrix} y_{\mathcal{Z}} \\ \tilde{y}_{\mathcal{Z}} \end{pmatrix}$

TAB. 1. The initial simplex tableau of the ℓ_1 fitting problem.

As we know, the simplex method is an iterative procedure in which each iteration is characterized by specifying which m of $2m+n$ variables are basic. For the ℓ_1 approximation, we are only concerned with those vertices which are formed by a set of interpolation points. For n interpolation points, the basic variables consist of n of the coefficient parameters c and $m-n$ of the parameters $\tilde{u}_{\mathcal{Z}}$ corresponding to the non-interpolation points.

Let B be the $m \times m$ basis matrix whose columns consist of the m columns associated with the basic variables. Then

BV	u_z	r
c	A_Z^{-1}	$A_Z^{-1}y_Z$
\tilde{u}_Z	$-\tilde{A}_Z A_Z^{-1}$	\tilde{r}_Z
Z	$-e_{m-n}^T(\tilde{A}_Z A_Z^{-1}) - e_n^T$	$e_{m-n}^T \tilde{r}_Z$

TAB. 2. The condensed simplex tableau associated with a set of interpolation points.

$$B = \left(\begin{array}{c|c} A_Z & 0 \\ \hline \tilde{A}_Z & I \end{array} \right). \tag{2.2}$$

It is readily verified that the inverse of B can be written in the form of (2.3) as long as A_Z is invertible.

$$B^{-1} = \left(\begin{array}{c|c} A_Z^{-1} & 0 \\ \hline -\tilde{A}_Z A_Z^{-1} & I \end{array} \right). \tag{2.3}$$

Equation(2.3) shows that the explicit inverse computation of an $m \times m$ matrix in the form of (2.2) can be achieved by dealing with an inverse of an $n \times n$ matrix, and in general, $n \ll m$.

To make the m non-basic variables become basic, we multiply the whole simplex tableau by B^{-1} , and omit the identity and zero matrices. Then new simplex tableau is given in Table 2.

An arbitrary choice of the interpolation set Z may cause some of the values in the right hand side column to become negative. Although it is permissible for the coefficient parameters c to be negative, for those rows having negative residuals \tilde{r}_Z , we restore feasibility by exchanging the corresponding \tilde{u}_Z for \tilde{v}_Z . This exchanging can be made by subtracting twice those rows from the objective row and changing the sign of the original rows [1].

Such an exchange process can be expressed in matrix terms by introducing a sign vector

$$\tilde{\lambda}_Z = \text{sign}(\tilde{r}_Z).$$

Let \tilde{A}_{Z_s} represent the matrix which is obtained by multiplying those rows of \tilde{A}_Z associated with negative residuals by -1 ,

$$\tilde{A}_{Z_s} = \text{diag}(\tilde{\lambda}_Z)\tilde{A}_Z.$$

BV	u_z	r
c	A_Z^{-1}	$A_Z^{-1}y_Z$
\tilde{u}_Z	$-\tilde{A}_{Z_s}A_Z^{-1}$	$ \tilde{r}_Z $
Z	$-\tilde{\lambda}_Z^T(\tilde{A}_Z A_Z^{-1}) - e_n^T$	$\tilde{\lambda}_Z^T \tilde{r}_Z$

TAB. 3. Restoration of feasibility of the simplex tableau.

The simplex tableau after restoring feasibility is shown in Table 3.

The point to be removed from Z is decided by the values of the objective row. Each time the maximum value of the objective row (including the suppressed columns) is chosen, we let the index of this element be k . In order to choose which new point is to join the set Z , we compute the value of the pivotal column, the k th column in the simplex tableau. Since the simplex tableau is in the form of

$$\begin{bmatrix} I \\ -\tilde{A}_{Z_s} \end{bmatrix} A_Z^{-1},$$

the k th column can be obtained by using \tilde{A}_{Z_s} and the k th column of A_Z^{-1} .

The BR algorithm pivoting strategy is adopted to decide which new point is to be added to the interpolation set, when a new set of indices Z is generated. We repeat the process in an iterative manner until the optimal solution is achieved.

Table 3 is in fact identical to the simplex tableau of the BR algorithm in stage 2. The difference here is that the BR algorithm is implemented by a simplex pivoting approach, while the transformation of the simplex tableau in the form of Table 3 can be accomplished in a numerically more stable manner.

3 The improved method

The improved method starts with a predetermined interpolation set Z , the minimum requirement for Z being that it forms a well-behaved matrix A_Z . For B-spline basis functions, we can choose any set of points satisfying the Schoenberg-Whitney condition [6]. For a Chebyshev polynomial basis, points close to the n Chebyshev zeros can be regarded as the initial interpolation set. In other cases, we can choose points approximate to them or even uniformly distributed.

If we denote the set of λ_i , $i \in Z$, as λ_Z , we can rewrite the characterization equation (1.2) as

$$A_Z^T \lambda_Z = -\tilde{A}_Z^T \tilde{\lambda}_Z, \quad (3.1)$$

and λ_Z can be obtained mathematically from

$$\lambda_Z = -(A_Z^T)^{-1}(\tilde{A}_Z^T \tilde{\lambda}_Z). \quad (3.2)$$

Table 3 shows that the objective row can be computed as

$$\text{Objective row} = -(\tilde{\lambda}_Z^T \tilde{A}_Z) A_Z^{-1} - e_n^T. \quad (3.3)$$

Thus, using (3.2) we conclude that

$$\text{Objective row} = \lambda_Z^T - e_n^T. \quad (3.4)$$

We know that at the ℓ_1 solution all the values in the objective row are in the range $[-2, 0]$, and also $|\lambda| \leq 1$. This latter result can be explained in terms of the former by the relationship (3.4).

(3.4) is useful because it can be used to verify whether an interpolation set forms an optimal solution, or to compute λ from the values of the objective row. We use it to compute the values of the objective row.

The improved method can be summarized as follows;

- (1) Choose an initial set of interpolation points and form the set Z .
- (2) Construct A_Z, y_Z and their counterpart \tilde{A}_Z, \tilde{y}_Z accordingly.
- (3) Solve the equation $A_Z c = y_Z$ for c , and compute

$$\tilde{r}_Z = \tilde{y}_Z - \tilde{A}_Z c, \quad \text{and} \quad \tilde{\lambda}_Z = \text{sign}(\tilde{r}_Z).$$

- (4) Obtain the values of λ_Z from the equation

$$A_Z^T \lambda_Z = -\tilde{A}_Z^T \tilde{\lambda}_Z. \quad (3.5)$$

- (5) If $|\lambda_Z| \leq 1$ hold, the current solution is optimal, and the algorithm terminates. Otherwise, continue.
- (6) Obtain the objective row of the BR simplex tableau from

$$\text{objective row} = \lambda_Z^T - e_n^T.$$
- (7) Examine the values of the objective row; the point associated with the maximum value of the objective row is chosen to leave the set Z .
- (8) Decide the point to add by the BR pivoting strategy. Obtain a new set of indices Z , and repeat from step 2.

4 Practical considerations and application to the ℓ_1 spline approximation

The robustness of the above algorithm stems from the reliable updating of the relevant parts of the simplex tableau in each iteration. The major computational work is obtaining (explicitly or implicitly) the inverse of an $n \times n$ matrix A_Z . It can be calculated and stored explicitly by using an LU or QR factorization, or preferably it can be expressed as a product of factors. Since A_Z differs from its predecessor by only one row, savings can be made by reusing results from the previous step. Necessary material is available [4, 7] regarding the stable implementation of this row updating procedure.

$m = 512$	Numbers of iterations		Execution Time (seconds)	
	New	BR	New	BR
$q =$				
44	57	125	1.6	14.7
49	75	111	2.2	13.4
54	71	134	2.4	20.2
59	83	156	3.0	26.8
64	78	160	3.1	32.4
69	88	194	4.0	42.4
74	75	165	3.7	36.0
79	87	189	4.8	48.1

TAB. 4. The number of iterations and execution time taken by the algorithm of this paper and the Barrodale and Roberts algorithm for a set of 512 response data points provided by the National Physical Laboratory.

Sparsity almost always is more important than matrix dimension. Additional savings can be made if the observation matrix A is sparse or structured. Approximation using a B-spline basis often occurs in practical applications. In such cases, A is block banded, and A_Z can be triangularized using $\mathcal{O}(n)$ flops [3]. Similarly, the sparsity of A can be exploited to compute other relevant parts of the simplex tableau efficiently.

We have applied our method to solve the least absolute deviations curve-fitting problems by B-splines using various numbers of interior knots. All software was written in MATLAB and implemented on a Sun Workstation. The initial interpolation points are chosen to be those points corresponding to the maximum value in each column of the observation matrix A .

Some of our computational results are reported in Tables 4 and 5. Each table presents the outcomes of a particular set of data points by the new method and by the BR algorithm.

All the experimental results exhibit the effectiveness of the improved method on large, sparse systems. Although these tables show that the improved method is faster than the BR algorithm, it would be unfair to judge the convergence speed purely based upon the time taken, since the improved method embodies some MATLAB built-in functions, while the BR algorithm uses only user-defined functions. However, on average, the new method requires far fewer iterations than the BR algorithm, and is competitive with the BR algorithm both in efficiency and accuracy for a structured system.

Further work to be addressed by the authors will involve a definitive implementation of this algorithm in Fortran, and development of an error analysis for both the improved method and the BR algorithm.

$m = 1200$	Numbers of iterations		Execution Time (seconds)		
	$q =$	New	BR	New	BR
	50	82	143	4.0	58.7
	56	105	165	5.2	85.8
	62	113	190	6.1	110.2
	68	131	189	7.6	110.4
	74	121	223	7.8	157.9
	80	132	216	9.2	163.2
	86	155	245	11.8	209.8
	92	173	252	14.0	241.8
	98	153	272	13.6	292.6

TAB. 5. The number of iterations and execution time taken by the algorithm of this paper and the Barrodale and Roberts algorithm for a set of 1200 data points, generated by MATLAB command $x = \text{linspace}(1, 10, 1200)'$; $y = \log(x) + \text{randn}(1200, 1)$.

Bibliography

1. I. Barrodale and F. D. K. Roberts. An improved algorithm for discrete ℓ_1 linear approximation. *SIAM Journal of Numerical Analysis* **10**, 839–848, 1973.
2. R. H. Bartels. A stabilization of the simplex method. *Numerical Math.* **16**, 414–434, 1971.
3. M. G. Cox. The least squares solution of overdetermined linear equations having band or augmented band structure. *IMA Journal of Numerical Analysis* **1**, 3–22, 1981.
4. P. E. Gill and W. Murray. A numerically stable form of the simplex algorithm. *Linear Algebra and its Applications* **7**, 99–138, 1973.
5. D. Lei, I. J. Anderson, and M. G. Cox. An improved algorithm for approximating data in the ℓ_1 norm. In P. Ciarlini, M. G. Cox, E. Filipe, F. Pavese, and D. Richter, editors, *Advanced Mathematical and Computational Tools in Metrology V*, 247–250, Singapore, 2001. World Scientific Publishing.
6. M. J. D. Powell. *Approximation Theory and Methods*. Cambridge University Press, Cambridge, UK, 1981.
7. R. J. Vanderbei. *Linear Programming — Foundations and Extensions*. Kluwer Academic Publishers, Boston, MA, US, 1997.
8. G. A. Watson. *Approximation Theory and Numerical Methods*. Wiley, New York, US, 1980.