

New Algorithms for L₁ Norm Regression

Bijan Bidabad¹

Keywords: *L₁ norm, Regression, Algorithm, Computer*

Abstract

In this paper, we propose four algorithms for L₁ norm computation of regression parameters, where two of them are more efficient for simple and multiple regression models. However, we start with restricted simple linear regression and corresponding derivation and computation of weighted median problem. In this respect a computing function is coded. With discussion on the m parameters model we continue to expand the algorithm to include unrestricted simple linear regression and two crude and efficient algorithms are proposed. The procedures are then generalized to the m parameters model by presenting two new algorithms, where the algorithm 4 is selected as more efficient. Various properties of these algorithms are discussed.

1. Introduction

In Bidabad (1989a,b), various aspects of the L₁ norm regression were reviewed. We observed that the L₁ norm criterion is going to find its place in scientific analysis. Since it is not computationally comparable with other criteria such as L₂ norm, it needs more work to make it a hand tool. The closed form of the solution of the L₁ norm estimator has not been derived yet, and therefore, makes further inferences of the properties of this estimator difficult. Any attempt to give efficient computational algorithms which may introduce significant insight into the different characteristics of the problem is desirable. In this regard, we shall try to give a general procedure in this paper to solve L₁ norm linear regression problem. The proposed algorithms are based on a special descent method and use discrete differentiation technique. Primary designs of the algorithms have been discussed by Bidabad (1987a,b,88a,b). By manipulating these algorithms, more efficient ones were introduced by Bidabad (1989a,b) which has been shown to have better performance than other existing algorithms.

Consider the following regression model,

$$\begin{aligned} & \text{m} \\ & y_i = \sum_{j=1}^m \beta_j x_{ji} + u_i \quad i=1, \dots, n \end{aligned} \tag{1}$$

where β_j , $j=1, \dots, m$ are population parameters to be estimated, y_i , x_{ji} and u_i are dependent, independent and random variables respectively. We wish to estimate β_j 's by minimizing the expression,

¹ Research professor of economics, Monetary and Banking Research Academy, Central Bank of Iran, Pasdaran, Zarabkhaneh, Tehran, 16619, Iran. bijan_bidabad@msn.com.

I should express my sincere thanks to Professor Yadolah Dodge from Neuchatel University of Switzerland who taught me many things about L₁ norm when he was my Ph.D.dissertation advisor.

$$S = \sum_{i=1}^n |u_i^\wedge| = \sum_{i=1}^n |y_i - \sum_{j=1}^m \beta_j^\wedge x_{ji}| \quad (2)$$

Suppose $m=1$, then expression (2) reduces to,

$$S = \sum_{i=1}^n |y_i - y_i^\wedge| = \sum_{i=1}^n |y_i - \beta^\wedge x_i| = \sum_{i=1}^n S_i \quad (3)$$

A typical element, $S_i = |y_i - \beta^\wedge x_i|$ can be viewed as a broken line in the $Sx\beta^\wedge$ plane composed of two half-lines. S_i attains its minimum which is zero at,

$$\beta_i^\wedge = y_i/x_i \quad (4)$$

The slopes of the half-lines to the left and right of β_i^\wedge are $-|x_i|$ and $|x_i|$ respectively. So, S_i 's are all convex and hence their sum S is also convex with a slope at any β^\wedge equal to sum of the slopes of S_i 's at that value of β^\wedge . Since the slope of each S_i changes only at the corresponding β_i^\wedge , the minimum of S will lie on one of the β_i^\wedge . Thus, the regression line will pass through origin with the slope equal to β_i^\wedge which minimizes S . On the other hand, to find the L_1 norm estimate of β we need to find only one observation (see, Karst (1958), Taylor (1974)). Furthermore, Taylor (1974) concludes that: "This implies, of course, that the regression line must pass through the observation corresponding to the minimizing i . The regression line, therefore, is determined by the point of origin and the observation associated with minimizing β_i^\wedge ". But he did not continue this approach, that is, minimizing S with respect to subscript i . In this paper as Bidabad (1987a,88a) a first endeavor is to develop this point of view. In the next section after rewriting (3) in a suitable fashion, the value of i is determined by using discrete differentiation.

By a similar discussion it can be concluded that when the number of parameters is m , m observations must lie on the regression hyperplane. On the other hand, m equations of the form given by (5) are necessary to specify the regression hyperplane (see also Bidabad (1989a,b)).

$$y_i - \sum_{j=1}^m \beta_j^\wedge x_{ji} = 0 \quad (5)$$

Now, let us proceed with the simplest regression model.

2. Restricted simple linear regression

For model (1) consider the case of one independent variable with no intercept, namely, $y_i = \beta x_i + u_i$. To find the L_1 norm estimate of β , the following procedure can be suggested.

$$S = \sum_{i=1}^n |u_i| = \sum_{i=1}^n |y_i - \beta x_i| = \sum_{i=1}^n (y_i - \beta x_i) \text{sgn}(y_i - \beta x_i) = \sum_{i=1}^n |x_i|(y_i/x_i - \beta) \text{sgn}(y_i/x_i - \beta) \quad (6)$$

Let $z_i = y_i/x_i$ and sort z_i in descending order. Rename the resultant ordered z_i , $i=1, \dots, n$ to z_h , $h=1, \dots, n$. Elements of z_h should have the following property:

$$z_h > z_l \text{ if } h < l \text{ for all } h, l = 1, \dots, n.$$

Rewrite (6) with ordered observations as,

$$S = \sum_{h=1}^n |x_h|(z_h - \beta) \text{sgn}(z_h - \beta) \quad (7)$$

Let us denote the observation which will be on the regression line by (x_{t+1}, y_{t+1}) - that is the $(t+1)^{\text{th}}$ observation in the z_h array. Value of z_h is the slope of a ray passing through the origin and the h^{th} observation. Therefore,

- if, $h < t+1$ then, $z_h > \beta$ and $u_h > 0$
- if, $h = t+1$ then, $z_h = \beta$ and $u_h = 0$
- if, $h > t+1$ then, $z_h < \beta$ and $u_h < 0$

We can now rewrite (7) as follows,

$$S = \sum_{h=1}^t |x_h|(z_h - \beta) - \sum_{h=t+1}^n |x_h|(z_h - \beta) \quad (8)$$

To find the minimum of S, we need to differentiate it with respect to β and subscript t, because both β and t are unknowns. Note that β has continuous domain and t has discrete domain. Thus,

$$\frac{\delta S}{\delta \beta} = - \sum_{h=1}^t |x_h| + \sum_{h=t+1}^n |x_h| \quad (9)$$

The differentiation of S with respect to subscript t must be discrete derivative (see, Bender and Orszag (1978), Clarke (1983)):

$$\frac{\Delta(S)}{\Delta(t)} = \lim_{\Delta(t) \rightarrow 1} \frac{S[t+\Delta(t)] - S(t)}{\Delta(t)} = S(t+1) - S(t) =$$

$$\sum_{h=1}^{t+1} |x_h|(z_h - \beta) - \sum_{h=t+2}^n |x_h|(z_h - \beta) - \sum_{h=1}^t |x_h|(z_h - \beta) + \sum_{h=t+1}^n |x_h|(z_h - \beta) = 2|x_{t+1}|(z_{t+1} - \beta) = 0$$

thus,

$$\beta^\wedge = z_{t+1} = y_{t+1}/x_{t+1} \quad (10)$$

It should be noted that if $x_{t+1}=0$ then the value of $z_{t+1}=l$, and when the array z is sorted, infinite values of z locate at extremes of two tails of the array and make no problem in (10). Remember that equation (10) is the same as relation (4). Equations (9) and (10) are two equations with two unknowns t and β . The variable t can be found by rewriting (9) as follows,

$$D_k \equiv \sum_{h=1}^k |x_h| - \sum_{h=k+1}^n |x_h| \quad k=1, \dots, n \quad (11)$$

It is obvious that D_k is an increasing function of k. When k increases from one to n, D_k attains different values which increases from negative to positive. So, initially k is set equal to one and D_k is computed accordingly. If D_k is negative, k is increased by one and the procedure is repeated until D_k becomes positive. When D_k reaches the first positive value, then, $t+1=k$. By this procedure, value of $t+1$ is found. The observation corresponding to this subscript ($t+1=k$) is selected (x_{t+1}, y_{t+1}) . The L_1 norm estimate of β is found by substituting the values of x_{t+1} and y_{t+1} into (10), (see, Bidabad (1987a,88a)).

The presented procedure is essentially the weighted median procedure of Laplace (1818). Namely, this solution is the steps that Laplace took. Main difference is the mathematical derivation. Laplace found this solution by analyzing algebraic characteristic of the L_1 norm objective function; while the above procedure exactly differentiates the objective function. Another new contribution of this procedure is application of the partial discrete differentiation over subscript accompanying with conventional differentiation on a regular variable with continuous domain. It should be remembered that Boscovich (1757) solved this problem by a geometrical procedure and Karst (1958) by an analytical method.

2.1 Computation of weighted median

Two series of computations are necessary to compute weighted median. One sorting algorithm is essential to sort the z array and restoring the corresponding subscripts to compute the second part of calculation to find the optimal value of k by using D_k defined by (11).

Efficient sorting algorithms exist for the first part of the computation. The algorithms 'quicksort' of Hoare (1961,62), 'quicksort' of Scowen (1965) and 'sort' of Singleton (1969) have desirable performances and efficiencies. For the second part of the computation there is no special purpose procedure, but Bloomfield and Steiger (1980) used the partial sorting of Chambers (1971) to give an efficient way to combine the two steps of sorting the z array and finding the value of k . The superiority of this procedure is in sorting the smaller segments of z array rather than all its elements. With some modification, this procedure is used for our proposed algorithms in the forthcoming sections. This procedure can be stated as the following function.

FUNCTION LWMED (n,ys,w,l)

Step 0) Initialization.

Real: ys(n), w(n).

Integer: l(n), hi.

Set: ii=0, shi=0, slo=0, sz=0, sp=0, sn=0.

Step 1) Compute left, middle and right sum of weights.

Do loop for $i=1,n$: $w(i)=|w(i)|$; if $ys(i)<0$, then $sn=sn+w(i)$, if $ys(i)>0$, then $sp=sp+w(i)$, if $ys(i)=0$ then $sz=sz+w(i)$; end do.

If $shi \leq slo$ then go to step 2.b, otherwise go to step 2.a.

Step 2) Assign subscripts for arrays.

a. Let: $shi=0$.

Do loop for $i=1,n$: if $ys(i) \leq 0$ go to continue, otherwise $ii=ii+1$, $l(ii)=i$, continue, end do.

Go to step 2.c.

b. Let: $slo=0$.

Do loop for $i=1,n$: if $ys(i)>0$ go to continue, otherwise $ii=ii+1$, $l(ii)=i$, continue, end do.

c. Let: $lo=1$, $hi=ii$.

Step 3) Check for solution.

If $hi>lo+1$ then go to step 4, otherwise $lwmed=l(lo)$.

If $lo=hi$ return, otherwise if $ys(l(lo)) \leq ys(l(hi))$ go to step 3.a,
otherwise $lt=l(lo)$, $l(lo)=l(hi)$, $l(hi)=lt$, $lwmed=l(lo)$.

a. If $shi+w(l(hi))>slo+w(l(lo))$ then set $lwmed=l(hi)$, otherwise return.

Step 4) Divide the string into two halves then sort.

Set: $mid=(lo+hi)/2$, $lop=lo+1$, $lt=l(mid)$, $l(mid)=l(lop)$, $l(lop)=lt$.

a. If $ys(l(lop)) \leq ys(l(hi))$ then go to step 4.b, otherwise $lt=l(lop)$, $l(lop)=l(hi)$, $l(hi)=lt$.

b. If $ys(l(lo)) \leq ys(l(hi))$ then go to step 4.c, otherwise $lt=l(lo)$, $l(lo)=l(hi)$, $l(hi)=lt$.

c. If $ys(l(lop)) \leq ys(l(lo))$ then go to step 5, otherwise $lt=l(lop)$, $l(lop)=l(lo)$, $l(lo)=lt$.

Step 5) Compute the accumulation of weights.

Let: $lwmed=l(lo)$, $i=lop$, $j=hi$, $xt=ys(lwmed)$, $tlo=slo$, $thi=shi$.

a. Set: $tlo=tlo+w(l(i))$, $i=i+1$.

If $ys(l(i)) < xt$ then go to step 5.a, otherwise go to step 5.b.

b. Let: $thi=thi+w(l(j))$, $j=j-1$.

If $ys(l(j)) > xt$ then go to step 5.b, otherwise if $j \leq i$ then go to step 6,

otherwise $lt=l(i)$, $l(i)=l(j)$, $l(j)=lt$, go to step 5.a.

Step 6) Test for solution.

Let: $test=w(lwmed)$.

If $i \neq j$ then go to step 6.a, otherwise $\text{test} = \text{test} + w(l(i))$, $i = i + 1$, $j = j - 1$.

- a. If $\text{test} \geq |\text{thi}-\text{tlo}|$ then return, otherwise, if $\text{tlo} > \text{thi}$ then step 6.b, otherwise $\text{slo} = \text{tlo} + \text{test}$, $\text{lo} = \text{i}$, go to step 3.
 - b. Let: $\text{shi} = \text{thi} + \text{test}$, $\text{lo} = \text{lop}$, $\text{hi} = \text{j}$. go to step 3.

END

2.2 Discussion of the m parameters model

The procedure applied to the simple one parameter model can not be simply generalized to the m parameters model. This difficulty is due to the fact that we can not reorder observations in such a way that their corresponding residuals are increasingly or decreasingly ordered. On the other hand, to apply the discrete differentiation technique, primarily (2) should be rewritten as follows:

$$S = \sum_{h=1}^t (y_h - \sum_{j=1}^m \beta_j x_{jh}) - \sum_{h=t+1}^m (y_h - \sum_{j=1}^m \beta_j x_{jh}) \quad (12)$$

Expression (12) which is free of absolute value sign is generalization of (8) for m parameters. But our first problem is to find a logic which enables us to form (12). On the other hand, we need to reorder observations in such a way that when h is less, equal or greater than $t+1$, u_h be greater, equal or less than zero respectively; and as h increases from one to n , the corresponding residual u_h decreases accordingly.

If (2) could be written as (12), we could again differentiate it with respect to β_j for all $j=1,\dots,m$ and t . Differentiating S with respect to β_j for all $j=1,\dots,m$ would give,

$$\frac{\delta S}{\delta \beta_j} = - \sum_{h=1}^t x_{jh} + \sum_{h=t+1}^n x_{jh} \quad j=1, \dots, m \quad (13)$$

In order to differentiate S with respect to discrete domain variable t we would have,

$$\frac{\Delta(S)}{\Delta(t)} = \lim_{\Delta(t) \rightarrow 1} \frac{S[t+\Delta(t)] - S(t)}{\Delta(t)} = S(t+1) - S(t) =$$

$$\sum_{h=1}^{t+1} \sum_{j=1}^m y_h - \sum_{h=t+2}^t \sum_{j=1}^m u_h = \sum_{h=1}^{t+1} \sum_{j=1}^m y_h - \sum_{h=t+1}^t \sum_{j=1}^m y_h = 0 \quad (14)$$

m distinct values for t can be computed from (13) by using the following D_{kj} for each explanatory variable. The procedure to compute t is similar to what we did in (11) for the one parameter model.

$$D_{kj} \equiv \sum_{h=1}^k x_{jh} - \sum_{h=k+1}^n x_{jh} \quad j=1, \dots, m; i=1, \dots, n \quad (15)$$

It should be noted that these m values for t , all give a unique value for S in (12), because all of the errors corresponding to these t 's have zero values. This becomes clear by zero residuals property of the L_1 norm regression that there exist m points on the regression hyperplane which have zero residuals. In (12), these m points locate sequentially, because we had reordered observations in such a way that, when h increases from one to n , the errors decrease from greatest positive to lowest negative values. So, these m zero errors will locate one after another nearly in the middle of n elements sequence of (12). Corresponding to these m values of t , m observations are recognized. Substitute the values of these m observations into (14), m equations are found which can be solved simultaneously for m unknown β_j 's. These last m equations again confirm the above cited property of the L_1 norm regression.

Since for the m parameters model, the ordering of residuals before computing optimal values of β_j 's is not possible, another device should be adopted to compute the L_1 norm estimates of the multiple linear regression parameters.

3. Unrestricted simple linear regression

To find the L_1 norm estimates of $\{\beta_j, j=1, \dots, m\}$ for model (1), we shall try to propose algorithms to search for those observations on the optimal regression hyperplane. For $m=2$ and $x_{1i}=0$, $i=1, \dots, n$; (1) reduces to restricted simple linear regression with only one parameter. The L_1 norm estimate of the slope β_2 for simple model can be obtained by the algorithm given in the previous sections. Let us now consider a simple unrestricted linear model in which $m=2$ and $x_{1i}=1$ for all $i=1, \dots, n$; namely,

$$y_i = \beta_1 + \beta_2 x_{2i} + u_i \quad (16)$$

3.1 Algorithm 1

For the model given in (16), the L_1 norm objective function S to be minimized will be,

$$S = \sum_{i=1}^n |y_i - \beta_1 - \beta_2 x_{2i}| \quad (17)$$

Let k_1 denote a subscript which belongs to the range of one to n, and assume that k_1^{th} observation (y_{k1}, x_{2k1}) is a candidate to be on the regression line. If this is the case, then $u_{k1}=0$ and we can transfer the origin of YxX_2 coordinates to the point (y_{k1}, x_{2k1}) without any loss. For this, we should rewrite all observations as deviates from the point (y_{k1}, x_{2k1}) ,

$$y_i^{k1} = y_i - y_{k1}, \quad x_{2i}^{k1} = x_{2i} - x_{2k1} \quad i=1, \dots, n \quad (18)$$

Where the terms with superscripts $k1$ denote deviated values. Rearrange the terms;

$$y_i = y_i^{k1} + y_{k1}, \quad x_{2i} = x_{2i}^{k1} + x_{2k1} \quad i=1, \dots, n \quad (19)$$

Now, substitute (19) into (17), then, our L_1 norm minimization problem can be redefined as,

$$\min_{\beta_1, \beta_2} S_{k1} = \sum_{i=1}^n |y_i^{k1} - \beta_2 x_{2i}^{k1} + (y_{k1} - \beta_1 - \beta_2 x_{2k1})| \quad (20)$$

Since, we assumed that the $k1^{\text{th}}$ observation is on the regression line, the expression $y_{k1} - \beta_1 - \beta_2 x_{2k1} = 0$. Thus (20) reduces to,

$$\min_{\beta_2} S_{k1} = \sum_{i=1}^n |y_{ik1} - \beta_2 x_{2i}^{k1}| \quad (21)$$

Solution of the optimization problem (21), is the same as the one described before for one parameter linear model. Note that when $u_{k1}=0$, then the minimum value of (21) is equal to the minimum value of (17).

Let β_2 derived by minimizing (21) be denoted by β_2^{k1} . By changing $k1$ from one to n and minimizing (21), $\beta_2^1, \dots, \beta_2^n$ are attained accordingly. Now the question is: what value of $k1$ minimizes (17)? In other words, which observations are on the optimal regression line? Note that in the two parameters L_1 norm linear model, there exists at least two observations with zero errors. Transferring of YxX_2 coordinates to both these points does not change the minimum of (17). Suppose p^{th} and q^{th} are those observations on the regression line, thus,

$$u_p = y_p - \beta_1 - \beta_2 x_{2p} = 0, \quad u_q = y_q - \beta_1 - \beta_2 x_{2q} = 0 \quad (22)$$

Rewriting (18) and (19) for $k1=p, q$ and substituting them into (21), yields,

$$S_p = \sum_{i=1}^n |y_i^p - \beta_2 x_{2i}^p| \quad (23)$$

$$S_q = \sum_{i=1}^n |y_i^q - \beta_2 x_{2i}^q| \quad (24)$$

Using (18) and rewriting (23) and (24), it is shown that S_p is equal to S_q ,

$$S_p = \sum_{i=1}^n |y_i - \beta_2 x_{2i} - (y_p - \beta_2 x_{2p})| \quad (25)$$

$$S_q = \sum_{i=1}^n |y_i - \beta_2 x_{2i} - (y_q - \beta_2 x_{2q})| \quad (26)$$

S_p is equal to S_q if and only if the two parentheses inside the absolute value signs of (25) and (26) are equal. This can be concluded by solving the two equations of (22) for β_1 . That is,

$$\beta_1 = y_p - \beta_2 x_{2p} = y_q - \beta_2 x_{2q} \quad (27)$$

The equality of S_q and S_p is because the points p and q are on the regression line and therefore $u_p = u_q = 0$. Thus, $S_p = S_q$, so, the values of β_2^p and β_2^q derived by minimizing either S_p or S_q must be equal. This gives a criterion to find the desired β_2 from all β_2^{k1} for $k1=1, \dots, n$. That is, when $\beta_2^p = \beta_2^q$. The estimated value of β_2 is denoted by $\hat{\beta}_2$. Value of $\hat{\beta}_1$ is simply computed by (27). Now let us summarize the whole procedure for finding the values of $\hat{\beta}_1$ and $\hat{\beta}_2$ for the model given by (16).

Crude algorithm 1

Step 0) Set $k1=1$.

Step 1) Compute (18).

Step 2) Minimize (21) by using weighted median procedure and find β_2^{k1} .

Step 3) Check if $\beta_2^{k1} = \beta_2^{k1} - k$ for $0 < k < k1$, then set $\hat{\beta}_2 = \beta_2^{k1}$ and $\hat{\beta}_1 = y_{k1} - \hat{\beta}_2 x_{2k1}$ then stop.

Step 4) Increase $k1$ by one and go to step 1.

3.2 Algorithm 2

The crude algorithm 1 for finding $\hat{\beta}_1$ and $\hat{\beta}_2$ is not computationally efficient because it usually requires testing the majority of observations. In order to make this algorithm efficient, some elaborations are necessary.

Instead of setting $k1=1$, let us set $k1$ equal to an arbitrary integer value such as a where a is any integer from one to n. Suppose now, $u_a = y_a - \beta_1 - \beta_2 x_{2a} = 0$, and rewrite (21) as,

$$\min_{\beta_2} S_a = \sum_{i=1}^n |y_i^a - \beta_2 x_{2i}^a| \quad (28)$$

Minimizing (28) gives β_2^a which is equal to

$$\beta_2^a = \frac{y_b^a - y_a}{x_{2b}^a - x_{2a}} = \frac{y_b - y_a}{x_{2b} - x_{2a}} \quad (29)$$

On the other hand, by pivoting on the a^{th} observation another point such as b is found where b refers to that observation which has zero error in the minimum solution of (28), so, $u_b = y_b - \beta_1 - \beta_2 x_{2b} = 0$. Let us denote the minimum of S_a in (28) as S_a^* , then,

$$S_a^* = \sum_{i=1}^n |y_i^a - \beta_2^a x_{2i}^a| \quad (30)$$

Note that,

$$\beta_1 = y^a - \beta_2^a x_2^a = y_b - \beta_2^a x_{2b} \quad (31)$$

This comes from multiplying (29) by its denominator and rearranging the terms. Substitute (31) in (17),

$$S = \sum_{i=1}^n |y_i - y_a - \beta_2^a(x_{2i} - x_2^a)| = \sum_{i=1}^n |y_i - y_b - \beta_2^a(x_{2i} - x_{2a})|$$

or,

$$S = \sum_{i=1}^n |y_i^a - \beta_2^a x_{2i}^a| = \sum_{i=1}^n |y_i^b - \beta_2^a x_{2i}^b| \quad (32)$$

Using (30) the first sum in (32) is S_a^* and the second sum is S_b evaluated at $\beta_2 = \beta_2^a$. Thus, it can be concluded that,

$$S_a^* = S_b \Big|_{\beta_2 = \beta_2^a} \quad (33)$$

S_a^* is at minimum but $S_b \Big|_{\beta_2 = \beta_2^a}$ still can be minimized for other values of β_2 .

Therefore, an important result is derived, that is,

$$S_a^* > S_b^* \quad (34)$$

The inequality of (34) guarantees that if we choose an arbitrary point to transfer the origin of coordinates to it and minimizing the objective function (21) another point is found, then transferring the origin to this newly found point decreases the total sum of absolute errors. Therefore, at each transference we get closer to the minimum of S . By a similar discussion, this conclusion can be generalized as,

$$S_a^* > S_b^* > S_c^* > S_d^* \dots \quad (35)$$

Note that a is an arbitrary starting point. The point b is derived by minimizing S_a , c is derived by minimizing S_b and d is derived by minimizing S_c and so on.

Now, the question is, when the minimum value of S is reached? Suppose $S^* = S_f^*$; by transferring the origin of coordinates to the point f and minimizing S_f , the g^{th} observation is derived. When $S^* = S_f^*$ by minimizing S_g , f^{th} observation is again found, because $S_g^* = S_f^* = S^*$ and the g^{th} and f^{th} observations are both on the L_1 norm regression line. This conclusion can be a criterion to stop the procedure. Hence,

$$S_a^* > S_b^* > S_c^* > S_d^* \dots > S_f^* = S_g^* = S^* \quad (36)$$

It should be noted that if the minimum solution of S is not unique, that is function S has a horizontal segment, the procedure stops when it reaches the first minimum solution.

Now, let us introduce the whole stages of this algorithm to find the L_1 norm estimates of β_1 and β_2 in the simple linear model (16).

Algorithm 2

Step 0) Select an arbitrary integer value a between one to n and set $k1=a$.

Step 1) Compute (18) with $k1=a$.

Step 2) Minimize (21) by using weighted median procedure and find the appropriate observation on the line, namely observation b .

Step 3) Compute (18) with $k1=b$.

Step 4) Minimize (21) and find that observation on the line; observation c .

Step 5) Check if $c=a$, then $\beta_2^c = y_b^c / x_{2b}^c$, $\beta_1^c = y_c - \beta_2^c x_{2c}$ and stop.

Step 6) Set $a=b$ and go to step 1.

Operationally, the following steps are taken.

PROGRAM BL1S

Step 0) Initialization.

Parameter: n.

Real: y(n), x2(n), z(n), w(n).

Integer: l(n).

Set: k1=arbitrary, k1r=0, k1s=0, iter=0.

Read (y(i), x2(i), i=1,n)

Step 1) Compute weights and ratios.

Do loop for i=1,k1-1: w(i)=x2(i)-x2(k1), z(i)=(y(i)-y(k1))/w(i), end do.

Set: w(k1)=0, z(k1)=0.

Do loop for i=k1+1,n: w(i)=x2(i)-x2(k1), z(i)=(y(i)-y(k1))/w(i) end do.

Set: iter=iter+1.

Step 2) Compute weighted median.

Let: lm=LWMED(n,z,w,l).

Step 3) Check for optimality.

Set: k1s=k1r, k1r=k1.

If lm=k1s then go to step 4, otherwise k1=lm.

Go to step 1.

Step 4) Compute the solution.

Let b2=z(lm), b1=y(k1)-b2*x2(k1).

Print b1, b2, k1, lm, iter.

Stop.

END

4. Generalization to m parameters

Now we extend the above procedure to the restricted two parameters model, namely,

$$y_i = \beta_2 x_{2i} + \beta_3 x_{3i} + u_i \quad (37)$$

Let,

$$S = \sum_{i=1}^n |y_i - \beta_2 x_{2i} - \beta_3 x_{3i}| \quad (38)$$

S can be written as,

$$S = \sum_{i=1}^n |x_{2i}| |y_i/x_{2i} - \beta_2 - \beta_3 x_{3i}/x_{2i}| = \sum_{i=1}^n |x_{2i}| |y_i^{s1} - \beta_2 - \beta_3 x_{3i}^{s1}| \quad (39)$$

where,

$$y_i^{s1} = y_i/x_{2i}, \quad x_{3i}^{s1} = x_{3i}/x_{2i} \quad i=1, \dots, n \quad (40)$$

Minimization of (39) is similar to that of simple linear model explained in the previous section. An important distinction for solving (39) in comparison to (17) is the expression $|x_{2i}|$ which has been multiplied to $|y_i^{s1} - \beta_2 - \beta_3 x_{3i}^{s1}|$. This multiplication does not make any problem when (39) is minimized, because if we deviate y_i^{s1} and x_{3i}^{s1} from the point k1 as,

$$y_i^{s1k1} = y_i^{s1} - y_{k1}^{s1}, \quad x_{3i}^{s1k1} = x_{3i}^{s1} - x_{3k1}^{s1} \quad i=1, \dots, n \quad (41)$$

then we can rewrite (39) similar to (21), so,

$$S_{k1} = \sum_{i=1}^n |x_{2i}| |y_i^{s1k1} - \beta_3 x_{3i}^{s1k1}| \quad (42)$$

To minimize (42), we should use the following expression,

$$S_{k1} = \sum_{i=1}^n |x_{2i} x_{3i}^{s1k1}| |y_i^{s1k1}/x_{3i}^{s1k1} - \beta_3| \quad (43)$$

According to Bidabad (1987a,88a), in applying discrete derivative to (43) the term in the first absolute value sign is used to find the subscript of that point which locates on the regression line. In comparison to the simple unrestricted linear model (16) this is the main difference.

In the case of inclusion of an intercept in the model given by (37), we have,

$$S = \sum_{i=1}^n |y_i - \beta_1 - \beta_2 x_{2i} - \beta_3 x_{3i}| \quad (44)$$

Let $k1$ be an arbitrary subscript then, transference of the origin of coordinates to the $k1^{th}$ point is done by deviating all observations from this point, namely,

$$y_i^{k1} = y_i - y_{k1}, \quad x_{2i}^{k1} = x_{2i} - x_{2k1}, \quad x_{3i}^{k1} = x_{3i} - x_{3k1} \quad i=1, \dots, n \quad (45)$$

Rearrange the terms of (45) and substitute in (44), then we have,

$$S = \sum_{i=1}^n |y_i^{k1} - \beta_2 x_{2i}^{k1} - \beta_3 x_{3i}^{k1} + (y_{k1} - \beta_1 - \beta_2 x_{2k1} - \beta_3 x_{3k1})| \quad (46)$$

If $k1^{th}$ observation is on the regression plane, then

$$y_{k1} - \beta_1 - \beta_2 x_{2k1} - \beta_3 x_{3k1} = 0 \quad (47)$$

So, instead of minimizing (44), the following function is minimized:

$$S_{k1} = \sum_{i=1}^n |y_i^{k1} - \beta_2 x_{2i}^{k1} - \beta_3 x_{3i}^{k1}| \quad (48)$$

Minimization of (48) is completely similar to that of (38) and can be proceeded as follows,

$$S_{k1} = \sum_{i=1}^n |x_{2i}^{k1}| |y_i^{k1}/x_{2i}^{k1} - \beta_2 - \beta_3 x_{3i}^{k1}/x_{2i}^{k1}| = \sum_{i=1}^n |x_{2i}^{k1}| |y_i^{s1} - \beta_2 - \beta_3 x_{3i}^{s1}| \quad (49)$$

where,

$$y_i^{s1} = y_i^{k1}/x_{2i}^{k1}, \quad x_{3i}^{s1} = x_{3i}^{k1}/x_{2i}^{k1} \quad i=1, \dots, n \quad (50)$$

Now, again, transfer the origin of the two dimensional YxX_3^{s1} coordinates to an arbitrary point $k2$ by deviating y_i^{s1} and x_{3i}^{s1} from y_{k2}^{s1} and x_{3k2}^{s1} as follows,

$$y_i^{s1k2} = y_i^{s1} - y_{k2}^{s1}, \quad x_{3i}^{s1k2} = x_{3i}^{s1} - x_{3k2}^{s1} \quad i=1, \dots, n \quad (51)$$

By rearranging the terms of (51) and substituting them into (49) and assuming that the point $k2$ is on the regression plane, we can rewrite (49) as,

$$S_{k1k2} = \sum_{i=1}^n |x_{2i}^{k1}| |y_i^{s1k2} - \beta_3 x_{3i}^{s1k2}| \quad (52)$$

or,

$$S_{k1k2} = \sum_{i=1}^n |x_{2i}^{k1} x_{3i}^{s1k2}| |y_i^{s1k2}/x_{3i}^{s1k2} - \beta_3| \quad (53)$$

The objective function (53) can be minimized as suggested before. Now, the procedure from (49) to (53) can be repeated with different values of $k2$ as in algorithm 2 proposed for the simple linear model. When the last point (M) in the process of minimizing (53) is reached, the origin of three dimensional YxX_2xX_3 coordinates ($k1$) is transferred to this newly found point M and the procedure from (45) to (53) is again repeated with the exception that instead of assigning an arbitrary value to $k2$, we set $k2$ equal to the previous value of $k1$. This procedure continues until the point found by minimizing (53) is equal to previous value of $k1$. The values of β_1^{\wedge} , β_2^{\wedge} and β_3^{\wedge} can be computed according to the following formulas,

$$\beta_3^{\wedge} = y_M^{s1k2}/x_{3M}^{s1k2}, \quad \beta_2^{\wedge} = y_{k2}^{s1} - \beta_3^{\wedge} x_{3k2}^{s1}, \quad \beta_1^{\wedge} = y_{k1} - \beta_2^{\wedge} x_{2k1} - \beta_3^{\wedge} x_{3k1} \quad (54)$$

It should be noted that at each step it can be proved that we are getting closer to the minimum of S in (44). The proof is similar to that of the two parameters model given by (16).

It can be shown that for any two arbitrary points k_1 and k_2 , value of $S_{k_1 k_2}$ given by (53) for any commutation for k_1 and k_2 remains unchanged. This will be shown in the next sections. However,

$$S_{k_1 k_2} = S_{k_2 k_1} \quad (55)$$

Now, following the procedure of the proof given by (28) through (36) we can write,

$$S_{k_1 a}^* = S_{k_1 b} \Big|_{\beta_3 = \beta_3^{k_1 a}} \quad (56)$$

where $S_{k_1 a}^*$ denotes the optimal value of $S_{k_1 a}$ with respect to β_3 of (53) for any arbitrary integer a from one to n , namely,

$$S_{k_1 a}^* = \min_{\beta_3} (S_{k_1 a}) \quad (57)$$

Subscript b in (56) denotes the observation which is found by minimizing $S_{k_1 a}$. Since the a^{th} and b^{th} observations both are on the regression hyperplane, by a similar discussion stated by (28) through (32) we can conclude that $S_{k_1 a}^*$ is equal to $S_{k_1 b}$ which evaluated at that value of β_3 which is found by minimizing $S_{k_1 a}$. Since the left hand side of (56) is minimum and its right hand side can be decreased by other values of β_3 , it can be concluded that,

$$S_{k_1 a}^* > S_{k_1 b}^* \quad (58)$$

According to (55) we can write,

$$S_{k_1 a}^* > S_{k_1 b}^* = S_{b k_1}^* \quad (59)$$

Since in each step we discard an observation from the basis and replace it with the newly found one, we have,

$$S_{k_1 a}^* > S_{k_1 b}^* = S_{b k_1}^* > S_{b c}^* = S_{c d}^* \quad (60)$$

or generally,

$$S_{k_1 a}^* > S_{k_1 b}^* > S_{b c}^* > S_{c d}^* \quad (61)$$

The minimum solution S^* is reached when by entering the f^{th} observation in the basis we can not reduce the objective function value. That is the previous observation is again reached, namely,

$$S_{k_1 a}^* > S_{a b}^* > S_{b c}^* > S_{c d}^* > \dots > S_{f g}^* = S_{g f}^* = S^* \quad (62)$$

The relation (62) guarantees that at each step we are descending down the objective function surface.

4.1 Algorithm 3

To generalize the above algorithm to the m parameters linear model (1), we should reduce the number of parameters in the same fashion as in the case of three parameters model explained above. If the model is restricted, we can make it unrestricted by dividing all dependent and independent variables to one of independent variables as follows.

$$S = \sum_{i=1}^n \left| y_i - \sum_{j=2}^m \beta_j x_{ji} \right| = \sum_{i=1}^n \left| x_{2i} \left| y_i/x_{2i} - \beta_2 - \sum_{j=3}^m \beta_j x_{ji} \right| \right| \quad (63)$$

If the model is unrestricted, we can make it restricted by deviating all observations from an arbitrary one observation; namely,

$$S = \sum_{i=1}^n \left| y_i - \beta_1 - \sum_{j=2}^m \beta_j x_{ji} \right| = \sum_{i=1}^n \left| y_i^{k1} - \sum_{j=2}^m \beta_j x_{ji}^{k1} \right| \quad (64)$$

where,

$$y_i^{k1} = y_i - y^{k1}, \quad x_{ji}^{k1} = x_{ji} - x_{jk1} \quad i=1, \dots, n; j=2, \dots, m \quad (65)$$

Therefore, according to the transformations (63), (64) and (65) any m parameters model can be reduced to a simple restricted one parameter model and then be estimated as a weighted median problem. To conduct this transformation, if the model is unrestricted, we should deviate all observations from an arbitrary one and make the model restricted. Then

divide all dependent and independent variables to an arbitrary independent variable. This makes the model unrestricted. At this step we have reduced the number of parameters by one. By continuing this process, we can reduce any m parameters model to a one parameter model. If the model is restricted we should start by dividing all of the variables by one of the independent variables and make the model unrestricted. Now, we can again reduce one of the parameters by applying the procedures explained above in order to transform unrestricted model to restricted form. By solving the one parameter model, we can then solve for the two parameters and then three parameters models and so on.

The most delicate part of this algorithm is that, at the starting point, once $k_1, k_2, \dots, k_{(m-1)}$ are selected arbitrarily, then the algorithm assigns the best possible values to the integers $k_1, k_2, \dots, k_{(m-1)}$. To explain the procedure, let us deal with the four parameters unrestricted linear model. Once, value of k_1 is arbitrarily selected. Deviate all observations from k_1^{th} observation. In this way, the model reduces to a three parameters restricted model. By dividing all of the variables to one of the independent variables, our model becomes completely similar to (44). By minimizing (44) according to the algorithm previously explained for the three parameters model, subscript M corresponding to M^{th} point is derived. This is the newly found point which its subscript (M) is assigned to k_1 . The previous value of k_1 is assigned to k_2 and the previous value of k_2 is assigned to k_3 and the whole procedure is repeated again. The procedure stops when the value of M is equal to k_1 .

The above important assigning technique which is essential for pivoting on the origins of different size coordinates can be extended to more parameters models as we did before. Again, we should note that at each succeeding step we get closer to the minimum of S in (1). The proof is completely similar to the three parameters model explained before. The whole procedure is as follows.

Algorithm 3

- Step 0) Select arbitrary points $k_1, k_2, \dots, k_{(m-1)}$.
- Step 1) Set counter=1.
- Step 2) Proceed the transformations (63) through (65) to reduce model to a one parameter transformed restricted linear model.
- Step 3) Apply weighted median procedure to find the point M .
- Step 4) If M is equal to previous value of $k(\text{counter})$ go to step 5; otherwise set $k(\text{counter})=M$ and go to step 1.
- Step 5) If counter = $m-1$ go to step 6; otherwise let counter=counter+1 and go to step 4.
- Step 6) Set $k(m)=M$. By using the final values of k_1, \dots, k_m solve the following system of equations for β_j^{\wedge} , for $j=1, \dots, m$;

$$\sum_{j=1}^m \beta_j^{\wedge} x_{jki} = y_{ki} \quad i=1, \dots, m$$
 Stop.

4.2 Algorithm 4

In algorithm 3 to solve any m parameters model, the reduced $m-1$ parameters model should be solved first and therefore makes the algorithm rather costly. Because to solve a m parameters model we should reduce the objective function value of the reduced $m-1$ parameters model down to its minimum value and then going back to the m parameters model. In interior steps for $m-1, m-2, \dots, 1$ this procedure repeats. In order to make this algorithm more efficient another assigning technique may be adopted.

Once $m-1$ observations are selected arbitrarily. The S function is reduced to a one parameter weighted median problem as we did in algorithm 3 using (63) and (64). By solving

the weighted median problem a new observation M is found. Now, the newly found point is replaced by the most previous observation which entered the basis. The procedure is repeated until we can not find any new observation out of the current set of basis points and the deleted point in the previous iteration. On the other hand, in this algorithm m-1 point is selected as a basic feasible solution. By pivoting on this solution a new point is found to enter the basis and should be replaced by another one which was previously in the basis. The procedure ends when we can not enter any new point to reduce the objective function value. In this context it is similar to simplex procedure of linear programming technique.

Algorithm 4 is also a descent method. Because in each iteration the value of S is decreased. Proof is completely similar to that of algorithm 3 stated before. The main difference of this algorithm with algorithm 3 is in choosing the point which should be deleted from the basis. Therefore, in each step we need not to keep some points in the basis to solve smaller size models. On the other hand, in each m iterations all m points in the basis are discarded and new points enter. In algorithm 3 we kept some points in the basis until we reach an optimal point for smaller size reduced model. It should be noted that one point may be discarded and reenter the basis through m iterations. This is not contradictory and actually occurs especially when we are near the optimal solution.

As we will see in the forthcoming section, due to different properties of this algorithm many manipulations in computational steps may be adopted to start calculations. Before making this algorithm more sophisticated it is suitable to summarize its general steps as follows.

Algorithm 4

- Step 0) Select arbitrary points k1,...,k(m-1).
 - Step 1) Proceed the transformations (63) through (65) to reduce model to one parameter transformed restricted linear model.
 - Step 2) Apply weighted median procedure to find the point M.
 - Step 3) Assign M to one of the k1,...,k(m-1) which is oldest in the basis.
 - Step 4) Check if the newly found point M which enters the basis, discard the point which entered in previous iteration repeats m-1 times then go to step 5; otherwise go to step1.
 - Step 5) Set k(m)=M. By using the final values of k1,...,k(m), solve the following system of equations for β_j^* , for $j=1,...,m$;
- $$\sum_{j=1}^m \beta_j^* x_{jki} = y_{ki} \quad i=1,...,m$$
- Stop.

After discussing the properties of algorithm 4, we will give more expository steps of computation for this algorithm. By using these properties we can make this algorithm more efficient.

4.3 properties

Before discussing the properties of the proposed algorithms specifically algorithm 4, let us consider the following four parameters model L_1 norm objective function,

$$S = \sum_{i=1}^n |y_i - \beta_0 - \beta_1 x_{1i} - \beta_2 x_{2i} - \beta_3 x_{3i}| \quad (66)$$

Some steps should be taken to reduce (66) to a weighted median problem by entering three arbitrary integers k1, k2, k3 between one and n as follows. Note that function (1) has been

slightly redefined to include intercept as a simple term, in equation (66). Similar to the transformations (63) through (65) the following steps are taken sequentially.

Deviate all observations from $k1^{\text{th}}$ observation,

$$y_i^{k1} = y_i - y_{k1} \quad i=1, \dots, n \quad (67)$$

$$x_{ji}^{k1} = x_{ji} - x_{jk1} \quad i=1, \dots, n; j=1, 2, 3 \quad (68)$$

The objective function becomes,

$$S_{k1} = \sum_{i=1}^n |y_i^{k1} - \beta_1 - \beta_2 x_{2i}^{k1} - \beta_3 x_{3i}^{k1}| \quad (69)$$

Divide y_i^{k1} and x_{ji}^{k1} for $j=2, 3$ by x_{1i}^{k1} for all $i=1, \dots, n$;

$$y_i^{s1} = y_i^{k1}/x_{1i}^{k1} \quad i=1, \dots, n \quad (70)$$

$$x_{ji}^{s1} = x_{ji}^{k1}/x_{1i}^{k1} \quad i=1, \dots, n; j=2, 3 \quad (71)$$

The objective function becomes,

$$S_{k1} = \sum_{i=1}^n |x_{1i}^{k1}| |y_i^{s1} - \beta_1 - \beta_2 x_{2i}^{s1} - \beta_3 x_{3i}^{s1}| \quad (72)$$

Deviate y_i and x_{ji}^{s1} for $j=2, 3$ from $k2^{\text{th}}$ observation,

$$y_i^{k2} = y_i^{s1} - y_{k2}^{s1} \quad i=1, \dots, n \quad (73)$$

$$x_{ji}^{k2} = x_{ji}^{s1} - x_{jk2}^{s1} \quad i=1, \dots, n; j=2, 3 \quad (74)$$

The objective function reduces to,

$$S_{k1k2} = \sum_{i=1}^n |x_{1i}^{k1}| |y_i^{k2} - \beta_1 - \beta_2 x_{2i}^{k2} - \beta_3 x_{3i}^{k2}| \quad (75)$$

Divide y_i^{k2} and x_{ji}^{k2} for $j=3$ by x_{2i}^{k2} for all $i=1, \dots, n$;

$$y_i^{s2} = y_i^{k2}/x_{2i}^{k2} \quad i=1, \dots, n \quad (76)$$

$$x_{ji}^{s2} = x_{ji}^{k2}/x_{2i}^{k2} \quad i=1, \dots, n; j=3 \quad (77)$$

The objective function forms as,

$$S_{k1k2} = \sum_{i=1}^n |x_{1i}^{k1} x_{2i}^{k2}| |y_i^{s2} - \beta_1 - \beta_2 x_{3i}^{s2}| \quad (78)$$

Deviate y_i^{s2} and x_{ji}^{s2} from $k3^{\text{th}}$ observation,

$$y_i^{k3} = y_i^{s2} - y_{k3}^{s2} \quad i=1, \dots, n \quad (79)$$

$$x_{ji}^{k3} = x_{ji}^{s2} - x_{jk3}^{s2} \quad i=1, \dots, n; j=3 \quad (80)$$

Rewrite the objective function as,

$$S_{k1k2} = \sum_{i=1}^n |x_{1i}^{k1} x_{2i}^{k2} x_{3i}^{k3}| |y_i^{s3} - \beta_1 - \beta_2 x_{3i}^{s3}| \quad (81)$$

Divide y_i^{k3} by x_{3i}^{k3} for all $i=1, \dots, n$;

$$y_i^{s3} = y_i^{k3}/x_{3i}^{k3} \quad i=1, \dots, n \quad (82)$$

Finally the objective function becomes,

$$S_{k1k2k3} = \sum_{i=1}^n |x_{1i}^{k1} x_{2i}^{k2} x_{3i}^{k3}| |y_i^{s3} - \beta_1 - \beta_2 x_{3i}^{s3}| \quad (83)$$

Rewrite (83) as,

$$S_{k1k2k3} = \sum_{i=1}^n |w_i^{k1k2k3}| |r_i^{s1s2s3} - \beta_3| \quad (84)$$

where,

$$w_i^{k1k2k3} = x_{1i}^{k1} x_{2i}^{k2} x_{3i}^{k3} \quad (85)$$

$$r_i^{s1s2s3} = y_i^{s3} \quad (86)$$

The objective function (84) is a weighted median problem which may easily be minimized to find β_3 . Equation (84) for the general m parameters model

$$y_i = \sum_{j=0}^{m-1} x_{ji} \beta_j + u_i \quad i=1, \dots, n \quad (87)$$

$$x_{0i}=1 \quad i=1, \dots, n \quad (88)$$

with the corresponding objective function,

$$S = \sum_{i=1}^n \left| y_i - \sum_{j=0}^{m-1} x_{ji} \beta_j \right| \quad (89)$$

is,

$$S^{k1k2\dots k(m-1)} = \sum_{i=1}^n \left| w_i^{k1k2\dots k(m-1)} \right| \left| r_i^{s1s2\dots s(m-1)} - \beta_{m-1} \right| \quad (90)$$

where,

$$w_i^{k1k2\dots k(m-1)} = \left| x_{1i}^{k1} x_{2i}^{k2} \dots x_{(m-1)i}^{k(m-1)} \right| \quad (91)$$

$$r_i^{s1s2\dots s(m-1)} = y_i^{s(m-1)} \quad (92)$$

Now, we proceed to explain the properties of algorithm 4 by using the formulation described above.

Property 1

If we start the computation with $m-1$ points which do not belong to the original set of observations, value of S decreases to its global minimum at each iteration. Suppose the points $(y_{n+1}, \dots, x_{m-1,n+1}), \dots, (y_{n+m-1}, \dots, x_{m-1,n+m-1})$ do not belong to our sample points. If $n+1, n+2, \dots, n+m-1$ are assigned to $k1, k2, \dots, k(m-1)$, then minimizing the function in (90) leads to finding a new point M which belongs to the sample observations. Replacing $k1$ by M and deleting the $(n+1)^{th}$ point from the basis and minimizing (90) leads to finding another point M' in the sample which will be replaced by the $k2^{th}$ observation which is out of our sample. Thus, in each iteration one of the out of sample points is deleted and optimization procedure will be done on the sample points after $m-1$ iterations.

This property has a good performance on ill-conditioned data. That is if for example rounding error causes an observation to get incorrect rounded value, then this point in the next iteration will be replaced by a sample point and thus redirect the path of descending to the right points.

Property 2

Reduction of $m-1$ parameters model to fewer parameters model can be done by using the computation of $m-1$ parameters model and conversely. Thus stepwise and all possible regressions may be computed efficiently.

To show this property let us deal with equation (86). Value of r_i^{s1s2s3} may be computed by substituting (67), (68), (70), (71), (73), (74), (76), (77), (79), (80) and (82) into (86). The result is,

$$r_i^{s1s2s3} = \frac{\begin{vmatrix} y_i - y_{k1} & y_{k2} - y_{k1} \\ x_{1i} - x_{1k1} & x_{1k2} - x_{1k1} \end{vmatrix} - \begin{vmatrix} y_{k3} - y_{k1} & y_{k2} - y_{k1} \\ x_{1k3} - x_{1k1} & x_{1k2} - x_{1k1} \end{vmatrix}}{\begin{vmatrix} x_{2i} - x_{2k1} & x_{2k2} - x_{2k1} \\ x_{1i} - x_{1k1} & x_{1k2} - x_{1k1} \end{vmatrix} - \begin{vmatrix} x_{2k3} - x_{2k1} & x_{2k2} - x_{2k1} \\ x_{1k3} - x_{1k1} & x_{1k2} - x_{1k1} \end{vmatrix}} - \frac{\begin{vmatrix} x_{3i} - x_{3k1} & x_{3k2} - x_{3k1} \\ x_{1i} - x_{1k1} & x_{1k2} - x_{1k1} \end{vmatrix} - \begin{vmatrix} x_{3k3} - x_{3k1} & x_{3k2} - x_{3k1} \\ x_{1k3} - x_{1k1} & x_{1k2} - x_{1k1} \end{vmatrix}}{\begin{vmatrix} x_{2i} - x_{2k1} & x_{2k2} - x_{2k1} \\ x_{1i} - x_{1k1} & x_{1k2} - x_{1k1} \end{vmatrix} - \begin{vmatrix} x_{2k3} - x_{2k1} & x_{2k2} - x_{2k1} \\ x_{1k3} - x_{1k1} & x_{1k2} - x_{1k1} \end{vmatrix}} \quad (93)$$

By inspection, it is obvious that the contents of rectangle grid lines in (93) are r_i^{s1} , r_i^{s1s2} and r_i^{s1s2s3} . Generally, $r_i^{s1s2...s(m-1)}$ has the following scheme,

$$r_1^{s1...s(m-1)} = \begin{array}{c} r_i^{s1} \\ r_i^{s1s2} \\ \ddots \\ r_i^{s1s2...s(m-1)} \end{array} \quad (94)$$

A similar discussion may also be performed for $w_i^{k1k2...k(m-1)}$. Rewrite (91) by substitution of (68), (74) and (80) in (91),

$$w_i^{k1k2k3} = \left| \left(x_{1i} - x_{1k1} \right) \left(\frac{x_{2i} - x_{2k1}}{x_{1i} - x_{1k1}} - \frac{x_{2k2} - x_{2k1}}{x_{1k2} - x_{1k1}} \right) \left(\begin{array}{cc} \frac{x_{3i} - x_{3k1}}{x_{1i} - x_{1k1}} - \frac{x_{3k2} - x_{3k1}}{x_{1k2} - x_{1k1}} & \frac{x_{3k3} - x_{3k1}}{x_{1k3} - x_{1k1}} - \frac{x_{3k2} - x_{3k1}}{x_{1k2} - x_{1k1}} \\ \frac{x_{1i} - x_{1k1}}{x_{1k3} - x_{1k1}} - \frac{x_{1k2} - x_{1k1}}{x_{1k2} - x_{1k1}} & \frac{x_{2i} - x_{2k1}}{x_{1k3} - x_{1k1}} - \frac{x_{2k2} - x_{2k1}}{x_{1k2} - x_{1k1}} \\ \frac{x_{2i} - x_{2k1}}{x_{1i} - x_{1k1}} - \frac{x_{2k2} - x_{2k1}}{x_{1k2} - x_{1k1}} & \frac{x_{2k3} - x_{2k1}}{x_{1k3} - x_{1k1}} - \frac{x_{2k2} - x_{2k1}}{x_{1k2} - x_{1k1}} \\ \frac{x_{1i} - x_{1k1}}{x_{1k3} - x_{1k1}} - \frac{x_{1k2} - x_{1k1}}{x_{1k2} - x_{1k1}} & \frac{x_{1k3} - x_{1k1}}{x_{1k2} - x_{1k1}} - \frac{x_{1k2} - x_{1k1}}{x_{1k2} - x_{1k1}} \end{array} \right) \right| \quad (95)$$

By inspection of (93), (94) and (95) it is obvious that calculation of $r_i^{s1s2...sh}$ and $w_i^{s1s2...sh}$ where $0 < h \leq m-1$ which are necessary for h parameters regression may be derived from the m parameters model steps of computation.

Property 3

By any commutation for kj for $j=1,...,m-1$, values of $r_i^{s1s2...s(m-1)}$ defined by (92) for all $i \neq k1, k2, ..., k(m-1)$ remain unchanged. This calculation can be made by deleting the suitable terms in (93) and (94) after taking common denominator. Because of this property, we can commute for the values of $k1, ..., k(m-1)$ in the process of computation without changing the value of $r_i^{s1...s(m-1)}$.

Property 4

By any commutation for kj for $j=1,...,m-1$, values of $w_i^{k1...k(m-1)}$ for all $i=1,...,n$ defined by (91) remain unchanged. This conclusion is made by multiplication of sequential parentheses of (97) and deleting the suitable terms. This property accompanying with property 3 guarantees that any commutation for $k1, ..., k(m-1)$ does not change the necessary items of (91) and (92) in computation of weighted median problem (90).

Property 5

Value of $r_{k(m-1)}^{s1...s(m-1)}$ is equal to zero. By substituting $i=k(m-1)$ in (93) or (94) this property is concluded. Since, this term is equal to zero we should be aware of divide check (dividing by zero) in the process of computation of $r_{k1}^{s1}, r_{k2}^{s1s2}, \dots, r_{k(m-1)}^{s1s2...s(m-1)}$.

Property 6

Values of $w_h^{k1k2...k(m-1)}=0$ for all $h=k1,k2,\dots,k(m-1)$. If h is equal to one of the $k1,k2,\dots,k(m-1)$, then one of the parentheses of (95) is equal to zero and this property is assured. According to this property to avoid divide check, without computing $w_h^{k1k2...k(m-1)}$ for $h=k1,k2,\dots,k(m-1)$ we can set these terms equal to zero.

Property 7

By any commutation for kj for $j=1,\dots,(m-1)$, the M^{th} point derived by minimizing (89) remains unchanged. Because, by this commutation values of $r_i^{s1s2...s(m-1)}$ and $w_i^{k1k2...k(m-1)}$ remain unchanged (properties 3,4,5 and 6).

Property 8

Updating procedure to evaluate different subsets of sample observations can be simply done by assigning zero values to their $w_h^{k1k2...k(m-1)}$ in (90) for deleting points. This makes those unwanted points h 's be unaffected in calculation.

Property 9

According to properties 6 and 8, since the value of $w_h^{k1k2...k(m-1)}=0$ for all $h=k1,k2,\dots,k(m-1)$; the point M which derives from solving weighted median problem (90) will not be equal to $k1,k2,\dots,k(m-1)$. This property guarantees that when we enter an observation into the basis we do not find this point again in the same iteration.

Property 10

If all kj for $j=1,\dots,m-1$ are equal in starting time, no problem occurs in the process of computation, but the execution time may be increased. Conversely if the kj points are selected in such a way that they are near the minimum of S , the algorithm converges to the optimal solution with less iterations.

Property 11

Values of $w_i^{k1k2...k(m-1)}$ and $r_i^{s1s2...s(m-1)}$ which are essential to solve the weighted median problem of (90) can be retrieved from their values in the previous iteration. That is when the point kh is discarded from the basis and replaced by kj , values of $w_i^{k1k2...k(h-1)}$ and $r_i^{s1s2...s(h-1)}$ do not change and need not be computed again. Applying this property to algorithm 4 makes it more efficient.

Property 12

Since at each iteration, values of $r_i^{s1s2...s(m-1)}$ and $w_i^{k1k2...k(m-1)}$ should be computed from the source input data y and X , rounding error does not accumulate and convergence does not perturb. This property makes the algorithm safe from accumulation of rounding error which exists in simplex-type algorithms.

Property 13

When m optimal points are reached, the optimal values of β_j^\wedge for $j=1,\dots,m$ can be computed as a recursive system of equations efficiently. Values of β_j^\wedge will be,

$$\begin{pmatrix} \beta_{m-1}^{\wedge} \\ \beta_{m-2}^{\wedge} \\ \vdots \\ \beta_2^{\wedge} \\ \beta_1^{\wedge} \\ \beta_0^{\wedge} \end{pmatrix} = \begin{pmatrix} y_M^{s(m-1)} \\ y_{k(m-1)}^{s(m-2)} \\ \vdots \\ y_{k3}^{s2} \\ y_{k2}^{s1} \\ y_{k1} \end{pmatrix} - \begin{pmatrix} 0 & 0 & \cdots & 0 & 0 & 0 \\ x_{(m-1)k(m-1)}^{s(m-2)} & 0 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ x_{(m-1)k3}^{s2} & x_{(m-2)k3}^{s2} & \cdots & 0 & 0 & 0 \\ x_{(m-1)k2}^{s1} & x_{(m-2)k2}^{s1} & \cdots & x_{2k2}^{s1} & 0 & 0 \\ x_{(m-1)k1} & x_{(m-2)k1} & \cdots & x_{2k1} & x_{1k1} & 0 \end{pmatrix} \begin{pmatrix} \beta_{m-1}^{\wedge} \\ \beta_{m-2}^{\wedge} \\ \vdots \\ \beta_2^{\wedge} \\ \beta_1^{\wedge} \\ \beta_0^{\wedge} \end{pmatrix} \quad (96)$$

The recursive system of equations (96) can be obtained as follows. Solve (67) and (68) for y_i and x_{ji} , $j=1,2,3$; and substitute them into (66), gives

$$S_{k1} = \sum_{i=1}^n |y_i^{k1} - \beta_1 x_{1i}^{k1} - \beta_2 x_{2i}^{k1} - \beta_3 x_{3i}^{k1} - (y_{k1} - \beta_0 - \beta_1 x_{2k1} - \beta_3 x_{3k1})| \quad (97)$$

Since we force the regression to pass through the point $k1$, the expression inside the parentheses of (97) is equal to zero. Hence,

$$y_{k1} - \beta_0 - \beta_1 x_{2k1} - \beta_3 x_{3k1} = 0 \quad (98)$$

We can solve (73) and (74) again for y_i^{s1} and x_{ji}^{s1} , where, $j=2,3$ and substitute them into (72) which leads to the following objective function,

$$S_{k1k2} = \sum_{i=1}^n |x_{1i}^{k1}| |y_i^{k2} - \beta_2 x_{2i}^{k2} - \beta_3 x_{3i}^{k2} - (y_{k2}^{s1} - \beta_1 - \beta_2 x_{2k2}^{s1} - \beta_3 x_{3k2}^{s1})| \quad (99)$$

By a similar discussion on (97) we have,

$$y_{k2}^{s1} - \beta_1 - \beta_2 x_{2k2}^{s1} - \beta_3 x_{3k2}^{s1} = 0 \quad (100)$$

Similarly we can derive,

$$S_{k1k2k3} = \sum_{i=1}^n |x_{1i}^{k1} x_{2i}^{k2}| |y_i^{k3} - \beta_3 x_{3i}^{k3} + (y_{k3}^{s2} - \beta_2 - \beta_3 x_{3k3}^{s2})| \quad (101)$$

and based on (101),

$$y_{k3}^{s2} - \beta_2 - \beta_3 x_{3k3}^{s2} = 0 \quad (102)$$

Finally, by solving weighted median problem (83) we find a point M which gives,

$$\beta_3 = y_M^{s3} \quad (103)$$

Equations (103), (102), (100) and (98) can be solved sequentially for $\beta_3^{\wedge}, \dots, \beta_0^{\wedge}$. In a similar fashion the general solution for m parameters model is given by (96).

Now let us use these properties and give the computational steps of algorithm 4 more expository in the following BL1 program.

PROGRAM BL1

Step 0) Initialization.

Parameter: n, m, m1=m-1, m2=m-2.

Real: y(n), x(n,m1), xsk(m1), yw(n), xkw(n), w(n), ys(n), xs(n,m1), b(m), xw(n,2:m1), ysol(m1), xsol(m1,m1).

Integer: l(n), kk(m1).

Common: /c1/i1,i2.

Read: (y(i),(x(i,j),j=1,m1),i=1,n).

Let: iter=0, kr=0, mm=1, (kk(j)=arbitrary,j=1,m1).

Step 1) Refill working arrays.

Do loop for i=1,n: ys(i)=y(i), do loop for j=1,m1: xs(i,j)=x(i,j), end do, end do.

Step 2) Store weights and ratios for next iteration.

Do loop for $i=1,n$: $w(i)=xkw(i)$, $ys(i)=yw(i)$, do loop for $j=1,m1$: $xs(i,j)=xw(i,j)$, end do,
end do.

Step 3) Compute the arguments for weighted median.

a. Set: $jj=mm$, $k=kk(jj)$, $ysk=ys(k)$, $i1=1$, $i2=k-1$.

Do loop for $j=jj,m1$: $xsk(j)=xs(k,j)$, end do.

b. Do loop for $j=jj,m1$: call COL1($xsk(j),xs(1,j)$) end do.

Call COL2($ysk,jj,w,ys,xs(1,jj)$).

If $i2=n$ go to step 3.c; otherwise set: $i1=k+1$, $i2=n$, go to step 3.b.

c. Set: $w(k)=0$.

If $jj=m1$ go to step 4; otherwise $i1=1$, $i2=k-1$, go to step 3.d.

d. Do loop for $j=jj+1,m1$: call COL3($xs(1,j),xs(1,jj)$), end do.

If $i=n$ go to step 3.e; otherwise $i1=k+1$, $i2=n$, go to step 3.d.

e. If $jj\neq mm$ $jj=jj+1$, go to step 3, otherwise do loop for $i=1,n$: $xkw(i)=w(i)$, $yw(i)=ys(i)$; do
loop for $j=jj+1,m1$: $xw(i,j)=xs(i,j)$, end do; end do.

Set: $jj=jj+1$, go to step 3.

Step 4) Compute the weighted median.

Set: $ys(k)=0$, $iter=iter+1$, $lm=LWMED(n,ys,w,l)$.

Step 5) Test for optimality.

If $lm=kr$ go to step 5.b; otherwise $iopt=0$ go to step 5.a.

a. If $mm=m1$ set $mm=1$, $kr=kk(mm)$, $kk(mm)=lm$, go to step 1; otherwise
set $mm=mm+1$, $kr=kk(mm)$, $kk(mm)=lm$, go to step 2.

b. Set: $iopt=iopt+1$.

If $iopt\neq m1$ go to step 5.a, otherwise go to step 6.

Step 6) Compute the solution.

Set: $b(m)=ys(lm)$.

Do loop for $i=1,m1$: $ysol(i)=y(kk(i))$; do loop for $j=1,m1$: $xsol(i,j)=x(kk(i),j)$, end do;
end do.

Set: $jj=1$.

a. Set: $ysk=ysol(jj)$.

Do loop for $j=jj,m1$: $xsx(j)=xsol(jj,j)$, end do.

Do loop for $i=jj,m1$: if $i=jj$ go to continue; otherwise

$ysol(i)=ysol(i)-ysk$, do loop for $j=jj,m1$: $xsol(i,j)=xsol(i,j)-xsk(j)$, end do;
set $ysol(i)=ysol(i)/xsol(i,jj)$, continue, end do.

b. Do loop for $i=jj,m1$: if $i=jj$ go to continue, otherwise, do loop for
 $j=jj+1,m1$: $xsol(i,j)=xsol(i,j)/xsol(i,jj)$, end do; continue; end do.

c. If $jj=m2$ go to step 6.d; otherwise go to step 6.a.

d. Do loop for $i=1,m2$: $k=m-i$, $s=ysol(k)$; do loop for $j=k,m1$,
 $s=s-b(j+1)*xsol(k,j)$ end do, $b(k)=s$, end do.

Set: $s=y(kk(1))$.

Do loop for $j=1,m1$: $s=s-b(j+1)*x(kk(1),j)$, $b(1)=s$, end do.

Print: $((b(j),j=1,m),(kk(j),j=1,m1),lm,iter)$.

Stop.

END

The major portion of computation in this program is transformation of two dimensional arrays. Passing columns of these arrays to other subroutines which involves only one dimensional arrays saves the time of computation (see, Barrodale and Roberts (1974)). Subroutine COL1, COL2 and COL3 have been coded to do this task for subtraction, multiplication and division, and for only division respectively. Function LWMED which is used to compute weighted median has been introduced in section 2.1.

SUBROUTINE COL1(v1,v2)

Step 0) Initialization

Real: v2(1).

Common /c1/i1,i2.

Step 1) Subtraction.

Do loop for i=i1,i2: v2(i)=v2(i)-v1, end do.

Return.

END

SUBROUTINE COL2(ysk,jj,v1,ys,v2)

Step 0) Initialization.

Real: v1(1),v2(1),ys(1).

Common /c1/i1,i2.

Step 1) Compute weights and ratios.

If jj≠1 go to step 1.a.; otherwise do loop for i=i1,i2:

v1(i)=v2(i), ys(i)=(ys(i)-ysk)/v2(i).

Return.

a. Do loop for i=i1,i2: v1(i)=v1(i)*v2(i), ys(i)=(ys(i)-ysk)/v2(i),
end do.

Return.

END

SUBROUTINE COL3(v1,v2)

Step 0) Initialization.

Real: v1(1),v2(1),ys(1).

Common /c1/i1,i2.

Step 1) Division.

Do loop for i=i1,i2: v1(i)=v1(i)/v2(i), end do.

Return.

END

4.4 Initial value problem

Selection of the arbitrary points k_1, k_2, \dots, k_{m-1} at the first stage of computation has important role in number of iterations necessary to reach the optimal solution. One way to select these $m-1$ points may be based on applying another estimator to guess those points which their residuals are smallest in absolute values among all estimated residuals. That is one estimator for example least squares is applied on the data and smallest $m-1$ residuals in absolute values are selected and the corresponding points to these residuals are nominated for starting algorithms 1, 2, 3 or 4. However, we examined this procedure for algorithms 2 and 4 and found that it makes them more efficient.

In the process of this study we found some points which might be leading to develop the field of L_1 norm computation. In the proposed algorithm 4, we could not find a criterion to delete that observation from the current set of observations in the basis which reduces the objective function value more than other points. If anyone finds a criterion like the heuristic method of Bloomfield and Steiger (1980), number of iterations will decrease very much and will reduce computation time.

The main obstacle to apply discrete differentiation technique to operational problems, is reordering of the residuals in descending order before starting computation. If this obstacle is removed, a similar algorithm to that of simple restricted linear regression for multiple regression may be proposed.

5. References

- I. Barrodale, F.D.K. Roberts (1974) Algorithm 478: Solution of an overdetermined system of equations in the L_1 norm. Comm. ACM, 17, 319-320.
- C.M. Bender, S.A. Orszag (1978) Advanced mathematical methods for scientists and engineers, McGraw-Hill.
- B. Bidabad (1987a) Least absolute error estimation. First International Conference on Statistical Data Analysis Based on the L_1 norm and Related Methods, Neuchatel, Switzerland.
- B. Bidabad (1987b) Least absolute error estimation, part II. First International Conference on Statistical Data Analysis Based on the L_1 norm and Related Methods, Neuchatel, Switzerland.
- B. Bidabad (1988a) A proposed algorithm for least absolute error estimation. Proceedings of the Third Seminar of Mathematical Analysis. Shiraz University, 24-34, Shiraz, Iran.
- B. Bidabad (1988b) A proposed algorithm for least absolute error estimation, part II. Proceedings of the Third Seminar of Mathematical Analysis, Shiraz University, 35-50, Shiraz, Iran.
- B. Bidabad (1989a) Discrete and continuous L_1 norm regressions, proposition of discrete approximation algorithms and continuous smoothing of concentration surface, Ph.D. thesis, Islamic Azad University, Tehran, Iran.
- B. Bidabad (1989b) Discrete and continuous L_1 norm regressions, proposition of discrete approximation algorithms and continuous smoothing of concentration surface, Ph.D. thesis, Islamic Azad University, Tehran, Iran. Farsi translation.
- P. Bloomfield, W. Steiger (1980) Least absolute deviations curve fitting. SIAM J. Sci. Stat. Comput. 1, 290-301.
- R.J. Boscovich (1757) De litteraria expeditione per pontificiam ditionem, et synopsis amplioris operis..., 'Bononiensi' scientiarum et artum instituto atque academia commetarii, vol.4, 353-396. Reprinted with a Serbo-Croatian translation by N. Cubranic, Institute of higher geodesy, University of Zagreb 1961.
- J. Chambers (1971) Algorithm 410: partial sorting. Comm. ACM, 14, 357-358.
- F.H. Clarke (1983) Optimization and nonsmooth analysis. Wiley, New York C.A.R. Hoare (1961) Algorithm 63 partition; 64, quicksort; and 65, find., Comm. ACM, 4, July, 321-322.
- C.A.R. Hoare (1962) Quicksort. Comput. J., 5, 10-15.
- O.J. Karst (1958) Linear curve fitting using least deviations. JASA, 53, 118-132.
- P.S. Laplace (1812) Theorie analytique des probabilites, Mme courrier Paris 1820 Reprinted in his oeuvres, vol.7, Imprimerie Royale, Paris, 1847, and Gauthier-Villars et fils, Paris 1886.
- P.S. Laplace (1818) Duexieme supplement to Laplace (1812).
- S. Scowen (1965) Algorithm 271 Quicksort. Comm. ACM, 8, 669-670.
- R.S. Singleton (1969) Algorithm 347 Sort. Comm. ACM, 12, 185-186.
- L.D. Taylor (1974) Estimating by minimizing the sum of absolute errors. In P. Zarembka (ed.) Frontiers in econometrics. Academic Press.