A proposed algorithm for Least Absolute Error estimation

by

Bijan BIDABAD

Regional Planning Bureau, Plan & Budget Ministry, Tehran, Iran.

Islamic Azad University, Tehran, Iran.

Abstract: This note proposes a method to find the Least Absolute Error estimate of b in the model $y_i=bX_i+u_i$. The proposed method is an alternative to linear programming algorithm. It uses a technique of discrete differentiation over subscripts. Unlike the estimation method based on linear programming, the proposed procedure can be simply handled.

Keywords: Least Absolute Error, Estimation, Linear Programming.

1. Introduction

Historically, The use of Least Absolute Error(LAE)by Edgeworth goes back to 1888. Since then, though attempts are made to overcome the computational difficulties there still remains the derivation of the sampling distributions of these estimators. In this note an algorithm is proposed to reduce the computatuinal difficulties.

This new procedure might also help the derivation of the sampling distribution for the simple models*. Consider the following regression model:

$$y_i = \sum_{j=1}^{m} b_j x_{ij} + u_i \qquad i=1,\ldots,n \qquad (1)$$

*The author is grateful to Dr.J.Mojarrad and Dr. A.Monajemi for valuable advice on various stages of this work.

*Rosenberg and Carlson [6].

where b_j are population parameters, y_i, x_{ij} and u_i are dependent, independent and random variables respectively. We wish to estimate b_js by minimizing the expression:

$$S = \sum_{i=1}^{n} |\hat{u}_i| = \sum_{i=1}^{n} |y_i - \sum_{j=1}^{m} \hat{b}_j x_{ij}| \qquad (2)$$

Taylor [7], clarifies the mechanics of LAE estimation. Suppose m = 1, (2) reduces to:

$$S = \sum_{i=1}^{n} |y_i - \hat{y}_i| = \sum_{i=1}^{n} |y_i - \hat{b}x_i| = \sum_{i=1}^{n} S_i \qquad (3)$$

A typical element, $S_i = y_i - \hat{b}x_i$ can be viewed as a broken line in the $(S,\hat{b})$ plane composed of two half-lines. S_i attains its minimum at $S_i = 0$. At that point:

$$\hat{b}_i = \frac{y_i}{x_i} \qquad (4)$$

The slope of the half-lines to the left and right of $\hat{b}_i$ is equal to $-|x_i|$ and $|x_i|$ respectively. So S_i is always convex and their sum S is also convex with slope at any b equal to the sum of the slopes of all S_i at that value of $\hat{b}$. Since the slope of the individual S_i changes only at the minimum of S_i, the minimum of S will lie on one of the minimum points of S_i. Thus, the regression line will pass through origin with the slope $\hat{b}_i$ in (4) that minimizes S. On the other hand, to find the LAE estimate of b we need to find only one observation. Furthermore, Taylor concludes that: "This implies, of course, that the regression line must pass through the observation corresponding to minimizing i. The regression line, therefore, is determined by the point of origin and observation associated with minimizing $\hat{b}_i$". But he did not continue this approach-minimizing S with respect to subscript i. This paper tries 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 Taylor concludes, when the number of parameters is m, m observations must lie on the regression hyperplane. On the other hand, m equations of the form in (5) are nesessary to specify the regression hyperplane.

$$y_i - \sum_{j=1}^{m} b_j x_{ij} \qquad (5)$$

In order to minimize (2), Fair [3] computed the LAE estimate of b_j by an interative weighted-least-squares procedure (see Maddala [4]). Instead of minimizing $\Sigma|u_i|$, Fair minimized $\Sigma \frac{u_i^2}{|u_i|} = \Sigma w_i u_i^2$, where $w_i = \frac{1}{|u_i|}$ used as weights. Once initial values for u_i are estimated by least squares, w_i are computed and they are used to minimize $\Sigma w_i u_i^2$. New residuals are then computed and procedure is repeated. Fair observed that the estimates of b_j did not change after the second or third interations. In cases where any residual is zero, continuation of procedure is impossible.

Charnes et al. [2] showed that the LAE estimator can be obtained as the solution to a linear-programming (LP) problem. Rewriting u_i as a difference between two non-negative auxiliary variables (v_i, w_i):

$$u_i = v_i - w_i = y_i - \sum_{j=1}^{m} b_j X_{ij} \qquad v_i, w_i \geq 0 \qquad i=1,\ldots,n \qquad (6)$$

$S^* = \sum_{i=1}^{n} (v_i + w_i)$ can then be defined as an objective function to be minimized with respect to b_j, v_i and w_i. Although S^* and S in (2) are different functionals, those values of v_i and w_i that minimize S^* is equivalent to the values of u_i which minimize S. Now the equivalent LP problem to LAE is to minimize S^*, with respect to b_j, b_j^*, v_i and w_i:

$$\min : \sum_{i=1}^{n} (v_i + w_i) \tag{7}$$

$$\text{S.To} : v_i - w_i + \sum_{j=1}^{m} b_j X_{ij} - \sum_{j=1}^{m} b_j^* X_{ij} = y_i \quad \forall i$$

$$v_i \geq 0 \quad \forall i \tag{8}$$

$$w_i \geq 0 \quad \forall i$$

$$b_j \geq 0 \quad \forall j$$

$$b_j \geq 0 \quad \forall j$$

$$i = 1,\ldots,n$$

$$j = 1,\ldots,m$$

Inserting b_j^* in (8) is a device to enable the coefficients of the independent variables to be of either positive or negative sings. It should be noted that only one member of each of the pairs of nonslack variables (b_j, b_j^*) and also slack variables (v_i, w_i) for each j and i can be nonzero in any solution. This comes from the dependency of each pair in the n equality constraints in (8) (see Taylor [7], PP.175-177). So minimization of (7) and (2) will be equal.

The first simplex tableau for solving LP problem in (7) and (8) is of dimension $(n+1) \times (2m+2n)$ which makes it computationally cumbersome; though it will always yield a solution. In order to avoid this difficulty an alternative to LP is proposed.

2. A proposed algorithm

For medel (1) consider the case of one independent variable with no intercept, namely $y_i = bX_i + u_i$. For the LAE estimate of b for the above model the following procedure can be suggested:

$$S = \sum_{i=1}^{n} |u_i| = \sum_{i=1}^{n} |y_i - bX_i|$$

$$= \sum_{i=1}^{n} (y_i - bX_i) \operatorname{sign}(y_i - bX_i)$$

$$= \sum_{i=1}^{n} X_i (\frac{y_i}{X_i} - b) \operatorname{sign}(\frac{y_i}{X_i} - b) \operatorname{sign}(X_i)$$

$$= \sum_{i=1}^{n} X_i (\frac{y_i}{X_i} - b) \operatorname{sign}(\frac{y_i}{X_i} - b) \qquad (9)$$

Let $Z_i = y_i/X_i$ and sort Z_i in a descending order. Rename the resulting ordered $Z_i (i=1,\ldots,n)$ to $Z_h (h=1,\ldots,n)$. Z_h elements sould have the following property:

$$Z_h > Z_l \quad \text{if} \quad h < l \quad \text{for} \quad h, \; l=1,\ldots,n$$

Rewrite (9) with ordered observations as:

$$S = \sum_{h=1}^{n} |X_n| (Z_h - b) \operatorname{sign}(Z_h - b) \qquad (10)$$

Let us denote the observation which will be on the regression line by (X_{t+1}, y_{t+1})- the(t+1)th observation. Value of Z_h is the slope of a ray passing through the origin and the hth observation. Therefore,

if, $h < t$ then; $Z_h > b$ and $u_h > 0$

if, $h = t$ then; $Z_h = b$ and $u_h = 0$

if, $h > t$ then; Z_h b and $u_h < 0$

We can now rewrite (10) as follows:

$$S = \sum_{h=1}^{t} |X_h| (Z_h - b) - \sum_{h=t+1}^{n} |X_h| (Z_h - b) \qquad (11)$$

To find the minimum of S we need to differentiate S with respect to b and subscript t and equate them to zero:

$$\frac{\partial S}{\partial b} = -\sum_{h=1}^{t} |X_h| + \sum_{h=t+1}^{n} |X_h| = 0 \qquad (12)$$

The differentiation of S with respect to subscript t must be the discrete derivative (see, Bender and Orszag [1]):

$$\frac{\Delta S}{\Delta t} = S(t+1) - S(t) = \sum_{h=1}^{t+1} |X_h|(Z_h - b) - \sum_{h=t+2}^{n} |X_h|(Z_h - b)$$

$$- \sum_{h=1}^{t} |X_h|(Z_h - b) + \sum_{h=t+1}^{n} |X_h|(Z_h - b) = 0$$

or:

$$\hat{b} = Z_{t+1} = \frac{y_{t+1}}{X_{t+1}} \qquad (13)$$

Note that (13) is again Taylor's relation (4). (12) and (13) are two equations with two unknowns t and b. t can be found by rewriting (12) as follow:

$$D_k = \sum_{h=1}^{k} |X_h| - \sum_{h=k+1}^{n} |X_h| \qquad k=1,\ldots,n \qquad (14)$$

It is obvious 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 less than zero, k is increased by one unit and procedure is repeated until D_k is greater than zero. When D_k reaches the first positive value, t+1=k. By this procedure, value of t+1 is found so the observation corresponding to this subscript (t+1=k) is selected (X_{t+1}, y_{t+1}). LAE estimate of b is found by substituting the values of X_{t+1} and y_{t+1} into (13).

The procedure applied to the model with only one parameter can not be simply generalized to the m parameter model in (1). To apply this method to (1), we need to reorder observations in a way that (2) could be

decomposed as follow:

$$S = \sum_{h=1}^{t} (y_h - \sum_{j=1}^{m} b_j X_{hj}) - \sum_{h=t+1}^{n} (y_h - \sum_{j=1}^{m} b_j X_{hj}) \qquad (15)$$

Which is the generalization of (11) to m parameters. But our first problem is to find a logic which enables us to form (15). On the other hand we need to reorder observations in such a way, when h, is less, equal or greater than t+1, u_n is greater, equal or less than zero; and as h increases from one to n, u_h decreases accordingly.

If (2) could be written as (15), we can again differentiate it with respect to b_j and t and set them equal to zero:

$$\frac{\partial S}{\partial b_j} = -\sum_{h=1}^{t} X_{hj} + \sum_{h=t+1}^{n} X_{hj} = 0 \qquad j=1,\ldots,m \qquad (16)$$

$$\frac{\Delta S}{\Delta t} = y_{t+1} - \cdot \sum_{j=1}^{m} b_j X_{t+1,j} = 0 \qquad (17)$$

m distinct values for t could be computed from (16) by using D_{kj} in (18) for each explanatory variables.

$$D_{kj} = \sum_{h=1}^{k} X_{hj} - \sum_{h=k+1}^{n} X_{hj} \qquad \begin{cases} j=1,\ldots,m \\ k=1,\ldots,n \end{cases} \qquad (18)$$

It should be noted that these m values for t, all will give a unique value for S (15), becaues all the errors corresponding to these ts have zero values. This becomes clear from the Taylor's conclusion that there exist m points on the regression hyperplane with zero errors. In (15), these m points locate sequentially, because we have reordered the observations such that when h increases from one to n errors decreases from the greatest positive to the lowest negative values. So these m zero errors will locate one after another nearly in the middle of n expressions in (15).

The procedure could be similar to that of (14). Corresponding to these m values for t, m observations are recognized. Substituting the values of these m observations in (17), m equations are found which could be solved simultaneously for m b_js. Note that these last m equations again confirm the Taylor's conclusion (5).

3. Evaluations

In comparison with LP method, the proposed Algorithm (PA) is very efficient both in execution time and storage requirements. In order to compare these two algorithm, 30 random experiments for the model $y_i=bX_i+u_i$ have been explored. In each experiment n normal random number u_i have been generated (see, M.J.Mojarrad [5]). Also n fix values are selected for X_i which are uniformly distributed. y_i has been computed for ten different values of b. The results of experiments are presented in table 1. In this table the execution times for each experiments have been compared. Total number of interations of LP method for reaching the minimum solution have been also encountered. This shows the complexity of the LP procedure.

Table 1

Comparison of CPU time for PA for different sample sizes

no. of interation	n=20 LP*	PA**	no. of interation	n=50 LP*	PA**	no. of interation	n=100 LP*	PA**
33	8.63	0.10	88	9.53	0.13	182	10.18	0.22
35	8.52	0.09	87	9.36	0.14	178	9.92	0.22
35	8.94	0.10	81	9.32	0.13	170	10.02	0.21
33	9.09	0.10	81	9.58	0.13	166	9.88	0.22
33	8.82	0.10	80	9.33	0.15	166	10.34	0.22
31	8.60	0.09	78	9.51	0.14	152	9.88	0.22
31	8.93	0.10	77	9.72	0.13	148	10.03	0.22
28	9.02	0.10	72	10.01	0.14	145	9.87	0.21
26	8.68	0.10	72	9.75	0.14	133	9.62	0.22
24	8.86	0.10	67	9.27	0.13	133	9.86	0.21

n : Sample size

LP : Linear Programming

PA : Proposed Algorithm

Time: CPU execution time in second

*Computed by MPSX/370 V1M4 PTF7

**Computed by Fortran H Extended Level 2.2

-In both algorithms pre-execution times of Compiler and Linkage-Editor have been excluded.

-All Runs are on BASF 7.68(MVS).

In all of these experiments PA in highly faster than LP.

Table 2 compares the storage requirements for both Algorithms for different sample sizes.

Table 2

Comparison of memory* requirements for LP and PA for different sample sizes.

	n = 20	n= 50	n = 100
LP	59392	71680	86016
PA	1406	1886	2686

* Bytes

As it can be seen from tables 1 and 2, the proposed algorithm is markedly superior to linear programming in execution time, storge requirements and also simplicity in programming.

References

1. C.M. Bender and Orszag, Advanced Mathematical Methods for Scientists and Engineers (McGraw-Hill, 1978) PP.36-37.

2. A. Charnes, W.W. Cooper, R.O. Ferguson, Optimal Estimation of Executive Compensation by Linear Programming, management science (1955) PP.138-151.

3. R.C. Fair, A Comparison of FIML and Robust Estimates of a Nonlinear Econometric Model, National Bureau of Economic Research working paper no.15 (Oct. 1973).

4. G.S. Maddala, Econometrics (McGraw-Hill, 1977) PP.308-314.

5. M.J. Mojarrad. The Application of Comparative Monte Carlo Methods to Econometrics: An Efficient Monte Carlo Study of Finite Sample Properties of Iterative Instrumental Variables Estimation, PH.D. dissertation, University of Pennsylvania (1977), PP.125-144.

6. B. Rosenberg, D. Carlson, The Sampling Distribution of Least Absolute Residuals Regression Estimates, working paper no. IP-164, Inst. of Business and Econ. Res., Univ. of California, Berkeley, California.

7. L.D. Taylor, Estimation by Minmizing the Sum of Absolute Errors, in: P. Zarembka (Ed.), Frontiers in Econometrics (Academic Press, New York, 1974) PP. 169-190.