STATISTICS, OPTIMIZATION AND INFORMATION COMPUTING
Stat., Optim. Inf. Comput., Vol. **11**, September 2023, pp 798–810.
Published online in International Academic Press (www.IAPress.org)

Confidence intervals from local minimums of objective function

Azzouz DERMOUNE[1], Daoud OUNAISSI[2] and Yousri SLAOUI[3,*]

[1] *CNRS, Laboratoire Paul Painlev, UMR 8524, Université de Lille*
[2] *ESA, Angers, France*
[3] *Univ. Poitiers, Lab. Math. et Appl., Poitiers, France*

Abstract The objective function of the linear regression, using the least absolute deviations (LAD), is convex and more complex than the minimization of the sum of squares. It has only one global minimum but many minimizers. The weighted median plays a central role in this optimization. We propose a nonlinear regression using (LAD). Our objective function $f(a, l, s)$ is non-convex with respect to the parameters a, l, s, and is such that for each fixed l, s the minimizer of $a \to f(a, l, s)$ is the weighted median $med(x(l, s), w(l, s))$ of a sequence $x(l, s)$ endowed with the weights $w(l, s)$ (all depend on l, s). We analyse and compare theoretically the minimizers of the function $(a, l, s) \to f(a, l, s)$ and the surface $(l, s) \to f(med(x(l, s), w(l, s)), l, s)$. As a numerical application we propose to fit the daily infections of COVID 19 in China using Gaussian model. The parameters (a, l, s) are respectively the pick, the location of the pick and the width of the first wave of COVID 19 in China. We derive confident interval for the daily infections from each local minimum.

Keywords Gaussian model, Least absolute deviation, daily infection, Simplex algorithm, Nelder-Mead, Confidence intervals, `optim` function

AMS 2010 subject classifications 60G15, 78M50, 62F25.

DOI: 10.19139/soic-2310-5070-1738

1. Introduction

We follow the introduction proposed in [Bloomfield and Steiger(1980)]. The least absolute deviations (LAD) method of curve-fitting consists of fitting the data (x_i, y_i) to a function $f(x_i, \theta)$, with $i = 1, \dots, n$. The parameter $\theta \in \mathbb{R}^p$ minimizes the sum of absolute deviations

$$\sum_{i=1}^{n} |y_i - f(x_i, \theta)|.$$

According to [Eisenhart(1961)], in the linear regression case $f(x_i, \theta) = \sum_{j=1}^{p} x_{ij}\theta_j$, the minimization of the quantity

$$\sum_{i=1}^{n} |y_i - \sum_{j=1}^{p} x_{ij}\theta_j|$$

was suggested by Boscovitch (1757) (some asymptotic results are given in [Koenker and Bassett(1985)]). The latter objective function is convex with respect to the parameter θ. Hence it has only one minimum, but may have many minimizers.

*Correspondence to: Yousri Slaoui (Email: Yousri.Slaoui@math.univ-poitiers.fr).

ISSN 2310-5070 (online) ISSN 2311-004X (print)

When there is only one degree of freedom in the fit, i.e.,

$$\sum_{i=1}^{n} |y_i - \theta x_i| = \sum_{i=1}^{n} |x_i||\frac{y_i}{x_i} - \theta|,$$

the minimizing value of θ is the weighted median of the ratios $\frac{y_i}{x_i}$ with respect to the weights $|x_i|$, and $x_i \neq 0, for\ all\ i$. A minimizing θ may always be chosen such that some residual $y_i - \theta x_i$ vanishes (This idea is developed in Section 4). This observation motivates the minimization of the two parameter minimization of

$$\sum_{i=1}^{n} |y_i - \theta_1 - \theta_2 x_i|.$$

The method, described by [Rhodes(1930)] and [Karst(1958)], is the basis of a computer algorithm published by [Sadovski(1974)]. First we minimize $f_1(n, \theta_2) = \sum_{i=1}^{n} |y_i - \theta_2 x_i|$. Pick a minimizer $\theta_2(1)$ (as a first candidat for θ_2) and the index i_1 such that the residual $y_{i_1} - \theta_2(1)x_{i_1} = 0$. Now we minimize

$$f_2(n, \theta_1, \theta_2) = \sum_{i=1}^{n} |y_i - \theta_1 - \theta_2 x_i|, \quad \text{under the constraint} \quad y_{i_1} - \theta_1 - \theta_2 x_{i_1} = 0.$$

We derive that $\theta_1 = y_{i_1} - \theta_2 x_{i_1}$, and then we minimize $f_2(n, y_{i_1} - \theta_2 x_{i_1}, \theta_2) = \sum_{i=1}^{n} |y_i - y_{i_1} - \theta_2(x_i - x_{i_1})|$. We obtain the minimizer $(\theta_1(2), \theta_2(2))$ and an index i_2 such that $y_{i_2} - \theta_1(2) - \theta_2(2)x_{i_2} = 0$. Observe that $f_2(n, \theta_1(2), \theta_2(2)) < f_1(n, \theta_2(1)$. Having a minimizer $(\theta_1(k), \theta_2(k)$ and the index i_k, we consider the minimization of

$$f_{k+1}(n, \theta_1, \theta_2) = \sum_{i=1}^{n} |y_i - \theta_1 - \theta_2 x_i|, \quad \text{under the constraint} \quad y_{i_k} - \theta_1 - \theta_2 x_{i_k} = 0.$$

We repeat this algorithm until the minimizer $(\theta_1(k), \theta_2(k))$ does not change, or equivalently until the decreasing sum of absolute deviations $f_k(n, \theta_1(k), \theta_2(k))$ converges.

This algorithm may degenerate in the sense that more than two residuals are zero. In this case the algorithm may cycle endlessly or may terminate prematurely (see [Bloomfield and Steiger(1980)]). [Narula(1977)] and [Bloomfield and Steiger(1980)] described an efficient method based on linear programming. This method generalizes [Rhodes(1930)] and [Karst(1958)] technique and lends itself for multiple regression (1.1).

As we have just seen even in the linear regression case of LAD optimization is inherently more complex than the minimization of the sum of squares. The interest in LAD method is associated with the development of robust methods. LAD method is more resistant to the outliers in the data (see [Dielman(2005)] and [Li and Arce(2004)]).

The aim of our work is to analyse LAD minimization using a nonlinear regression motivated by the daily infections of COVID 19 in China during the first wave. The parameter $\theta = (a, l, s)$, with a, l, s denote respectively the pick, the location of the pick and the width of the first wave of COVID 19. The variable $t = 1, \ldots, T$ represents day $1, \ldots, T$. We denote $I(t)$ the observed number of infected persons at time $t \in [1, T]$ with $T \leq 60$ (see Figure 1). Justified by the sigmoidal nature of a pandemic, we propose the Gaussian model (see [Barmparis and Tsironis(2020)])

$$I_m(t) = a \exp(-\frac{(t-l)^2}{s^2})$$

as a prediction of $I(t)$. The subscript m is used to distinguish data $I(t)$ from the model $I_m(t)$. The parameters of the model are respectively the pick a, the location of the pick l and the width s^2.

To estimate the three parameters a, l, s based on the T observations, we consider LAD nonlinear regression

$$f(T, a, l, s) = \frac{\sum_{t=1}^{T} |I(t) - I_m(t)|}{T}.$$

We expect that the minimization of LAD in the nonlinear case to be more complex than the linear case.

2. Probabilistic interpretation of LAD regression

Let us assume that

$$I(t) = I_m(t) + e(t),$$

where the errors $(e(t))$ are i.i.d. with the common probability distribution

$$\frac{1}{2\lambda}\exp(-\frac{|e|}{\lambda}), \quad \text{with the scale} \quad \lambda > 0.$$

Based on the data $(I(1), \ldots, I(T))$ the likelihood is equal to

$$\prod_{t=1}^{T} \frac{1}{2\lambda}\exp(-\frac{|I(t) - I_m(t)|}{\lambda}).$$

It comes that the maximum likelihood estimator of the parameters a, l, s and λ are

$$\begin{cases} (\hat{a}, \hat{l}, \hat{s}) & = & \arg\min\{f(T, a, l, s) : a, l, s\} \\ \hat{\lambda} & = & f(T, \hat{a}, \hat{l}, \hat{s}). \end{cases}$$

In practice $(\hat{a}, \hat{l}, \hat{s})$ are given by an algorithm of optimization, and usually they are only local minimizer. Having $(\hat{a}, \hat{l}, \hat{s})$ and the scale $\hat{\lambda}$ we derive a confidence interval for $I(t)$ with $t > T$ as

solution of the equation

$$\int_{-q}^{q} \frac{1}{2\hat{\lambda}}\exp(-\frac{|e|}{\hat{\lambda}})de = 0.95$$

is given by $q = -\hat{\lambda}\ln(0.05) = 2.995732\hat{\lambda}$. We derive the confidence interval

$$IC_{0.95}\left[I(t)\right] = \left[\hat{a}\exp(-\frac{(t-\hat{l})^2}{(\hat{s})^2}) - 2.995732\hat{\lambda}; \hat{a}\exp(-\frac{(t-\hat{l})^2}{(\hat{s})^2}) + 2.995732\hat{\lambda}\right]$$

of $I(t)$ with the confidence level 0.95.

3. Solving the proposed LAD regression using Nelder-Mead algorithm

The Nelder-Mead algorithm ([Dielman(2005)], [Lagarias et al.(1998)] and [Gao et al.(2010)]) is able to optimize functions without derivatives. It is a simplex method for finding a local minimum of a function, and is the most widely used direct search method for solving optimization problem and is considered as one of the most popular derivative free nonlinear optimization algorithm.

We are going to solve our proposed LAD regression using the simplex algorithm Nelder-Mead implemented by `optim` function in `R` software. It's known that the output of the `optim` function depends on the initialization and is in general not a minimizer of the objective function. Restarting the Nelder-Mead algorithm from the last solution obtained (and continuing to restart it until there is no further improvement) can only improve the final solution and the latter is in general a local minimizer.

Based on the `optim()` function, we define the `opt`(T, θ_0, K) function developed in Algorithm 1 which allows to produce the output of Nelder-Mead algorithm of $f(T, \cdot)$ after K restarts with the initialization θ_0.

The source code of the `opt()` function is given in Appendix B.1.

4. LAD regression analysis using weighted median

Before going forward we recall the weighted median definition.

Algorithm 1 The ouput of `optim` function after K restarts.

Input: T, and K.
initialization θ_0,
for $k = 1, \dots, K$ **do**
$\theta(k, \theta_0) =$optim$(\theta(k-1, \theta_0), f(T, \cdot))$ // *optim function applied to $f(T, \cdot)$ with the initialization $\theta(k-1, \theta_0)$.*
end for
output: $(\theta(K, \theta_0), f(T, \theta(K, \theta_0)) = \mathtt{opt}(T, \theta_0, K)$.

4.1. Weighted median

We recall in the following proposition the definition and the calculation of the weighted median. For more details, we advise the reader to see the work of [Novoselac(2020)].

Proposition 1
Let us consider a sequence $(x(t), w(t))$ of real numbers with positive weighted $w(t) > 0$ and $t = 1, \dots, T$. The minimizer of the function $a \to \sum_{t=1}^{T} w(t)|a - x(t)|$ (called the weighted median) is given as follows. We calculate the permutation $p(1), \dots, p(T)$ which rearranges the sequence $(x(t) : t = 1, \dots, T)$ into ascending order. We form the sequence $(w(p(t)) : t = 1, \dots, T)$, then we find the largest integer k which satisfies

$$\sum_{t=1}^{k} w(p(t)) \leq \frac{\sum_{t=1}^{T} w(t)}{2}.$$

If

$$\sum_{t=1}^{k} w(p(t)) < \frac{\sum_{t=1}^{T} w(t)}{2},$$

then weighted median $a = x(p(k+1))$.

If $\sum_{t=1}^{k} w(p(t)) = \frac{\sum_{t=1}^{n} w(t)}{2}$, then the weighted median $[x(p(k), x(p(k+1))]$ is equal to the interval $[x(p(k), x(p(k+1))]$.

4.1.1. Back to our proposed LAD regression

The following equality

$$f(T, a, l, s) = \frac{1}{T} \sum_{t=1}^{T} \exp(-\frac{(t-l)^2}{s^2})|a - I(t)\exp(\frac{(t-l)^2}{s^2})|$$

leads us to consider the following corollary.

Corollary 1
For each (l, s) fixed the minimum of the function $a \to f(T, a, l, s)$ is attained at the weighted median $a(T, l, s)$ of the sequence $(x(t) = I(t)\exp(\frac{(t-l)^2}{s^2}) : t = 1, \dots, T)$ endowed with the weights $(w(t) = \exp(-\frac{(t-l)^2}{s^2}) : t = 1, \dots, T)$. Moreover if (a^*, l^*, s^*) is a local minimizer of the function $(a, l, s) \to f(T, a, l, s)$ then a^* is the weighted median of $\Big(x^*(t) = I(t)\exp(\frac{(t-l^*)^2}{(s^*)^2}) : t = 1, \dots, T\Big)$ endowed with the weights $\Big(w^*(t) = \exp(-\frac{(t-l^*)^2}{(s^*)^2}) : t = 1, \dots, T\Big)$.

Proof
We observe that for l, s fixed, the map $a \to f(T, a, l, s)$ is a convex function. Now, let us assume that (a^*, l^*, s^*) is

a local minimizer of the function $(a,l,s)\to f(T,a,l,s)$. Then a^* is the global minimizer of the convex function $a\to f(T,a,l^*,s^*)$. Hence a^* is the weighted median of $(x^*(t)=I(t)\exp(\frac{(t-l^*)^2}{(s^*)^2}):t=1,\ldots,T)$ endowed with the weights $(w^*(t)=\exp(-\frac{(t-l^*)^2}{(s^*)^2}):t=1,\ldots,T)$. □

4.1.2. Comparison of the minimizers of the surface $(l,s)\to f(T,a(T,l,s),l,s)$ and the minimizers of the map $(a,l,s)\to f(T,a,l,s)$

The following proposition is obvious.

Proposition 2

1) For each fixed a, the surface $(l,s)\to f(T,a,l,s)$ is above the surface $(l,s)\to f(T,a(T,l,s),l,s)$ and they intersect at the curve $a=a(T,l,s)$.

2) If (l^*,s^*) is a local minimizer of the surface $(l,s)\to f(T,a(T,l,s),l,s)$, then $(a(T,l^*,s^*),l^*,s^*)$ is also a local minimizer of the map $(a,l,s)\to f(T,a,l,s)$.

3) The local minimizers of the map $(a,l,s)\to f(T,a,l,s)$ belong to the set $\{(a(T,l,s),l,s):l,s\}$. If $(a(T,l^*,s^*),l^*,s^*)$ is a local minimizer of the map $(a,l,s)\to f(T,a,l,s)$, then in general (l^*,s^*) is not a local minimizer of the surface $(l,s)\to f(T,a(T,l,s),l,s)$. However if $(a(T,l^*,s^*),l^*,s^*)$ is a global minimizer of $(a,l,s)\to f(T,a,l,s)$, then (l^*,s^*) is also a global minimizer of $(l,s)\to f(T,a(T,l,s),l,s)$.

Proof

1), 3) are obvious. The proof of 2) works as follows. There exists a neighborhood V of (l^*,s^*) such that

$$f(T,a(T,l,s),l,s)\geq f(T,a(T,l^*,s^*),l^*,s^*)$$

for each $(l,s)\in V$. By definition of $a(T,l,s)$, we have $f(T,a,l,s)\geq f(T,a(T,l,s),l,s)$ for each couple (l,s). It follows that $(a(T,l^*,s^*),l^*,s^*)$ is a local minimizer of $(a,l,s)\to f(T,a,l,s)$. □

Proposition 3

Assume that $(l,s)\to f(T,a(T,l,s),l,s)$ has only a global minimizer (one mode surface). Then the map $(a,l,s)\to f(T,a,l,s)$ does not necessarily have one mode, and $(l,s)\to a(T,l,s)$ is discontinuous at any couple (l^*,s^*) such that $(a(T,l^*,s^*),l^*,s^*)$ is a local minimizer of the map $(a,l,s)\to f(T,a,l,s)$.

Proof

By definition of local minimizer, there exists a neigbordhood V of $(a(T,l^*,s^*),l^*,s^*)$ such that $f(T,a,l,s)\geq f(T,a(T,l^*,s^*),l^*,s^*)$ for each point $(a,l,s)\in V$. Necessarily $(a(T,l,s),l,s)$ is not in V for at least one point (l,s) near (l^*,s^*), if not $f(T,a(T,l,s),l,s)\geq f(T,a(T,l^*,s^*),l^*,s^*)$ for all point (l,s) near (l^*,s^*), and then (l^*,s^*) is a local minimizer of the map $(l,s)\to f(T,a(T,l,s),l,s)$. This is absurd because $(l,s)\to f(T,a(T,l,s),l,s)$ has only a global minimizer. □

4.2. Exploration of the minimizers of $(a,l,s)\to f(T,a,l,s)$

We propose three methods. In the first and the second method, we grope to find the global minimum and some local minima. In the third method we propose an algorithm which explores all the local minima starting from the global minimum.

4.2.1. Method 1

Let us consider $T=10$. The pick is not yet attained, and then we expect that the pick is higher that $\max(I(t):t=1,\ldots,T)$. Then, it is natural to start from the initializations $a_0=j\max(I(t):t=1,\ldots,T)$, $l_0=T+1$, $s_0=1,\ldots,10$, $j=1,2,\ldots$.

4.2.2. Method 2

We start from θ_1, then calculate $\theta_2=\mathtt{opt}(\theta_1,f(2,\cdot),k)$, …, $\theta_T=\mathtt{opt}(\theta_{T-1},f(T,\cdot),K)$. The initialization at time T equals $\theta_{T-1}=\mathtt{opt}(\theta_{T-2},f(T-1,\cdot),k)$. It depends on the previous observations $I(1),\ldots,I(T-1)$ and a fixed number of restarts k. By varying θ_1 and the number of restarts k we obtain a large number of initializations of the algorithm $\texttt{opt}(init, f(T,\cdot), K)$. We choose K very large in order to assure the convergence of `optim` function at time T. However, the number of restarts k is arbitrary. We found that imposing the convergence of the `optim` function before T provides fewer local minima than not imposing it.

4.2.3. Method 3

The general idea of this method is described in the following three steps:

Step 1: we start from any initialization a_0, l_0, s_0 then we execute the `optim` function with several restarts until convergence to a local minimum a_1, l_1, s_1.

Step 2: We draw a sample of points around a_1, l_1, s_1 according to the truncated Gaussian law of average a_1, l_1, s_1 with the constraint $l > l_1$, by keeping the same variance and using the function `rtnorm` of the package `msm`. For each of the points we repeat step 1. We thus collect a new list of local minimum.

Step 3: We repeat Step 2 around each local minimum.

Now we give more details about our algorithm:

Algorithm 2 Method 3

Starters: T, N, K are integer values and a, l and s are real values.

Initialization: Assign to each (a, l, s) an initial value (a_0, l_0, s_0) calculate $(a_0^{opt}, l_0^{opt}, s_0^{opt}, min_0^{opt}) = \mathbf{opt}(\mathbf{T}, \mathbf{c}(\mathbf{a_0}, \mathbf{l_0}, \mathbf{s_0}), \mathbf{K})$, using the `opt()` function with K restarts proposed in Algorithm 1. The source code is given in Appendix B.1.

Step1: Generate a sample of points (s_m, l_m) according to a Truncated normal distribution of average (l_0^{opt}, s_0^{opt}) such that $l_m \geq l_0^{opt}$ and $s_m \geq 1$ for $m = 1, ..., N$, by keeping the same variance and using the function `rtnorm` in the `msm` library (see Appendix A).

Step2: Compute the **opt()** function (see Algorithm 1) with K restarts and the initialization $(a(T, l_m, s_m), l_m, s_m)$ to obtain the list $\mathbb{S} = \{(a_m^{opt}, l_m^{opt}, s_m^{opt}, min_m^{opt}) : \quad m = 1, ..., N\}$. For each element $P \in \mathbb{S}$ we calculate the **opt()** function with one restart, then we obtain P_1^*.

while $P \neq P_1^*$ **do**

we calculate **opt()** with one restart and the initialization P_1^*. We obtain the point P_2^*. We repeat this process until $P_n^* = P_{n+1}^*$. Then $P^* = P_n^*$ is a true minimizer.

end while

Step3: Select from Step2 the set $\mathbb{S}^*$ of the true minimizers P^*, and repeat Step1 and Step2 for the element (l^*, s^*) which corresponds to the $max(min_n^{opt})$ of the set of the true minimizers.

Stopping criterion: We set a threshold$= 40000$.

Using the algorithm of the third method for the surface $(l, s) \to f(T, a(T, l, s), l, s)$, we show that it has only one minimizer. As shown in proposition 3, this minimum coincides with the global minimum of the map $(a, l, s) \to f(T, a, l, s)$.

The numerical results of Method 2 are given in Table 3. We note that this last method allows us to obtain a large number of local minimum.

5. Numerical results

In China the COVID 19 appeared on December 23, 2019 in the Wuhan region and after its fast-initial spreading, strict rules of social distancing were imposed almost a month later. Three months after the initially reported cases, the spreading in China has subsided. Data of China Figure 1 are extracted from owid/covid-19-data available on the web. The pick ($\texttt{pick} = 15136$), and its location ($\texttt{location} = 22$).

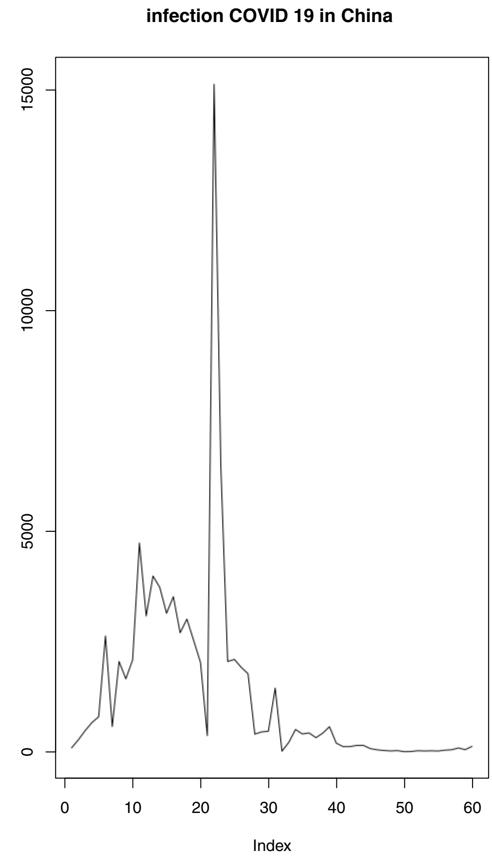

Figure 1. Prediction of pick and location by considering China data.

As an example of the function optim developed in Appendix B, we consider $T = 10$, $K = 3$, with the initialization $\theta_0 = (a_0 = I(T), l_0 = 11, s_0 = 10)$. Table 1 gives the outputs of three iterations in opt((T, θ_0, K)). We observe the convergence of optim function in 2 restarts to the local minimizer 319.5446.

Table 1. The outputs of the three iterations in opt((T, θ_0, K))

a.opt	l.opt	s.opt	min.loc
2108.690	10.60930	6.136000	319.7714
2091.262	10.22868	5.787523	319.5446
2088.911	10.11930	5.712179	319.5446

Figure 2 show the behavior of the weighted median around the local minimizer $a^* = 2088.911$, $l^* = 10.11930$, $s^* = 5.712179$ at time $T = 10$. More precisely, we plot $l \in [10, 22] \to a(10, l, 5.712179)$ and $s \in [4, 8] \to a(10, 10.11930, s)$.

The minimization of the function $(l, s) \to f(T, a(T, l, s), l, s)$ at time $T = 10$ using optim function with the initialization $l_0 = 11$, $s_0 = 10$ converges only on 1 start to $l^{opt} = 10.119$ and $s^{opt} = 5.712$ with the values

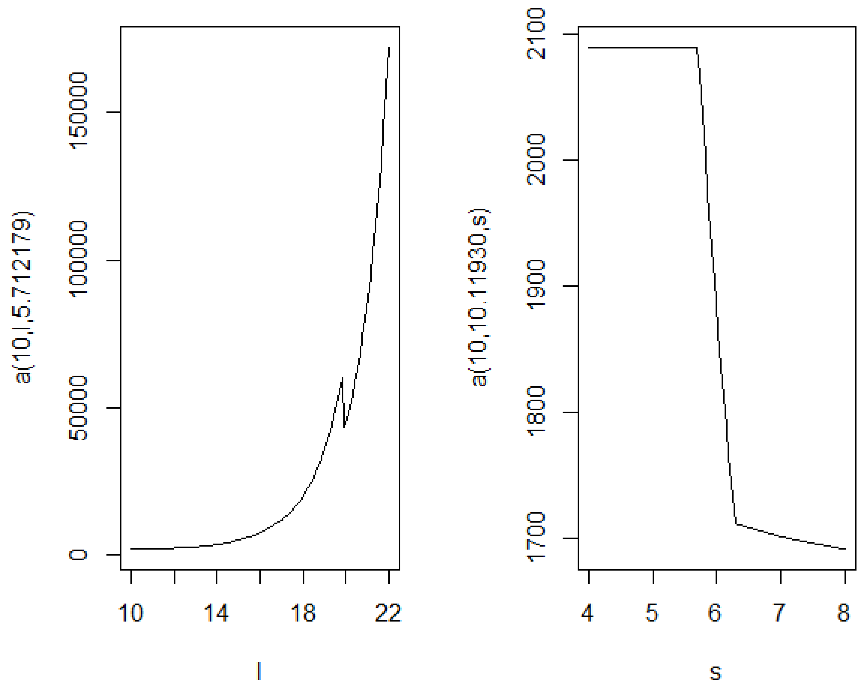

Figure 2. The behavior of the weighted median around the local minimizer $a^* = 2088.911$, $l^* = 10.11930$, $s^* = 5.712179$ at time $T = 10$.

$f(T, a(T, l^{opt}, s^{opt}), l^{opt}, s^{opt}) = 319.545$ and $a(T, l^{opt}, s^{opt}) = 2088.920$. We will see later that the surface $(l, s) \to f(T, a(T, l, s), l, s)$ has only one mode.

The output of `optim` function with 40 restarts is given in Table 3. The largest minimum 714.2 corresponds to the minimizer $a^* = 4370.113$, $l^* = -2.916943e + 08$, $s^* = -2.125995e + 08$. It's clear that this minimizer is not realistic in our case. The other minimums are suitable. The minimum 330.6708 corresponds to the minimizer $a^* = 5082.187$, $l^* = 21.17187$, $s^* = 11.845252$. We recall that the observed location is $l = 22$. The minimum 337.0503 corresponds to the minimizer $a^* = 15377.106$, $l^* = 32.11637$, $s = 15.651661$. We recall that the observed pick is $a = 15136$.

Remark 1
By minimizing the surface $(l, s) \to f(T, a(T, l, s), l, s)$ using `optim` function and starting from $l_0 = T + 1$, $s_0 = 1, \dots, 10$ we obtain the minimizer $l^{opt} = 10.119$ and $s^{opt} = 5.712$ with the values $f(T, a(T, l^{opt}, s^{opt}), l^{opt}, s^{opt}) = 319.545$ and $a(T, l^{opt}, s^{opt}) = 2088.920$. The convergence of `optim` function happens with only one start.

The reported `R` product (see results 3) shows the output of Method 1, by considering $T = 10$, $N = 1000$, $a_0 = I[T]$, $l_0 = T + 1$, $s_0 = T$, $K_0 = 40$.

```
 [1] 319.5446 319.5446 319.5446 319.5446 319.5446 319.5446 319.5446 319.5446 319.5446 319.5446 319.5446 319.5446 319.5446
[14] 319.5446 319.5446 319.5446 319.5446 319.5446 319.5446 319.5446 319.5446 319.5446 319.5446 319.5446 319.5446 319.5446
[27] 319.5446 319.5446 319.5446 319.5446 319.5446 319.5446 319.5446 319.5446 319.5446 319.5446 321.8664 321.8707 322.0838
[40] 322.1098 322.1202 322.3761 322.7857 324.1939 324.8832 330.2843 330.6708 334.1362 334.6030 334.9502 334.9526 335.0073
[53] 335.0288 335.2480 335.2940 335.3267 335.4049 335.4492 335.4640 335.5625 335.7484 335.8189 335.9166 335.9730 336.7073
[66] 336.7277 337.0503 337.1877 337.1970 337.2580 337.2758 337.2938 337.3916 337.3930 337.5485 337.6380 337.6463 337.7860
[79] 337.8019 337.8070 337.8170 337.8386 337.9409 338.1015 338.1119 338.1417 338.1456 338.3395 338.3605 338.3642 338.3666
[92] 338.4161 338.4202 338.4308 338.4330 338.4563 338.5049 338.5340 338.5468 714.2000
```

Figure 3. The output of `optim` function with 40 restart by using the first Method

The numerical results of Method 2 are given in Table 2 below with $k = 1, ..., 5,\ s = 1, ..., 10,\ T = 10,\ \theta_1 = (I(1), 1, s)$:

Table 2. The numerical results of Method 2 with $k = 1, ..., 5$, $s = 1, ..., 10$, $T = 10$, $\theta_1 = (I(1), 1, s)$

a.opt	l.opt	s.opt	min.opt
2088.911	10.119	5.712	319.545
2145.336	11.075	6.528	321.138
2301.818	12.186	7.000	322.358
2934.332	15.245	8.991	325.186
9351.858	27.248	14.086	335.047
13688.820	30.982	15.301	336.655
13791.980	31.055	15.324	336.682
13821.960	31.076	15.331	336.689
14493.080	31.539	15.474	336.854
14779.040	31.730	15.533	336.920
35718.400	40.292	17.977	339.114
36466.400	40.492	18.030	339.152
38891.310	41.114	18.194	339.267

We showed numerically by methods 1 and 2 that $f(10, 2088.911, 10.119, 5.712) = 319.545$ is the global minimizer of $(a, l, s) \to f(10, a, l, s)$. As at time $T = 10$, we know that the location of the pick is not yet attained, then it's natural to look for local minimizers $(a^{opt}, l^{opt}, s^{opt})$ such that $l^{opt} > 10.119$.

The numerical results of Method 3 are given in Table 3 below with $T = 10$, $N = 1000$, $a_0 = I[T]$, $l_0 = T + 1$, $s_0 = T$, $K_0 = 40$.

5.1. Confidence intervals from local minimizers from data before the pick for $T = 10$

We recall that the confidence interval for $I(t)$ from the minimizer $(\hat{a}, \hat{l}, \hat{s})$ is given by

$$IC_{0.95}\,[I(t)] = \left[\hat{a}\exp(-\frac{(t-\hat{l})^2}{(\hat{s})^2}) - 2.995732\hat{\lambda};\, \hat{a}\exp(-\frac{(t-\hat{l})^2}{(\hat{s})^2}) + 2.995732\hat{\lambda}\right] \quad \text{with} \quad t > 10.$$

In figure 4 we present the confidence intervals for four local minimum of the list $T = 10$. An R source code is given in Appendix B, which can be used to determine the confidence intervals for the other values of T once the list of minimum is determined by using one of the three considered methods.

Table 3. The numerical results of Method 3 with $T = 10$, $N = 1000$, $a_0 = I[T]$, $l_0 = T + 1$, $s_0 = T$, $K_0 = 40$

iter	a.opt	l.opt	s.opt	min.opt
1	2088.910	10.120	5.710	319.540
2	2185.910	11.400	6.540	321.570
3	2201.050	11.520	6.610	321.700
4	2287.700	12.100	6.950	322.280
⋮	⋮	⋮	⋮	⋮
40	2949.790	15.310	9.030	325.230
41	3164.510	16.110	9.480	325.830
42	3187.810	16.200	9.530	325.890
⋮	⋮	⋮	⋮	⋮
80	6087.090	22.990	12.560	332.320
81	6123.980	23.050	12.580	332.370
82	6152.530	23.100	12.600	332.410
⋮	⋮	⋮	⋮	⋮
130	14,333.470	31.430	15.440	336.820
131	16,465.130	32.780	15.850	337.270
133	17,579.660	33.420	16.040	337.460
134	17,662.490	33.470	16.060	337.480
⋮	⋮	⋮	⋮	⋮
170	25146.960	36.900	17.050	338.390
171	25290.870	36.950	17.060	338.400
172	25318.860	36.960	17.070	338.410
⋮	⋮	⋮	⋮	⋮
218	37942.390	40.880	18.130	339.220
219	38522.670	41.020	18.170	339.250
220	38782.280	41.090	18.190	339.260

6. Conclusion

In this work we considered the map

$$f : (a, l, s) \to f(T, a, l, s) = \sum_{t=1}^{T} |a \exp(-\frac{(t-l)^2}{s^2}) - I(t)|,$$

with T and $I(1), \dots, I(T)$ are the data. We have associated to f the surface

$$S : (l, s) \to f(T, med(x(l, s), w(l, s)), l, s)$$

with the sequence $x(l, s) = (I(t)\exp(\frac{(t-l)^2}{s^2}, t = 1, \dots, T)$, and the weights $w(l, s) = (\exp(-\frac{(t-l)^2}{s^2}, t = 1, \dots, T)$ and $med(x(l, s), w(l, s))$ denotes the weighted median of $x(l, s)$ endowed with the weights $w(l, s)$. We showed that if (l^*, s^*) is a local minimizer of S, then $(med(x(l^*, s^*), l^*, s^*)$ is also a local minimizer of f. The converse is in general false, i.e., if (a^*, l^*, s^*) is a local minimizer of f, then (l^*, s^*) is not in general a local minimizer of the surface S. However if (a^*, l^*, s^*) is the global minimum of f, then (l^*, s^*) is also the global minimum of the surface S. We showed that if S has only a global minimum, then $(l, s) \to f(T, med(x(l, s), w(l, s)), l, s)$ is discontinuous at each local minimizer (a^*, l^*, s^*). Using the data of the daily infections of COVID 19 in China during the first wave, we showed numerically that the map f has a huge number of local minimums, but the surface S has only a

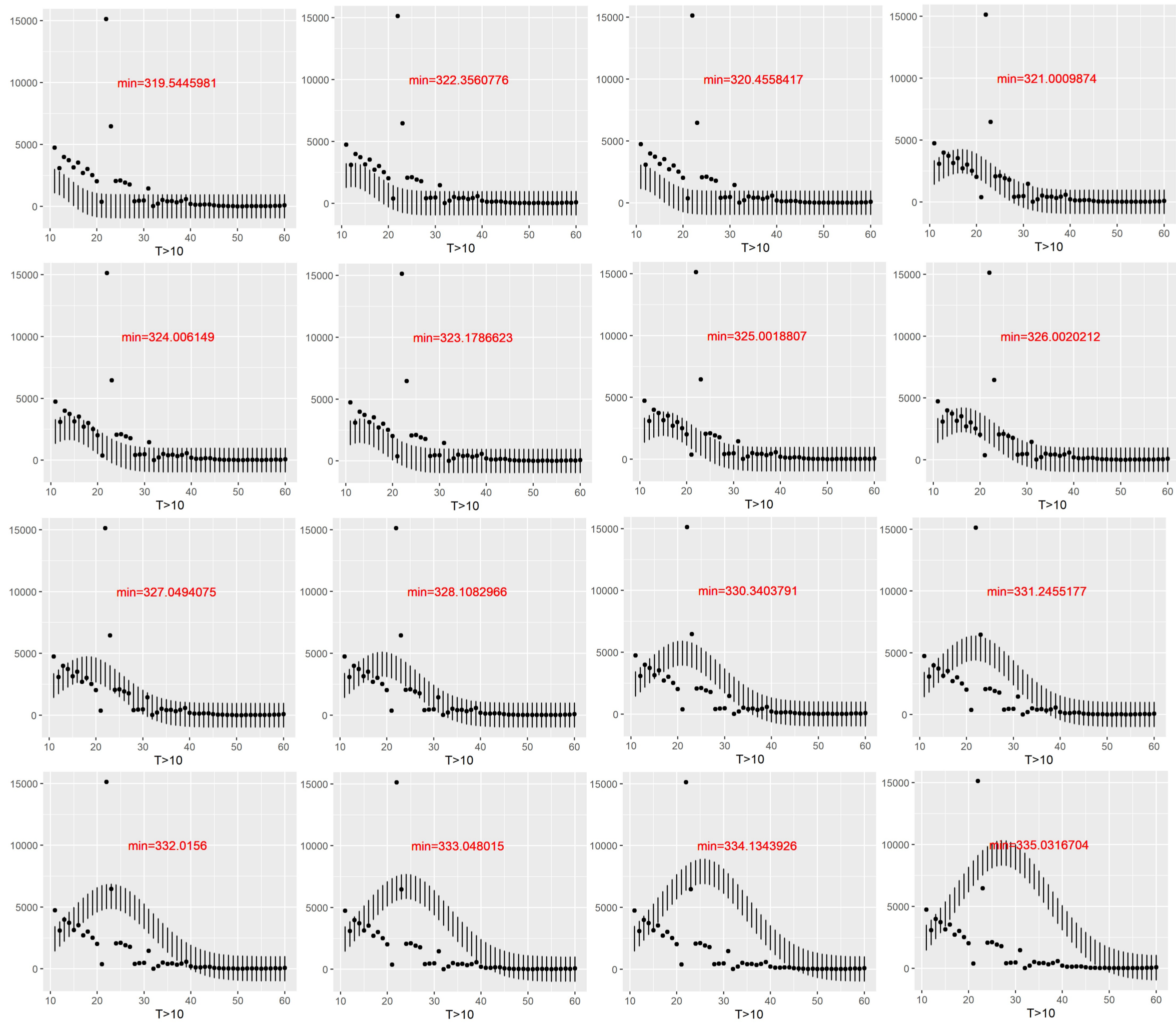

Figure 4. Confidence intervals for the mimnimum list of $T = 10$

global minimum which is also the global minimum of the map f. We can extend this problem to any continuous map $(a, l, s) \to f(a, l, s)$ such that the curve $a \to f(a, l, s)$ is convex. It follows that the graph of f is the union of the convex curves $a \to f(a, l, s)$. By Denoting by $a(l, s)$ a minimizer of the convex curve $a \to f(a, l, s)$, the set of the local minimizers $(a^*, l^*, s^*, f(a^*, l^*, s^*))$ of f is included in the surface $S = (a(l, s), l, s, f(a(l, s), l, s))$. The global minimum of f coincides with the global minimum of the surface S. If the minimums of S is reduced to its global minimum, then the surface S is discontinuous on any local minimum of f. Today we have no theoretical proof of the unicity of the minimum of the surface S. We will deal with this problem in a future work. We will also try to understand the graph in $\mathbb{R}^4$ of the map $(a, l, s) \to f(a, l, s)$ near a minimizer (a^*, l^*, s^*) using the graph in $\mathbb{R}^3$ of the maps $(a, l) \to f(a, l, s^*)$, $(a, s) \to f(a, l^*, s)$, and $(l, s) \to f(a^*, l, s)$.

A. Truncated normal distribution

The truncated normal distribution is defined for all x, such that $-\infty \leqslant a \leqslant x \leqslant b \leqslant +\infty$, by:

$$f(x; \mu, \sigma, \text{lower} = a, \text{upper} = b) = \frac{\exp\left(-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right)}{\int_a^b \exp\left(-\frac{1}{2}\left(\frac{u-\mu}{\sigma}\right)^2\right) du},$$

where μ is the mean, σ is the standard deviations, a and b are respectively the lower and the upper truncation points.

The random generation for the truncated Normal distribution was obtained by the function `rtnorm` in the `msm` library, (see [Robert(1995)]).

B. R source code

Here we give a source code for a better understanding of the proposed method.

B.1. opt() function

```
library(optim)
opt<-function(T,init,K){
  y<-function(theta) f(T,theta)
  matopt=NULL
  iter<-0
  while (iter<K) {
    init=optim(init,y)$par
    val.opt=optim(init,y)$value
    matopt=rbind(matopt,c("init"=init,"Min.loc"=val.opt))
    iter=iter+1
  }
  return(matopt)
}
```

B.2. Confidence intervals plot

```
library(ggplot2)
library(openxlsx)
conf.int.min=function(T,Min.list,T.inf,T.sup){
  # given T
  # Min.list is the list of minimum obtained for T
  ## T.inf = T+1 and T.sup=upper limit of days
j=1:(60-T)
Tj=T+j

born_sup=NULL
born_inf=NULL
confid.int=NULL
int_T=NULL
for(i0 in 1:nrow(Min.list)){
  for(i00 in 1:length(Tj)){
    born_sup=Min.list[i0,1]*exp(-(Tj[i00]-Min.list[i0,2])^2/(Min.list[i0,3]^2))
                                        +2.995732*Min.list[i0,4]
```

```
    born_inf=Min.list[i0,1]*exp(-(Tj[i00]-Min.list[i0,2])^2/(Min.list[i0,3]^2))
                            -2.995732*Min.list[i0,4]
    int_T=rbind(int_T,c(born_inf,i[Tj[i00]],born_sup))
  }
  colnames(int_T)=c("lower","i","upper")
  confid.int=cbind(confid.int,int_T)
  int_T=NULL

}
confid.int=data.frame(confid.int)
row.names(confid.int)=as.character(Tj)
#export all confidence intervals
write.xlsx(confid.int,file=paste0("confid_int_",T,".xlsx"))
#creation of a matrix which is equal to the matrix of
#confidence intervals for use with ggplot below
ggplot.confid=confid.int
iter.plot=1
while(ncol(ggplot.confid)>=3){
  ggsave(ggplot(ggplot.confid[,1:3], aes(x=T.inf:T.sup, y=ggplot.confid[,2])) +
  xlab(paste0("T>",T))+ylab(paste0("confid.int_",sep=Min.list[iter.plot,4]))+
  geom_errorbar(aes(ymin=ggplot.confid[,1], ymax=ggplot.confid[,3]), width=.1)+
  geom_point(),
  file=paste0("int_conf_min_",T,"_", iter.plot,".png"))
  iter.plot=iter.plot+1
  ggplot.confid=ggplot.confid[,-c(1:3)]
}

}
```

REFERENCES

. Barmparis, G.D. and Tsironis, G.P. (2020). Estimating the infection horizon of COVID- 19 in eight countries with a data-driven approach, Chaos, Solitons and Fractals, **135**, 109842.
. Bloomfield, P. and Steiger, W. L. (1980). Least Absolute Deviations Curve-Fitting. *SIAM J. Sci. Stat. Comput.*, **2**, 290–301.
. Dielman, T., E. (2005). Least absolute value regression: recent contributions. *J. Stat. Comput. Simul.*, **75**, 263–286.
. Eisenhart,C. (1961). Boscovitch and the combination of observations. *In Roger Joseph Boscovitch, ed. L.L. Whyte. New York: Fordham University Press.*
. Gao, F. and Lixing H. (2010). Implementing the Nelder-Mead simplex algorithm with adaptive parameters. *Comput. Optim. Appl.*, **51**, 259–277.
. Karst, O. J. (1958). Linear curve fitting using least deviations. *J. Amer. Stat. Assoc.*, **53**, 118–132.
. Koenker,R. and Bassett, G. (1985). On Boscovich's estimator. *Ann. Statist.*, **13**, 1625–1628.
. Lagarias, J. C., Reeds, J. A, Wright, M.W and Wright, P. E. (1998). Convergence properties of the Nelder–Mead simplex method in low dimensions. *SIAM J. Optim.*, **9**, 112–147.
. Li, Y.; Arce, G. R. (2004). A maximum likelihood approach to least absolute deviation regression. *EURASIP J. Appl. Signal Process.* **12**, 1762–1769.
. Narula, S.C. and Wellington, J.F. (1977). Algorithm AS 108. Multiple linear regression with minimum sum of absolute errors. *J. R. Stat. Soc. Ser. C. Appl. Stat.*, **26**, 106-111.
. Nelder J. A and Mead R. (1965). A simplex method for function minimization. *Comput. J.*, **7**, 308–313.
. Novoselac, V. (2020). The Component Weighted Median Absolute Deviations Problem. *Eur. J. Pure Appl. Math.*, **13**, 964–976.
. Rhodes, E. C. (1930). Reducing observations by the method of minimum deviations. *Phil Mag. Seventh Series*, **9**, 974-992.
. Robert, C. (1995). Simulation of truncated normal variables. *Stat. Comput.*, **5**, 121-125.
. Sadovski, A. N. (1974). Algorithm As 74. L1-norm fit of straight line. *J. R. Stat. Soc. Ser. C. Appl. Stat.*, **23**, 244–248.