Journal of Computational Mathematics, Vol.10, No.4, 1992, 291–295.

A SHORT NOTE ON AN L_1-NORM MINIMIZATION ALGORITHM*

Yang Zi-qiang Y. Yuan
(*Computing Center, Academia Sinica, Beijing, China*)

Abstract

In this short note, examples are constructed to show that a recent algorithm given by Soliman, Christensen and Rouhi[1] may give a non-optimal solution.

In [1], a linear least absolute value (LAV) estimate algorithm is presented. The linearly constrained LAV problem has the following form.

$$\min_{\theta\in\Re^n} ||H\theta - z||_1, \tag{1}$$

$$C\theta = d \tag{2}$$

where $H \in \Re^{m\times n}$, $z \in \Re^m$, $C \in \Re^{l\times n}$ and $d \in \Re^l$. One of the algorithms given in [1] is for solving problem (1)-(2). The algorithm can be restated as follows:

Algorithm 1[1]**. Step 1.** Calculate

$$\theta^* = \begin{bmatrix} H \\ C \end{bmatrix}^+ \begin{pmatrix} z \\ d \end{pmatrix}, \tag{3}$$

where B^+ is the Moore-Penrose generalized inverse of B.

Step 2. Compute

$$r^* = \begin{pmatrix} z \\ d \end{pmatrix} - \begin{bmatrix} H \\ C \end{bmatrix}\theta^*, \tag{4}$$

$$\bar{r} = \frac{1}{m+l}\sum_{i=1}^{m+l} r_i^*, \tag{5}$$

$$\sigma = \sqrt{\frac{1}{m-1}\sum_{i=1}^{m}(r_i^* - \bar{r})^2}. \tag{6}$$

Step 3. Let $J = \{j \mid |r_j^*| \le \sigma, 1 \le j \le m\}$ and

$$P_J = \sum_{j\in J} e_j e_j^T \tag{7}$$

* Received March 11, 1992.

where e_j $(j = 1, \cdots, m)$ are unit vectors in $\Re^m$.

Compute the new least squares solution

$$\theta^*_{\text{new}} = \begin{bmatrix} P_J H \\ C \end{bmatrix}^+ \begin{pmatrix} P_J z \\ d \end{pmatrix}, \tag{8}$$

$$r^*_{\text{new}} = z - H\theta^*_{\text{new}}. \tag{9}$$

Step 4. Let $I = \{i_1, \cdots, i_{n-l}\}$ be a subset of $\{1, \cdots, m\}$ which corresponds to the $n-l$ smallest residuals. Let $P_I = \sum_{i \in I} e_i e_i^T$ and solve

$$\begin{bmatrix} P_I H \\ C \end{bmatrix} \theta = \begin{pmatrix} P_I z \\ d \end{pmatrix} \tag{10}$$

to get $\bar{\theta}$. Accept $\bar{\theta}$ as a solution.

It should be noted that definition (6) is not the usual definition for standard deviation. We use (6) because it is the definition, as we understand, used by [1]. However, our examples are also valid if the usual definition of standard deviation is used. Another point that is worth mentioning is that r^*_{new} denotes first m residuals of the whole system, though θ^*_{new} is the least squares solution of a reduced system.

Soliman et al. [1] also extended the above algorithm to solving nonlinear LAV problems. For more details, see [1]. Now we give a linear LAV problem for which a non-optimal solution would be given by the above algorithm.

Example 1. Solve problem (1)–(2) with the following data:

$$H = \begin{bmatrix} 1 & 1 \\ 1 & 2 \\ 1 & 3 \\ 1 & 4 \\ 1 & 5 \\ \varepsilon & 0 \end{bmatrix}, \quad z = \begin{pmatrix} 2 \\ 2 \\ 3 \\ 4 \\ 0 \\ 0 \end{pmatrix}, \tag{11}$$

$$C = (1 \quad 6), \quad d = (5), \tag{12}$$

where $\varepsilon \in (0, 1)$ is a very small parameter.

Our example is very similar to Example 2.1 of [1]. We have added a very small row in the example, expecting that the corresponding residual will eventually be the smallest. The original $z_5 = 3$ (as in [1]) is changed to 0 to guarantee that the fifth residual will be the only measure to be deleted. It should be noted that, unlike Example 2.1 of [1], the above example can not be viewed as a straight line data fitting problem because $\varepsilon \neq 1$. However, we can still analyze the above algorithm for problem (1)–(2) with data given by (11)–(12).

It is easy to calculate

$$\theta^* = \frac{1}{105} \begin{pmatrix} 175 \\ 30 \end{pmatrix} + O(\varepsilon^2), \tag{13}$$