AD-A091 216 TEXAS UNIV AT AUSTIN CENTER FOR CYBERNETIC STUDIES F/G 12/1
AN ALGORITHM FOR A LEAST ABSOLUTE VALUE REGRESSION PROBLEM WITH--ETC(U)
JUN 80 R ARMSTRONG, M KUNG N00014-75-C-0569
UNCLASSIFIED CCS-378 NL

END
DATE FILMED
12-80
DTIC

AD A091216

LEVEL II

12

CENTER FOR CYBERNETIC STUDIES

The University of Texas
Austin, Texas 78712

Research Report 378

AN ALGORITHM FOR A LEAST ABSOLUTE VALUE REGRESSION PROBLEM WITH BOUNDS ON THE PARAMETERS

by

R. Armstrong
M. Kung*

June 1980

*York University, Downsview, Ontario, Canada

This research was partly supported by ONR Contracts N00014-75-C-0569 and N00014-80-C-0242 with the Center for Cybernetic Studies, The University of Texas at Austin. Reproduction in whole or in part is permitted for any purpose of the United States Government.

CENTER FOR CYBERNETIC STUDIES

A. Charnes, Director
Business-Economics Building, 203E
The University of Texas at Austin
Austin, Texas 78712
(512) 471-1821

An Algorithm for a Least Absolute Value
Regression Problem with Bounds on the Parameters

ABSTRACT

This paper presents a special purpose linear programming algorithm to solve a least absolute value regression problem with upper and lower bounds on the parameters. The algorithm exploits the problem's special structure by maintaining a compact representation of the basis inverse and by allowing for the capability to combine several simplex iterations into one. Computational results with a computer code implementation of the algorithm is given.

1. Introduction

Least absolute value (LAV) regression has become popular recently as a robust estimation technique [11] . This has come about as a result of an increasing awareness of the limitations of least squares analysis and the development of efficient algorithms for obtaining LAV estimates [2,5] .

This paper considers a restricted least absolute value regression problem of the following form.

$$\text{(1)} \qquad \text{Minimize } \sum_{i=1}^{n} \left| y_i - \sum_{j=1}^{m} x_{ij}\beta_j \right| ,$$

$$\text{subject to} \qquad L_j \leq \beta_j \leq U_j , \quad j = 1, 2, \ldots , m.$$

For purposes of exposition, it is assumed that all β_j's have upper and lower bounds. An absence of an upper (lower) bound on a parameter can be handled within the algorithm to be developed here by either assigning an arbitrarily large positive (negative) value for the bound, or by assuming the existence of this pseudo-bound and handling it logically (i.e., no value is actually assigned). The former method is used in our computer code implementation.

Special purpose algorithms for the least squares (LSQ) equivalent of (1) are given by Armstrong and Frome [1] and Waterman [18]. These algorithms were able to deviate from the standard linear programming (LP) approach to solving restricted LSQ problems, commonly called quadratic programming problems, by utilizing the special structure of the constraints. Because the objective function of (1) is convex, a modification of the algorithms found in [1] and [18] can also be used to solve the restricted LAV problem. The major change is that an unrestricted LAV problem must be solved at any stage rather than an unrestricted LSQ problem. However, inasmuch as the most efficient solution method for the unrestricted LAV problem utilizes LP and the constraints of (1) can be handled within the LP framework, it would seem appropriate to specialize the existing LAV-LP algorithms to solve (1). This is done in the next section.

2. Algorithm

Charnes, Cooper and Ferguson [8] appear to be the first to have demonstrated that linear LAV problems can be rewritten as LP problems. Employing their result here, problem (1) is equivalent to:

(2) Minimize $\sum_{i=1}^{n} (\delta_i^+ + \delta_i^-),$

subject to

$$\sum_{j=1}^{m} x_{ij}\beta_j + \delta_i^+ - \delta_i^- = y_i, \quad i = 1, 2, \ldots, n$$

$$L_j \leq \beta_j \leq U_j, \quad j = 1, 2, \ldots, m$$

$$\delta_i^+ \geq 0 \text{ and } \delta_i^- \geq 0, \quad i = 1, 2, \ldots, n.$$

Davies [10], Barrodale and Roberts [5], and Spyropoulos, Kiountouzis and Young [17] give closely related special purpose primal simplex algorithms to solve (2) when bounds on the parameters are not present. These algorithms currently appear to be the most efficient method to solve the unrestricted LAV problem. Armstrong and Hultz [3] and Barrodale and Roberts [6] develop primal LP algorithms for a LAV problem with arbitrary linear constraints on the parameters. Both algorithms can be applied to problem (2). Armstrong and Hultz [3] solve the restricted LAV problem using techniques from interval linear programming [9] while Barrodale and Roberts [6] utilize a modification of their unconstrained algorithm which was presented in [4]. The algorithm proposed here is a specialization of the method from [3]. It will be demonstrated that considerable computational simplications are possible when the constraints are of the important special form present in (2).

To aid in the discussion to follow, the constraints of (2) are rewritten in matrix notation as:

(3) $$X\beta + I\delta^{+} - I\delta^{-} = Y,$$

(4) $$L \leq I\beta \leq U,$$

(5) $$\delta^{+} \geq 0 \text{ and } \delta^{-} \geq 0.$$

The dual problem of (2) is:

(6) $$\text{Maximize } \sum_{i=1}^{n} \pi_i y_i + \sum_{j=1}^{m} w_j U_j + \sum_{j=1}^{m} v_j L_j$$

subject to

$$\sum_{i=1}^{n} \pi_i x_{ij} + w_j + v_j = 0, \; j = 1, 2, \ldots, m$$

$$\pi_i \leq 1, \; i = 1, 2, \ldots, n$$

$$\pi_i \geq -1, \; i = 1, 2, \ldots, n$$

$$w_j \leq 0, \; j = 1, 2, \ldots, m$$

$$v_j \geq 0, \; j = 1, 2, \ldots, m$$

The dual problem (6) will be used later with the explanation of the algorithm.

From the results presented by Armstrong and Hultz [3] , all the information required to execute a primal LP algorithm can efficiently be obtained from a list of indicators and the inverse of an m by m basic matrix. This matrix, X_B, can be represented (after row and column interchanges), relative to problem (2), as follows.

$$X_B = \begin{pmatrix} X_F & X_R \\ 0 & I \end{pmatrix}$$

where X_F is an r by r full rank matrix, X_R is an r by m-r matrix, 0 is an m-r by r zero matrix and I is an m-r by m-r identity matrix. This structure assumes that r rows of X and m-r rows corresponding to the bound constraints are in the basis. The following discussion will assume that the rows and columns of the problem have been <u>explicitly</u> reordered to obtain this structure. The computer code utilizes two arrays to <u>implicitly</u> achieve this reordering.

It will be demonstrated that all the computations required in the algorithm can conveniently be performed by inverting only X_F, which shall be referred to as the working basis. The inverse of X_B is given by:

$$X_B^{-1} = \begin{pmatrix} X_F^{-1} & -X_F^{-1}X_R \\ 0 & I \end{pmatrix} .$$

The solution of (1), $\overline{\beta}$, can be partitioned into $\overline{\beta} = \begin{pmatrix} \overline{\beta}_F \\ \overline{\beta}_R \end{pmatrix}$ where $\overline{\beta}_F$ is the current value of β_j, $j=1,2,\ldots,r$, and $\overline{\beta}_R$ is the current value of β_j, $j=r+1,r+2,\ldots,m$. From equations (3) and (4), $\overline{\beta} = X_B^{-1} \begin{pmatrix} Y_B \\ C \end{pmatrix}$ where Y_B are the values of the dependent variable corresponding to the r rows of X in the basis and C assumes the value of either U or L depending on whether β_j, $j=r+1,r+2,\ldots,m$ is at its upper or lower bound.

$$\begin{pmatrix} \overline{\beta}_F \\ \overline{\beta}_R \end{pmatrix} = \begin{pmatrix} X_F^{-1} & -X_F^{-1}X_R \\ 0 & I \end{pmatrix} \begin{pmatrix} Y_B \\ C \end{pmatrix}$$

$$\overline{\beta}_R = C$$

$$\overline{\beta}_F = (X_F^{-1} \quad -X_F^{-1}X_R)\begin{pmatrix} Y_B \\ \overline{\beta}_R \end{pmatrix}$$

(7) $$\overline{\beta}_F = X_F^{-1} (Y_B - X_R\overline{\beta}_R)$$

By a list of bound indicators, the elements of $\overline{\beta}_R$ will be assigned to their upper(or lower) bound value.

Define IB to be the index set of the rows of X in the basis, and NB to be the index set of nonbasic rows of X. The basic and nonbasic rows of I will be determined implicitly within the computer code by the value of r and the current column ordering. The discussion here assumes explicit reordering at each iteration.

The deviations for (2), or the reduced costs for the dual problem (6), are given by:

(8) $$d_i = y_i - X_i\bar{\beta} \quad , \; i = 1,2, \ldots , n.$$

The dual variables corresponding to the nonbasic rows are assigned a value depending on the sign of the deviation, that is, $\bar{\pi}_i$ = sign (d_i), $i \varepsilon$ NB. Degeneracy occurs when $d_i = 0$, $i \varepsilon$ NB. In the computer code, the initial value assigned to π_i when $d_i = 0$, $i \varepsilon$ NB, is arbitrarily defined to be +1, and thereafter, the value is determined by the steps of the algorithm. A thorough treatment of the problem of degeneracy can be found elsewhere (see Charnes [7] for example) and this phenomenom will not be examined here.

Define the nonbasic dual variables w_j and v_j to be zero. Define τ_B to be the vector of the basic dual variables. τ_B can be partitioned into τ_{B_F} and τ_{B_R}; that is,

$$\tau_B = (\tau_{B_F} , \tau_{B_R})$$

$$\tau_B = (\tau_{B_1}, \tau_{B_2}, \ldots, \tau_{B_r}, \tau_{B_{r+1}}, \tau_{B_{r+2}}, \ldots, \tau_{B_m})$$

where
$$\tau_{B_F} = (\tau_{B_1}, \tau_{B_2}, \ldots, \tau_{B_r})$$

$$\tau_{B_R} = (\tau_{B_{r+1}}, \tau_{B_{r+2}}, \ldots, \tau_{B_m}) .$$

The vector τ_{B_F} consists of the basic dual variables corresponding to the r rows of X in the basis (X_B), and τ_{B_R} consists of the basic dual variables corresponding to the basic bound constraints. The nonbasic dual variables $\bar{\pi}_i$, $i \varepsilon$ NB, are set to

+1 or -1. The nonbasic w_j and v_j are zero. Thus, from (6), the current values of the basic dual variables can be obtained from the dual constraints.

$$\sum_{i=1}^{r} \bar{\tau}_{B_i} x_{ij} + \sum_{i \varepsilon NB} \bar{\pi}_i x_{ij} = 0, \qquad j=1, 2, \ldots, r$$

$$\sum_{i=1}^{r} \bar{\tau}_{B_i} x_{ij} + \bar{\tau}_{B_j} + \sum_{i \varepsilon NB} \bar{\pi}_i x_{ij} = 0, \qquad j= r+1, r+2, \ldots, m$$

$$\bar{\tau}_B X_B = -H_B^T$$

where $H_{B_j} = \sum_{i \varepsilon NB} \bar{\pi}_i x_{ij}$, $j = 1, 2, \ldots, m$

$$\bar{\tau}_B = -H_B^T X_B^{-1}$$

$$\begin{pmatrix} \bar{\tau}_{B_F} \\ \bar{\tau}_{B_R} \end{pmatrix} = - \left(H_{B_F}^T , H_{B_R}^T \right) \begin{bmatrix} X_F^{-1} & -X_F^{-1} X_R \\ 0 & I \end{bmatrix}$$

(9) $\qquad \bar{\tau}_{B_F} = -H_{B_F}^T X_F^{-1}$

(10) $\qquad \bar{\tau}_{B_R} = -\bar{\tau}_{B_F} X_R - H_{B_R}$

Since this is a primal algorithm, the necessary condition for optimality is dual feasibility. The optimality condition for (2) is:

(11) $\qquad -1 \leq \bar{\pi}_i \leq +1, \qquad i = 1,2,\ldots,n,$

(12) $\qquad \bar{w}_j \leq 0, \qquad j = 1,2,\ldots,m$

(13) and $\qquad \bar{v}_j \geq 0, \qquad j = 1,2,\ldots,m.$

If the optimality condition is not satisfied, then there exists one or more basic rows where condition (11), (12) or (13) is violated. Define τ_{B_k} to be the dual variable most violating a bound restriction. If $k \leq r$, the k-th row of X_B will be chosen to leave the basis. On the other hand, if $k > r$, the k-th row of I will be selected to leave the basis. Define $\rho = \text{sign}(\tau_{B_k})$. The value of ρ indicates if τ_{B_k} is to be increased or decreased, $\rho = -1$ implies that τ_{B_k} is to be increased and $\rho = +1$ implies that τ_{B_k} is to be decreased.

The simplex algorithm of linear programming maintains at zero the current deviations of all basic constraints except the k-th. In other words, the δ_i^+ and δ_i^- of the constraints $X_i\beta + \delta_i^+ - \delta_i^- = y_i$, $i \in IB$, $i \neq k$, remain zero and β_j, $j > r$, $j \neq k$ remain fixed at their upper or lower bound. The deviation of the k-th basic constraint is increased to a value, θ , and another constraint enters the basis with a deviation of zero. The algorithm determines the value, θ, to increase the deviation of the k-th basic constraint while (a) decreasing the objective value (ignoring degeneracy), (b) maintaining primal feasibility and (c) obtaining dual feasibility in the k-th basic variable of the dual problem.

If $k > r$, the algorithm guarantees that the bound restriction on β_k is not violated. The maximum change of β_k possible while still maintaining the feasibility of this constraint is $U_k - L_k$. Thus, an upper bound on the change in the value of β_k is given by θ_1.

(14) $$\theta_1 = \begin{cases} U_k - L_k & \text{for } k > r \\ +\infty & \text{for } k \leq r \end{cases} .$$

The algorithm next calculates Θ_2, the maximum change possible in the deviation of the k-th basic constraint if the primal feasibility in the β_j, $j=1,2,\ldots,r$ is to be maintained. Define

$$\xi = \begin{cases} X^{-1}_{F(k)} & \text{for } k \leq r \\ -X_F^{-1} X_{R(k)} & \text{for } k > r \end{cases}$$

where $X^{-1}_{F(k)}$ is the k-th column of X_F^{-1}, and $X_{R(k)}$ is the k-th column of X_R.

$$\text{(15)} \quad \Theta_2 = \min \begin{cases} (\beta_j - L_j)/(-\rho\xi_j) \text{ for } & \rho\xi_j < 0, \; j=1,2,\ldots,r; \\ (U_j - \beta_j)/(\rho\xi_j) \text{ for } & \rho\xi_j > 0, \; j=1,2,\ldots,r. \end{cases}$$

To calculate the maximum change in the deviation in the k-th basic constraint while *continually decreasing the* absolute sum of the deviations, Θ_3, the algorithm utilizes the basis entry tests of Barrodale and Roberts [4]. These tests determine a nonbasic dual variable which is to enter the basis. The procedure is to calculate the ratio values:

$$\text{(16)} \qquad (d_i)/(\rho\phi_i) \qquad \text{for } \rho\phi_i > 0, \; i=1,2,\ldots,n$$

where

$$\phi_i \begin{cases} (x_{i1}, x_{i2}\ldots, x_{ir})\xi & \text{for } k \leq r, \; i \in NB \\ (x_{i1}, x_{i2}\ldots, x_{ir})\xi + x_{ik} & \text{for } k > r, \; i \in NB \end{cases}$$

The dual variable $\pi_{o(p)}$ is chosen as a candidate to enter the basis by a ranking of the ratios and the following feasibility check on τ_{B_k}

(i) $|\tau_{B_k}| - \sum_{i=1}^{p-1} 2|\phi_{o(i)}| > 1 \quad$ for $k \leq r$

(ii) $|\tau_{B_k}| - \sum_{i=1}^{p-1} 2|\phi_{o(i)}| > 0 \quad$ for $k > r$

(iii) $|\tau_{B_k}| - \sum_{i=1}^{p} 2|\phi_{o(i)}| \leq 1$ for $k \leq r$

and

(iv) $|\tau_{B_k}| - \sum_{i=1}^{p} 2|\phi_{o(i)}| \leq 0$ for $k > r$

where $\left|\frac{d_{o(1)}}{\phi_{o(1)}}\right| \leq \left|\frac{d_{o(2)}}{\phi_{o(2)}}\right| \leq \cdots \leq \left|\frac{d_{o(p-1)}}{\phi_{o(p-1)}}\right| \leq \left|\frac{d_{o(p)}}{\phi_{o(p)}}\right|$

The value of Θ_3 equals $\left|\frac{d_{o(p)}}{\phi_{o(p)}}\right|$. This process is implemented in the computer code version of the algorithm with the partial sort process of Armstrorg, Frome and Kung [2].

The value of Θ is therefore

(17) $\Theta = \min\{\Theta_1, \Theta_2, \Theta_3\}$.

In the case of $\Theta = \Theta_1$, the working basis does not change. On the other hand, if $\Theta \neq \Theta_1$, a new basis is formed and X_F^{-1} must be updated. In any event, the nonbasic dual variables corresponding to the candidate list of ratios with ratio values less than Θ will remain nonbasic but will switch from their current bound to their opposite bound value. The other changes in the primal and dual variables are given as follows.

If $\Theta = \Theta_1$, β_k switches bound value and X_B is not changed. Relating this situation to the dual problem, the nonbasic w_j enters the basis and the basic v_j leaves the basis if β_k switches from its lower to upper bound. On the other hand, if β_k switches from its upper to lower bound value, the non-basic v_j enters the basis and the basic w_j leaves the basis.

If $\Theta = \Theta_2$, a nonbasic row of I will enter the basis, and X_{B_k} will leave the basis, where X_{B_k} is the k-th row of the basis X_B. In the dual problem, the nonbasic w_j or v_j will enter the basis and τ_{B_k} will leave the basis.

If $\Theta = \Theta_3$, a nonbasic row of X will become basic, the k-th row of X_B (associated with the basic dual variable, τ_{B_k}) will leave the basis. In terms of the dual problem, the nonbasic π_s will become basic.

There are five cases in the updating process of X_F^{-1}.

Case 1: If $k \leq r$, $\Theta = \Theta_3$, this is the situation when a row of X_B is leaving the basis and a row of X is entering the basis. The update of X_F^{-1} is the standard simplex pivot.

Case 2: If $k \leq r$, $\Theta = \Theta_2$, this is the situation when a row of X_B is leaving the basis and a row of I is entering the basis. Since X_F^{-1} contains the rows of X in the basis, the dimension of X_F^{-1} will be decreased by 1.

Case 3: If $k > r$, $\Theta = \Theta_1$, X_F^{-1} does not change.

Case 4: If $k > r$, $\Theta = \Theta_3$, this is the situation when a row of I is leaving the basis, and a row of X is entering the basis. The dimension of X_F^{-1} will be increased by 1.

Case 5: If $k > r$, $\Theta = \Theta_2$, this is the situation when a row of I is leaving the basis and a row of I is entering the basis. The dimension of X_F^{-1} does not change.

The algorithm then updates the deviations by

(18) $$d_i \leftarrow d_i - \rho\Theta X_i X_{B(k)}^{-1}, \quad i=1,2,\ldots,n.$$

The algorithm also updates the indicators used for re-ordering purposes. The iterative process continues until the condition $-1 \leq \tau_{B_i} \leq +1$, $i=1,2,\ldots,r$ and $\tau_{B_i} \leq 0$ or $\tau_{B_i} \geq 0$, $i=r+1,r+2,\ldots,m$ is satisfied.

3. Steps of the Algorithm

In this section a step-by-step statement of the algorithm is outlined.

Step 1. The initial values of $\bar{\beta}_R$ are found from the following:

(i) If $L_j \leq \beta_j \leq U_j$, the j-th element of $\bar{\beta}$, $\bar{\beta}_j$, is set equal to L_j.

(ii) If $L_j \leq \beta_j \leq \infty$, set $\bar{\beta}_j = L_j$.

(iii) If $-\infty \leq \beta_j \leq U_j$, set $\bar{\beta}_j = U_j$.

After interchanges, $\bar{\beta} = (\bar{\beta}_F, \bar{\beta}_R)^T$, where $\bar{\beta}_R$ is a vector of values equal to the upper or lower bound values of the restricted parameters.

Step 2. Formulate the initial basis, X_B.

(i) If all the parameters are restricted, X_B is an m by m identity matrix.

(ii) When all the parameters of the problem do not have bound restrictions, the algorithm attempts to form the basis matrix X_B by choosing a full rank X_F matrix from the coefficients of the unrestricted parameters. If this procedure fails to create an initial X_B, the algorithm advances to (iii).

(iii) A full rank matrix X_B is now formed by including in the initial basis as many additional rows from the identity of (4) as is necessary. The β_j's corresponding to these rows are assigned upper and lower bounds of plus and minus infinity.

Only the values of X_F^{-1} and the corresponding parameter indices of X_F^{-1} are used for the remaining computations.

Step 3. Calculate $\bar{\beta}_F$ and d_i based on (7) and (8). Determine the values of $\bar{\pi}_i$, $i \in NB$, based on the sign of d_i.

Step 4. Calculate τ_{B_F} and τ_{B_R} based on (9) and (10).

Step 5. Check for the optimality conditions given by (11), (12) and (13). If these are satisfied, terminate. Otherwise, go to step 6.

Step 6. Determine the value of ρ from the sign of τ_{B_k} where τ_{B_k} is the dual variable most violating a bound restriction.

Step 7. Calculate Θ_1, Θ_2, Θ_3, and Θ from (14), (15), (16) and (17).

Step 8. If $\Theta = \Theta_1$, β_k switches to its opposite bound value. Go to step 12.

Step 9. If $\Theta = \Theta_2$, the dimension of X_F^{-1} is decreased by 1 if $k \leq r$. On the other hand, the dimension of X_F^{-1} does not change if $k > r$. Go to step 11.

Step 10. If $\Theta = \Theta_3$, the dimension of X_F^{-1} remains the same if $k \leq r$. If $k > r$, the dimension of X_F^{-1} will be increased by 1. Go to step 11.

Step 11. Update X_F^{-1} and the other basic indicators.

Step 12. Update the deviations based on (18), and the values of π_i, $i \in NB$, based on the sign of the updated values of d_i. Go to step 4.

4. Computational Results

To evaluate the efficiency of the algorithm given here, a FORTRAN version of the algorithm presented here called RESL1 was tested against the algorithm INTBND developed by Armstrong and Hultz [3]. The INTBND code solves least absolute value problems with arbitrary linear constraints on the parameters. Both codes employ the revised simplex method of linear programming. The observations for the study have been drawn from various uniform distributions using a random number generator. All the problems were solved by the CDC 6600 in the University of Texas at Austin. The results presented in Table 1 are mean times and iteration counts for a set of 5 problems with the same characteristics.

Table 1

(A comparison between RESL1 and INTBND)

Number of parameters	Number of observations	Number of unrestricted parameters	Time (in CPU milleseconds)		Number of iterations	
			RESL1	INTBND	RESL1	INTBND
5	50	0	29	56	5	6
5	50	2	75	115	9	11
10	100	0	158	376	11	18
10	100	5	679	968	31	44
15	150	0	323	1004	14	28
15	150	15	2750	5629	49	144
15	200	0	830	1671	23	37
15	200	10	2111	5463	39	95
20	200	0	1383	3016	32	55
20	200	12	6230	9334	80	152
20	250	0	1693	3897	34	60
20	250	14	7565	12566	77	165
25	250	0	1263	5663	27	73
25	250	20	8970	14445	97	251
25	300	0	1311	6086	24	68
25	300	18	14128	18751	139	275
30	300	0	2543	9142	41	86
15	500	10	7644	16872	71	147
15	500	0	1673	4643	26	64

Our computational study indicates that the algorithm RESL1 is consistently faster than INTBND on all problem sizes. Also, RESL1 utilizes a reduced basis inverse and requires less computer storage than INTBND.

4. Conclusion

Since the time of Laplace [16], minimizing the sum of absolute deviations has been considered as a criterion for parameter estimation. Recently, LAV estimators have been widely examined as a robust estimation technique [11] and with this interest has come a demand for efficient algorithms to provide LAV estimates for a variety of statistical models. This demand has been satisfied primarily by the specialization of linear programming to take advantage of the distinctive characteristics of the problem.

This paper presents an approach to solve least absolute value problems with bound restrictions on the parameters. The algorithm presented here exploits the structure of the problem to maintain a compact representation of the basis inverse. All computations required in the algorithm can be performed by means of a reduced basis. The algorithm also combines several standard simplex pivots into one. Computational experience with a FORTRAN version of this algorithm compared to a FORTRAN version of the algorithm developed by Armstrong and Hultz [3] indicates its superiority in terms of computer time and storage.

REFERENCES

1. Armstrong, R. D. and Frome, E. L. (1976), "A Branch-and-Bound of a Restricted Least Squares Problem," Technometrics, 18, 447-450.
2. Armstrong, R. D., Frome, E. L., and Kung, D. S. (1979), "A Revised Simplex Algorithm for the Absolute Deviation Curve-fitting Problem," Communications in Statistics, B8(2), 175-190.
3. Armstrong, R. D. and Hultz, J. W. (1977), "An Algorithm for a Restricted Discrete Approximation Problem in the L_1 Norm," SIAM Journal on Numerical Analysis, 14, 555-565.
4. Barrodale, I. and Roberts, F. D. K. (1973), "An Improved Algorithm for Discrete L_1 Linear Approximation," SIAM Journal on Numerical Analysis, 10, 839-848.
5. Barrodale, I. and Roberts, F. D. K. (1974), "Solution of an Overdetermined System of Equations in the L_1 Norm," Communications of the ACM, 17, 319-320.
6. Barrodale, I. and Roberts, F. D. K. (1977), "An Efficient Algorithm for Discrete L_1 Linear Approximation with Linear Constraints," Department of Mathematics Report, University of Victoria.
7. Charnes, A. (1952), "Optimality and Degeneracy in Linear Programming," Econometrica, 20, 160-170.

8. Charnes, A., Cooper, W. W., and Ferguson, R. (1955), "Optimal Estimation of Executive Compensation by Linear Programming," Management Science, 2, 138-151.

9. Charnes, A., Granot, F., and Phillips, F. (1977), "An Algorithm for Solving Interval Linear Programming Problems," Operations Research, 25, 688-695.

10. Davies, M. (1967), "Linear Approximation Using the Criterion of Least Total Deviation," Journal of the Royal Statistical Society, 29, 101-109.

11. Gentle, J. E. (1977), "Least Absolute Values Estimation: An Introduction," Communications in Statistics, B6(4), 313-328.

12. Harter, H. L. (1975), "The Method of Least Squares and Some Alternatives - Part V," International Statistical Review, 43, 269-278.

13. Hogg, R. V. and Randles, R. H. (1975), "Adaption Distribution - Free Regression Methods and Their Application," Technometrics, 17, 399-407.

14. Huber, P. J. (1972), "Robust Statistics: A Review," The Annals of Mathematical Statistics, 43, 1041-1067.

15. Judge, G. G. and Takayama, T. (1966), "Inequality Restrictions in Regression Analysis," Journal of American Statistical Association, 61, 166-181.

16. Laplace, P. S. (1793), "Sur Quelques Points du Systems du Monde," Memoires de l'Academic Royale des Sciences de Paris Annee 1789, 1-87. Reprinted in Oeuvres Completes de Laplace, Vol. II, Paris, Gauthier-Villars (1895), 477-558.

17. Spyropoulos, K., Kiountouzis, E., and Young, A. (1973), "Discrete Approximations in the L_1 Norm," The Computer Journal, 16, 180-186.

18. Waterman, M. S. (1974), "A Restricted Least Squares Problem," Technometrics, 16, 135-136.

Unclassified
Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) Center for Cybernetic Studies The University of Texas at Austin	2a. REPORT SECURITY CLASSIFICATION Unclassified 2b. GROUP

3. REPORT TITLE
An Algorithm for a Least Absolute Value Regression Problem with Bounds on the Parameters

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

5. AUTHOR(S) (First name, middle initial, last name)
Ronald Armstrong, Mabel Kung

6. REPORT DATE June 1980	7a. TOTAL NO. OF PAGES 20	7b. NO. OF REFS 18

8a. CONTRACT OR GRANT NO. N00014-75-C-0569 and N00014-80-C-0242 b. PROJECT NO. NR047-021 and NR047-071 c. d.	9a. ORIGINATOR'S REPORT NUMBER(S) CCS 378 9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)

10. DISTRIBUTION STATEMENT
This document has been approved for public release and sale; its distribution is unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY Office of Naval Research (Code 434) Washington, DC

13. ABSTRACT

This paper presents a special purpose linear programming algorithm to solve a least absolute value regression problem with upper and lower bounds on the parameters. The algorithm exploits the problem's special structure by maintaining a compact representation of the basis inverse and by allowing for the capability to combine several simplex iterations into one. Computational results with a computer code implementation of the algorithm is given.

DD FORM 1 NOV 65 1473 (PAGE 1)
S/N 0101-807-6811

Unclassified
Security Classification
A-31408

Unclassified

Security Classification

14 KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Regression						
Least Absolute Value						
Upper Bounding						

DD FORM 1 NOV 65 1473 (BACK)

S/N 0102-014-6600

Unclassified

Security Classification

A-31409