

RSBN: Regression with Stochastically Bounded Noises

Xiaoming Huo* and Xuelei Ni

*School of Industrial & Systems Engineering, Georgia Institute of Technology,
Atlanta, GA 30332-0205, USA.*

Abstract

We consider M-estimates in a regression model where the noises are of unknown but stochastically bounded distribution. An asymptotic minimax M-estimate is derived. Simulations demonstrate the robustness of this approach, as well as advantages over commonly used estimates such as the ordinary least square estimate and the Huber's estimate. The new method is named *regression with stochastically bounded noises* (RSBN). We provide an iterative numerical solution, which is derived from the proximal point method. The iterative method is elegant, however may not have fast rate of convergence. RSBN can also be solved by applying existing state-of-the-art nonlinear optimization software. We present SNOPT as one example. Insights from RSBN are discussed.

Key words: Regression, Asymptotic Minimax Estimate, M-estimate, Proximal Point Method

* Corresponding author. Email: xiaoming@isye.gatech.edu.

1 Introduction

We consider a regression problem when the noise distribution is unknown, but some probabilistic information is available. More specifically, we consider the cases when the noises are stochastically bounded: there exist constants $\delta > 0$ and $0 < \alpha < 1$, such that $\text{Prob.}\{|\text{noise}| > \delta\} < \alpha$. In a regression framework, we derive the asymptotic minimax estimate of the coefficients for all noise distributions satisfying the above constraint.

Interesting similarity between the derived minimax estimate and some recently emerged criterion functions in model selection is inspiring. Specifically, the fact that the objective function become linear outside a neighborhood of the origin coincides with the ℓ_1 -norm principle that has recently gained popularity via methods such as LASSO (Tibshirani, 1996) and Basis Pursuit (Chen, Donoho, and Saunders, 2001).

RSBN can be viewed as an extension of the well-developed Huber's M-estimate. Hence, it is a development in the line of robust statistics. We found that by deriving the exact form of the asymptotic minimax estimate of the coefficients, we can achieve better numerical performance. Simulations on synthetic data are reported to demonstrate our findings.

Using the proximal point method in optimization, we develop an iterative approach that is extremely simple to implement — it takes a few lines in MATLAB. However, its numerical performance is not stable: it can converge extremely fast in some situations, and extremely slow in some pathological cases. We give our analysis on the speed of convergence in some situations. We also present an alternative: using existing state-of-the-art optimization software packages, e.g., SNOPT.

In this paper, Section 2 presents the formulation and the main theoretical result. Section 3 establishes the asymptotic minimaxity of the proposed estimate. Section 4 describes the numerical algorithm that is derived from the proximal point method. Related analysis on the convergence of this algorithm is presented. Section 5 presents an alternative numerical approach, which utilizes a state-of-the-art but commercialized optimization software. Simulations that consolidate our findings are presented in Section 6. Discussions and

Conclusions are presented in Section 7 and Section 8, respectively.

2 Formulation and Main Theoretical Result

A regression model is

$$\mathbf{y} = A\mathbf{x} + \varepsilon, \quad (2.1)$$

where $\mathbf{y} = (y_1, y_2, \dots, y_n)^T \in \mathbf{R}^n$ is the response vector, $\mathbf{x} \in \mathbf{R}^m$ is a vector of coefficients, model matrix is $A = [a_1, a_2, \dots, a_n]^T \in \mathbf{R}^{n \times m}$, and a random error vector is $\varepsilon = (\varepsilon_1, \varepsilon_2, \dots, \varepsilon_n)^T$. Without loss of generality, we assume that model matrix A is of full column rank (equivalently, matrix inverse $(A^T A)^{-1}$ exists). Furthermore, we assume that the random errors $\varepsilon_i, i = 1, 2, \dots, n$, are i.i.d. with a common density function f .

Given a set of coefficients \mathbf{x} , the residual associated with the i th response is $r_i = y_i - a_i^T \mathbf{x}$. One can estimate the set of coefficients by solving the following optimization problem:

$$\begin{aligned} & \text{minimize} \quad \sum_{i=1}^n \rho(r_i), \\ & \text{subject to} \quad r_i = y_i - a_i^T \mathbf{x}, i = 1, 2, \dots, n. \end{aligned} \quad (2.2)$$

Here, we normally require function ρ to be convex, because convex optimization problem in principle is much more amenable than other optimization problems (e.g., combinatoric optimization problems). If we define a residual vector $\mathbf{r} = (r_1, r_2, \dots, r_n)^T \in \mathbf{R}^n$, the restriction of the above optimization problem can be rewritten as $\mathbf{r} = \mathbf{y} - A\mathbf{x}$. Another way to express the optimization problem in (2.2) is:

$$\begin{aligned} & \text{minimize} \quad \rho(\mathbf{r}) = \sum_{i=1}^n \rho(r_i), \\ & \text{subject to} \quad \mathbf{r} = \mathbf{y} - A\mathbf{x}. \end{aligned}$$

A key feature of the above formulation is that the criterion function (which is also the objective) is an additive function with respect to the residuals r_i 's. The criterion depicted in (2.2) covers many known approaches. For example, when $\rho(x) = x^2$, we have

the ordinary least square estimate.

We consider the situation when the random i.i.d. errors ε_i 's satisfy the following condition.

Condition 2.1 (stochastically bounded noises) *In a regression model, if for $1 \leq i \leq n$, we have*

$$Prob.(|\varepsilon_i| > \delta) \leq \alpha,$$

where $\delta > 0$ and $0 < \alpha < 1$ are predetermined, then we have stochastically bounded noises.

In this paper, we propose the following function for $\rho(x)$:

$$\rho(x) = \begin{cases} -\log \cos \lambda_1(x/\delta), & \text{if } |x/\delta| < 1; \\ \lambda_1 \tan \lambda_1 \cdot |x/\delta| - \lambda_1 \tan \lambda_1 - \log \cos \lambda_1, & \text{if } |x/\delta| \geq 1. \end{cases} \quad (2.3)$$

where $0 < \lambda_1 < \pi/2$ is a function of α . The analytic relation between λ_1 and α will be established when we derive the asymptotic minimaxity of the above estimate. Figure 1 gives a graphical comparison between the above ρ and the objective functions that are used in the least square estimate and the Huber's M-estimate.

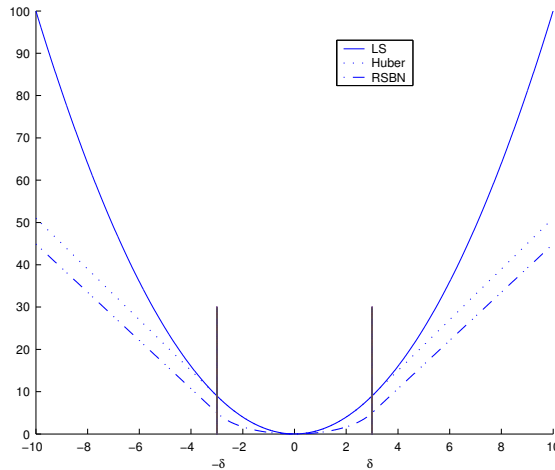


Fig. 1. Objective function $\rho(x)$ in ordinary least square, Huber's M-estimate, and RSBN.

When the function $\rho(x)$ has the form in (2.3), the obtained estimate is called a *regression with stochastically bounded noise* (RSBN) estimate. With our choice of ρ , problem (2.2) turns into a nonlinear optimization problem. The main reason to choose the function ρ in (2.3) is the following theorem.

Theorem 2.2 *Under the ‘stochastically bounded noises’ condition, the estimate from (2.2) with the function ρ specified in (2.3) is the asymptotic local minimax estimate of the coefficient vector \mathbf{x} .*

The above theorem will be established in the next section. Note we proved local minimaxity, instead of global minimaxity. Distinction between the two will be discussed in Section 7.5.

3 Regression Achieving Asymptotic Minimavity

Theoretical foundation of RSBN will be presented in the following subsections.

- Asymptotic normality (Section 3.1): we establish that the solution to (2.2) is asymptotically normal.
- Minimum asymptotic variance estimation (Section 3.2): we derive the estimate that achieves the minimum asymptotic variance.
- Least informative distribution (Section 3.3): we study the worst case in estimation, which is equivalent to finding the least informative distribution. By doing so, we get a locally asymptotic minimax estimate.
- Regression with stochastically bounded noises (RSBN) (Section 3.4 and 3.5): we present our regression method, by specifying the function $\rho(\cdot)$ in (2.2).
- Fisher information (Section 3.6) and asymptotic variance (Section 3.7): we derive the Fisher information for the least informative distribution and the asymptotic variance for the RSBN estimate.
- Robustness (Section 3.8): we consider the robustness of the estimate by specifying its breakdown point.

3.1 Asymptotic Normality

The solution to (2.2) is an M-estimate. In this section, we derive the asymptotic normality of an M-estimate.

We start with assumptions and notations. First, we consider location estimation. In (2.2), we temporarily assume that $m = 1$ and $a_i = 1, i = 1, 2, \dots, n$. Suppose ρ has the second derivative. Let $\psi = \rho'$ be the first derivative of ρ . Define a function $\lambda(t, F) = \int \psi(\xi - t)dF(\xi)$, for $t \in \mathbf{R}$, where F is the cumulative distribution function (c.d.f.) of random variable ξ . Define a functional \mathbf{T} from distribution space to \mathbf{R} , such that $\lambda(\mathbf{T}(F), F) = 0$. Value $\mathbf{T}(F)$ is defined as the true location parameter. Let F_n be the empirical c.d.f. Note that $\lambda(x, F_n) = 0$ is the first order necessary condition (FOC) for a minimizer of (2.2). It is easy to see that $\mathbf{T}(F_n)$, which satisfies $\lambda(\mathbf{T}(F_n), F_n) = 0$, is an M-estimate for n samples.

The asymptotic normality of $\mathbf{T}(F_n)$ is typically derived in the following three steps:

(1) Firstly, we have

$$\begin{aligned} 0 &= \lambda(\mathbf{T}(F), F) - \lambda(\mathbf{T}(F_n), F) + \lambda(\mathbf{T}(F_n), F) - \lambda(\mathbf{T}(F_n), F_n) \\ &= [\mathbf{T}(F) - \mathbf{T}(F_n)] \frac{\lambda(\mathbf{T}(F), F) - \lambda(\mathbf{T}(F_n), F)}{\mathbf{T}(F) - \mathbf{T}(F_n)} \\ &\quad - \frac{1}{n} \sum_{i=1}^n [\psi(y_i - \mathbf{T}(F_n)) - \lambda(\mathbf{T}(F_n), F)]. \end{aligned} \quad (3.4)$$

(2) We assume some regularity conditions are satisfied, and $\mathbf{T}(F_n) \rightarrow \mathbf{T}(F)$. (As long as ψ is monotone and $F_n \Rightarrow F$, which are generally satisfied conditions, $\mathbf{T}(F_n) \rightarrow \mathbf{T}(F)$ is true.) We have

$$\begin{aligned} \frac{\lambda(\mathbf{T}(F), F) - \lambda(\mathbf{T}(F_n), F)}{\mathbf{T}(F) - \mathbf{T}(F_n)} &\Rightarrow \frac{\partial}{\partial t} \lambda(t, F)|_{t=\mathbf{T}(F)} \\ &= \int \psi'(x - \mathbf{T}(F))dF(x). \end{aligned} \quad (3.5)$$

Since ρ has the second derivative, the derivative ψ' exists. The above also implies that the right hand side of (3.5) is integrable.

(3) Observe

$$\begin{aligned} & \frac{1}{n} \sum_{i=1}^n [\psi(y_i - \mathbf{T}(F_n)) - \lambda(\mathbf{T}(F_n), F)] \\ \Rightarrow & \frac{1}{\sqrt{n}} \text{Normal} \left(0, \int \psi^2(x - \mathbf{T}(F)) dF(x) \right). \end{aligned} \quad (3.6)$$

This is a direct result from central limit theorem (CLT), because the left hand side is a sum of i.i.d. random variables. We suppose to check the Lindeberg condition. In this paper, we assume the condition is satisfied. For more details, see Hampel, Ronchetti, Rousseeuw, and Stahel (1986).

Combining (3.4), (3.5) and (3.6), we have

$$\mathbf{T}(F_n) - \mathbf{T}(F) \sim \frac{1}{\sqrt{n}} N \left(0, \frac{\int \psi^2 dF}{(\int \psi' dF)^2} \right).$$

The asymptotic variance is equal to $\frac{\int \psi^2 dF}{(\int \psi' dF)^2}$.

The above result can be generalized to a multivariate case. When $m > 1$ and matrix A is of full column rank, the asymptotic variance/covariance matrix of an M-estimate will be $\frac{\int \psi^2 dF}{(\int \psi' dF)^2} (A^T A)^{-1}$. For reference, please see Chapter 7.6 in Huber (1981).

Lemma 3.1 *Given function $\rho(\cdot)$ that has a monotone first derivative $\psi = \rho'$ and whose second derivative is integrable in (3.5), the estimate given by (2.2) has the asymptotic distribution*

$$N \left(\mathbf{x}_0, \frac{1}{n} \frac{\int \psi^2 dF}{(\int \psi' dF)^2} (A^T A)^{-1} \right),$$

where the vector \mathbf{x}_0 is made of the true values of the coefficients.

We take the quantity $\frac{\int \psi^2 dF}{(\int \psi' dF)^2}$ as a natural measure of performance for an M-estimate. The smaller this quantity, the closer the M-estimate is to the true parameter.

3.2 Minimum Asymptotic Variance Estimation

We call quantity $\frac{\int \psi^2 dF}{(\int \psi' dF)^2}$ the *asymptotic variance*. It is known that the asymptotic variance is lower bounded by the inverse of Fisher information. The following analysis is

well-adopted in mathematical statistics.

Let $f_\theta = f(x - \theta)$ be the p.d.f. associated with c.d.f. F_θ and location parameter θ . $I(f)$ is the Fisher information with respect to θ . We have

$$\lambda(\theta, F_\theta) = \int \psi(x - \theta) f(x - \theta) dx = \text{constant}.$$

Taking the operator $\frac{\partial}{\partial \theta}$ on both sides, we get

$$0 = - \int \psi'(x - \theta) f(x - \theta) dx - \int \psi(x - \theta) f'(x - \theta) dx. \quad (3.7)$$

Here we assume both ψ and f are absolutely continuous and have first derivatives. From (3.7),

$$\begin{aligned} 1 &= \left[\int \left(\frac{\psi}{\int \psi' f} \right) \cdot \left(-\frac{f'}{f} \right) f \right]^2 \\ &\stackrel{\text{Cauchy}}{\leq} \int \left(\frac{\psi}{\int \psi' f} \right)^2 f \cdot \int \left(-\frac{f'}{f} \right)^2 f \\ &= \int \left(\frac{\psi}{\int \psi' f} \right)^2 f \cdot I(f), \end{aligned}$$

where $I(f)$ is the Fisher information of f . So asymptotic variance $\int \left(\frac{\psi}{\int \psi' f} \right)^2 f \geq \frac{1}{I(f)}$.

It achieves equality iff $\rho' = \psi \propto -\frac{f'}{f} = (-\log f)'$, in which case the M-estimate is also the maximum likelihood estimate (MLE). When $\rho = -\log f$, we call the solution to (2.2) the *minimum asymptotic variance estimate*. The result in this subsection is summarized as the following lemma.

Lemma 3.2 *The asymptotic variance of the estimate from (2.2) is lower bounded by $1/I(f)$. The lower bound is achieved when $\rho \propto (-\log f)$, i.e., when the estimate is the maximum likelihood estimate.*

3.3 Least Informative Distribution

The smaller the Fisher information $I(f)$ is, the larger is the lower bound of the asymptotic variance. We are interested in the least informative distribution, which is the solution to the following optimization problem: (note the variable is a function f)

$$\begin{aligned} & \text{minimize} && I(f), \\ & \text{subject to} && \int v(x)f(x)dx \leq 0, \\ & && \int f(x)dx = 1. \end{aligned} \tag{3.8}$$

Note in our framework, function f is assumed to have second derivative. Otherwise, a piecewise constant function f may lead to $I(f) = 0$, which leads to infinite asymptotic variance. Such a case is excluded by demanding the existence of the second derivative.

The first constraint is a general form of many types of restrictions on the noise distribution. For example, if

$$v(x) = \begin{cases} -\alpha, & |x| < \delta, \\ 1 - \alpha, & |x| \geq \delta, \end{cases} \tag{3.9}$$

we have $\int_{-\delta}^{\delta} f \geq 1 - \alpha$. This implies stochastically bounded noises. This condition is meaningful when there are outliers. If $v(x) = x^2 - B$, we have $\int x^2 f(x)dx \leq B$, which is the second moment constraint. Similarly, we can have some other moments constraints. The second constraint in (3.8) is the constraint of a p.d.f.

To find the solution to (3.8), we consider the following function:

$$\mu(f) = I(f) + \beta_1 \left[\int v(x)f(x)dx + \gamma^2 \right] + \beta_2 \left[\int f(x)dx - 1 \right],$$

where β_1 and β_2 are the Lagrange multipliers, and $\gamma \in \mathbf{R}$ is a pseudo-variable: $\int v(x)f(x)dx + \gamma^2 = 0$. We consider a variational approach. Assume function f_0 is a minimizer in (3.8).

For any other p.d.f. f_1 , consider $f_t = (1-t)f_0 + tf_1$, $0 \leq t \leq 1$. Because f_0 is a minimizer, we must have $\frac{d}{dt}\mu(f_t)|_{t=0} \geq 0$ for any f_1 , which is equivalent to

$$-4 \int \frac{(\sqrt{f_0})''}{\sqrt{f_0}}(f_1 - f_0)dx + \beta_1 \int v \cdot (f_1 - f_0)dx + \beta_2 \int (f_1 - f_0)dx \geq 0.$$

The above holds if and only if

$$4 \frac{(\sqrt{f_0})''}{\sqrt{f_0}} - \beta_1 \cdot v - \beta_2 = 0. \tag{3.10}$$

Note the above is a necessary condition for f_0 to be the solution to (3.8).

Lemma 3.3 *If a function f_0 has second derivative and achieves a local minimum in (3.8), then it satisfies the equation (3.10).*

In the next subsection, we construct a function f_0 that satisfies (3.10). This constructed function f_0 leads to the objective function that is used in RSBN.

3.4 Regression with Stochastically Bounded Noises (RSBN)

Recall our objective is to find an appropriate ρ in (2.2), so that the solution to (2.2) is both easy to compute and optimal within a family of distributions for random errors.

In our construction, the following conditions are satisfied.

- [Conditions for probability density function] Function f is a probability density function. Function f is from real numbers to nonnegative real numbers $f : \mathbf{R} \rightarrow \mathbf{R}^+$ ($f \geq 0$) and $\int f = 1$. In previous discussion, we implied that function f has finite Fisher information, $I(f) < \infty$. We also assume that the density function f is symmetric about 0.
- [Conditions for stochastically bounded noises] We have $\int_{-\delta}^{\delta} f \geq 1 - \alpha$. This means that the probability of noises having absolute values no larger than δ is at least $1 - \alpha$. Usually α is small. It is equivalent to say that no more than proportion α of noises can

have absolute values greater than δ . As mentioned earlier, an equivalent expression of this condition is $\int v(x)f(x)dx \leq 0$, where function v is defined in (3.9).

- [Conditions for convexity] The function $\rho(x) = -\log f_0(x)$ must be convex, otherwise we will not have a convex optimization problem. The first derivative of ρ , ρ' , exists and has first derivative as well. Complying with these, (2.2) becomes a nonlinear convex optimization problem.
- [Conditions for minimaxity] When $\rho(x) = -\log f(x)$, according to Lemma 3.2, the minimum asymptotic variance is achieved. If density f also minimize the objective in (3.8), the minimum variance is achieved in the worst scenario. Such an estimate is called an asymptotic minimax estimate. From Lemma 3.3, the above mentioned minimizer f should satisfy equation (3.10).

Readers can verify that the following function is a solution to equation (3.10).

$$f_0(x) = \begin{cases} c \left[\cos \lambda_1 \frac{x}{\delta} \right]^2, & |x| < \delta, \\ c \cdot \exp\left(-2\lambda_2 \frac{|x|}{\delta}\right) \cdot \cos^2 \lambda_1 \cdot \exp(2\lambda_2), & |x| \geq \delta, \end{cases} \quad (3.11)$$

where $0 < \lambda_1 < \frac{\pi}{2}$, $\lambda_2 > 0$. The above is constructed by considering the general solutions to the differential equation (3.10). One of the simplest form that satisfies all the aforementioned conditions is chosen. Special care is given to ensure that $\log(f_0)$ has second derivative, as readers will see later. More discussion regarding our choice of function f_0 , especially how it differentiates from Huber's estimator, will be provided in Section 7.2.

Recall $\rho = -\log f_0$, we have

$$\rho(x) = \begin{cases} -\log c - 2 \log \cos \lambda_1 \frac{x}{\delta}, & |x| < \delta, \\ -\log c + 2\lambda_2 \frac{|x|}{\delta} - 2\lambda_2 - 2 \log \cos \lambda_1, & |x| \geq \delta. \end{cases} \quad (3.12)$$

Note $\rho(x)$ can be simplified without changing the optimization problem in (2.2): i.e., replacing $\rho(x)$ with $a\rho(x) + b$, $a > 0$ in (2.2) gives an equivalent optimization problem. Note that $\rho(x)$ is linear outside the interval $[-\delta, \delta]$.

3.5 Parameters in RSBN

The parameters $c, \delta, \alpha, \lambda_1, \lambda_2$ satisfy the following conditions:

$$\int_{-\delta}^{\delta} f_0(x) dx = 1 - \alpha; \quad (3.13)$$

$$\lim_{x \rightarrow \delta^+} f'(x) = \lim_{x \rightarrow \delta^-} f'(x);$$

or equivalently,

$$\lim_{x \rightarrow \delta^+} \rho'(x) = \lim_{x \rightarrow \delta^-} \rho'(x); \quad (3.14)$$

$$\int_{\delta}^{+\infty} f_0(x) dx = \frac{\alpha}{2}. \quad (3.15)$$

From (3.14), we have

$$\lambda_2 = \lambda_1 \tan \lambda_1. \quad (3.16)$$

From (3.13) and (3.15), we have,

$$1 - \alpha = c \int_{-\delta}^{\delta} \left[\cos \lambda_1 \frac{x}{\delta} \right]^2 dx$$

$$= \frac{c\delta}{2} \left(\frac{1}{\lambda_1} \sin 2\lambda_1 + 2 \right); \quad (3.17)$$

$$\frac{\alpha}{2} = c \cdot \cos^2 \lambda_1 \cdot \exp(2\lambda_2) \int_{\delta}^{+\infty} \exp\left(-2\lambda_2 \frac{x}{\delta}\right) dx$$

$$= \frac{c\delta}{2} [\cos \lambda_1]^2 \frac{1}{\lambda_2}, \quad (3.18)$$

respectively. From (3.17), (3.18) and (3.16), we have

$$\frac{\alpha}{1 - \alpha} \stackrel{(3.17), (3.18)}{=} \frac{\frac{1}{\lambda_2} \cdot \cos^2 \lambda_1}{1 + \frac{1}{2\lambda_1} \sin 2\lambda_1}$$

$$\stackrel{(3.16)}{=} \frac{\frac{1}{\lambda_1} \cos^3 \lambda_1 / \sin \lambda_1}{1 + \frac{1}{2\lambda_1} \sin 2\lambda_1}$$

$$= \frac{\cos^3 \lambda_1}{\lambda_1 \cdot \sin \lambda_1 + \sin^2 \lambda_1 \cdot \cos \lambda_1}.$$

Hence,

$$\alpha = \frac{\cos^3 \lambda_1}{\lambda_1 \cdot \sin \lambda_1 + \cos \lambda_1}. \quad (3.19)$$

Proposition 3.4 *The proportion α defined in the stochastically bounded noises condition and the parameter λ_1 in RSBN have a relation stated in (3.19).*

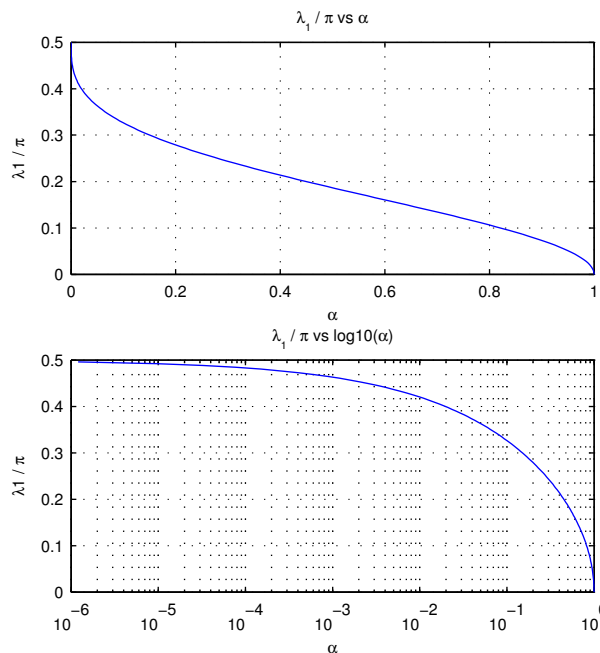


Fig. 2. Parameter λ_1 vs. α . The upper one is ordinary; the bottom takes $\log 10$ on α .

Figure 2 illustrates the relationship between α and λ_1 .

Now we consider a simplified version of (3.12). As an objective function in (2.2), the following ρ is equivalent to the one in (3.12).

$$\rho(x) = \begin{cases} -\log \cos \lambda_1 \frac{x}{\delta}, & |x| < \delta; \\ \lambda_2 \frac{|x|}{\delta} - \lambda_2 - \log \cos \lambda_1, & |x| \geq \delta. \end{cases} \quad (3.20)$$

Bringing in (3.16), we get exactly the expression in (2.3). Up to this point, we have established the Theorem 2.2.

We summarize the procedure of getting function ρ for RSBN. By some prior information, we know the values of α and δ . From (3.19), we can compute for λ_1 . From (3.16), we can compute for λ_2 . Substituting values λ_1 and λ_2 into (3.20), we have the close form formula for ρ . The following flow chart summarizes how to get ρ from α :

$$\alpha, \delta \xrightarrow{(3.19)} \lambda_1 \xrightarrow{(3.16)} \lambda_2 \xrightarrow{(3.20)} \rho.$$

3.6 Fisher Information of the Least Informative Distribution

We consider two important quantities associated with RSBN: Fisher information and asymptotic variance. For Fisher information, we give a close form solution with respect to λ_1 . Recall we have the relationship between λ_1 and α in (3.19), hence we know the relationship between the Fisher information and α . For the asymptotic variance, we describe how to compute it in a general case.

We start with the Fisher information $I(f_0)$. We consider location estimation. Let $f_\theta = f_0(x - \theta)$, where f_0 is the least informative distribution in Section 3.4. We have

$$I(f_0) = 4 \frac{\lambda_1^2}{\delta^2} \frac{\lambda_1 \cdot \sin \lambda_1}{\lambda_1 \cdot \sin \lambda_1 + \cos \lambda_1}. \quad (3.21)$$

The details in validating the above equation is postponed to Appendix A. Taking $\delta = 1.0$ and combining (3.19) and (3.21), we have the relationship between the Fisher information $I(f_0)$ and α . Since $\lambda_1 \in [0, \frac{\pi}{2}]$, the range of Fisher information $I(f_0)$ is from 0 to π^2/δ^2 . Figure 3 shows the relationship between α and the Fisher information $I(f_0)$. It is easy to find that small α leads to large Fisher information.

3.7 Asymptotic Variance of RSBN

As for asymptotic variance, in Section 3.1 we have already known that

$$\text{asymptotic variance} = \frac{\int \psi^2 dF}{(\int \psi' dF)^2}. \quad (3.22)$$

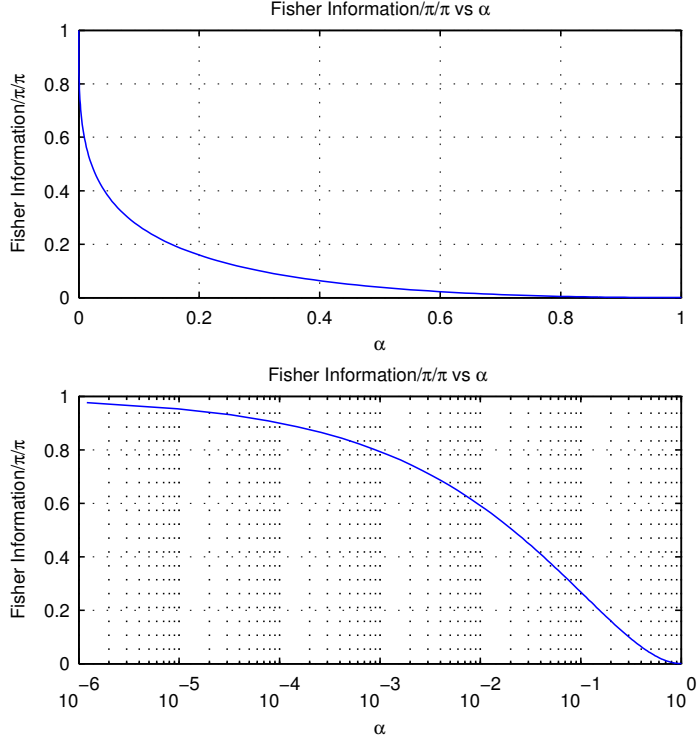


Fig. 3. Fisher information $I(f_0)$ versus α . The upper one takes ordinary coordinates; the lower takes $\log 10$ on α .

Since $\psi = \rho'$, we have

$$\psi(x) = \rho'(x) \stackrel{(3.20)}{=} \begin{cases} \frac{\lambda_1}{\delta} \tan \lambda_1 \frac{x}{\delta}, & |x| < \delta, \\ \text{sign}(x) \cdot \frac{\lambda_1}{\delta} \tan \lambda_1, & |x| \geq \delta; \end{cases} \quad (3.23)$$

and

$$\psi'(x) = \begin{cases} \frac{\lambda_1^2}{\delta^2} \sec \lambda_1 \frac{x}{\delta}, & |x| < \delta, \\ 0, & |x| \geq \delta. \end{cases}$$

Note ψ' is no longer continuous. As long as the noise has probability density function f that makes (3.22) meaningful, we can compute the asymptotic variance of the RSBN estimate.

3.8 Robustness

We now consider the robust property of RSBN. We compute the *breakdown* point—the maximum proportion of observations that can be arbitrarily distorted, while the estimate still does not “blow up” (i.e., not going to $\pm\infty$).

On page 16 in Huber (1977), we know that

$$\text{breakdown point} = \epsilon^* = \frac{\eta}{1 + \eta},$$

where $\eta = \min \left\{ -\frac{\psi(-\infty)}{\psi(+\infty)}, -\frac{\psi(+\infty)}{\psi(-\infty)} \right\}$. From the formula of ψ in the last section, we have $\eta = 1$. Hence $\epsilon^* = 1/2$, which is the largest breakdown point we can have for M-estimates.

Lemma 3.5 *The breakdown point of the RSBN estimate is 1/2.*

4 Numerical Algorithm: Proximal Point Method

In this subsection, we describe a proximal point algorithm. The purpose is to give readers who may not have access to a sophisticated software package an extremely–easy–to–use algorithm.

The rest of this section is organized as follows. Section 4.1 describes the general idea of a proximal point method. The RSBN can be formulated as a *partial inverse problem*, which is described in Section 4.2. Section 4.3 describes how to solve a partial inverse problem. An algorithm that solves RSBN is provided in Section 4.4. Some analysis regarding the convergence speed of the proposed algorithm is presented in Section 4.5.

4.1 General Idea

The proximal point algorithm solves the following problem:

$$\begin{aligned} \text{Find } \mu \in \mathbf{R}^n : 0 = \mathbf{U}(\mu), \\ \text{where } \mathbf{U} : \mathbf{R}^n \rightarrow \mathbf{R}^n \text{ is an operator.} \end{aligned} \tag{4.24}$$

The proximal point algorithm includes two steps:

Algorithm to Solve $0 = \mathbf{U}(\mu)$.

(1) **Choose** $\mu^{(0)}, n = 0$.

(2) **Repeat**

$$\begin{aligned} \mu^{(n+1)} &= (\mathbf{I} + \mathbf{U})^{-1} \mu^{(n)}, \\ n &= n + 1, \end{aligned}$$

Until convergence.

Here \mathbf{I} is the identity operator, and $(\mathbf{I} + \mathbf{U})^{-1}$ is the inverse of operator $(\mathbf{I} + \mathbf{U})$. The following results are known (Spingarn, 1983).

- Let μ^0 denote the solution to (4.24), i.e. $0 = \mathbf{U}(\mu^0)$. If $\{\mu^{(n)}\}$ converges, then it converges to μ^0 .
- If \mathbf{U} is a monotone operator in \mathbf{R}^n , then $(\mathbf{I} + \mathbf{U})^{-1}$ is well defined. (Operator \mathbf{U} is a monotone operator if for any $x_1, x_2 \in \mathbf{R}^n$, the inner product $\langle x_1 - x_2, \mathbf{U}(x_1) - \mathbf{U}(x_2) \rangle \geq 0$.)
- If \mathbf{U} is a monotone operator in \mathbf{R}^n , then $\{\mu^{(n)}\}$ converges.

4.2 Partial Inverse

Problem (2.2) can be cast as a *partial inverse problem*. Suppose A is a subspace of \mathbf{R}^n , $A \subset \mathbf{R}^n$ and B is the perpendicular compliment of A , $B = A^\perp$. The *partial inverse problem* is:

$$\text{find } x, y \in \mathbf{R}^n : \begin{cases} x \in A, \\ y \in B, \\ y = \mathbf{U}(x). \end{cases} \quad (4.25)$$

If \mathbf{U} is strictly monotone, the solution of the *partial inverse* is unique.

Problem (4.25) can be formulated as (4.24). Suppose $x, y \in \mathbf{R}^n$ have decomposition:

$$x = x_A + x_B, \quad y = y_A + y_B,$$

where $x_A, y_A \in A$ and $x_B, y_B \in B$. We define a new operator \mathbf{U}_A , such that $x_B + y_A = \mathbf{U}_A(x_A + y_B)$ if and only if $y = y_A + y_B = \mathbf{U}(x_A + x_B) = \mathbf{U}(x)$. Suppose z has a decomposition: $z = z_A + z_B$, where $z_A \in A, z_B \in B$. A general theorem says that (x, y) is the solution to (4.25) if and only if

$$\exists z : 0 = \mathbf{U}_A(z), \quad (4.26)$$

where $x = z_A, y = z_B$. By solving (4.26), we get an exact solution to (4.25). Note that (4.26) has the same form as (4.24).

4.3 Solving Partial Inverse

Based on (4.26) and the algorithm in Section 4.1, the key to solving a *partial inverse problem* is to find $(\mathbf{I} + \mathbf{U}_A)^{-1}$. Following the notations in Section 4.2, since $x_B + y_A = \mathbf{U}_A(x_A + y_B)$, we have $x + y = (\mathbf{I} + \mathbf{U}_A)(x_A + y_B)$. In other words, $x_A + y_B = (\mathbf{I} + \mathbf{U}_A)^{-1}(x + y)$. In order to solve $(\mathbf{I} + \mathbf{U}_A)^{-1}(u)$, if we can find (x, y) satisfying

$$\begin{cases} u = x + y, \\ y = \mathbf{U}(x), \end{cases}$$

then $(\mathbf{I} + \mathbf{U}_A)^{-1}(u) = x_A + y_B$. Since $u = x + y = (\mathbf{I} + \mathbf{U})(x)$, we have

$$\begin{cases} x = (\mathbf{I} + \mathbf{U})^{-1}(u), \\ y = u - x. \end{cases}$$

Now we have the algorithm to solve $(\mathbf{I} + \mathbf{U}_A)^{-1}$.

Algorithm to Solve $(\mathbf{I} + \mathbf{U}_A)^{-1}$.

- Find x , so that $x = (\mathbf{I} + \mathbf{U})^{-1}(u)$;
- Let $y = u - x$;
- $(\mathbf{I} + \mathbf{U}_A)^{-1}(u) = x_A + y_B$.

Note this is a general method to solve (4.25). If $(\mathbf{I} + \mathbf{U})^{-1}$ is easy to implement, then $(\mathbf{I} + \mathbf{U}_A)^{-1}$ is easy to implement.

4.4 Application to RSN

Now we apply the previously developed method to RSN. Consider the first order necessary condition of (2.2), we have

$$0 = A^T \psi(Ax - y), \tag{4.27}$$

where $\psi = \rho'$, $\psi(y - Ax) = [\psi((y - Ax)_1), \psi((y - Ax)_2), \dots, \psi((y - Ax)_n)]^T$, $(y - Ax)_i$ denotes the i th component of vector $y - Ax$, and ψ is defined in (3.23). Equation (4.27) is equivalent to

$$\text{find } u, v : \begin{cases} u = Ax, \\ v = \psi(u - y), \\ 0 = A^T v. \end{cases} \quad (4.28)$$

In other words,

$$\text{find } u, v : \begin{cases} u \in \text{Range}(A), \\ v \in \text{Kernel}(A), \\ v = \psi(u - y). \end{cases}$$

Following the algorithm in Section 4.3, we have

Algorithm for RSBN

- (1) **Choose** $\mu^{(0)} \in \mathbf{R}^n, k = 0$.
- (2) **Find** u_i , such that $\psi(u_i - y_i) + u_i = \mu_i^{(k)}, i = 1, 2, \dots, n$.
- (3) **Let** $v_i = \mu_i^{(k)} - u_i, i = 1, 2, \dots, n$.
- (4) **Project** $u = (u_1, \dots, u_n)^T, v = (v_1, \dots, v_n)^T$.

$$\begin{aligned} \mu^{(k+1)} &= \mathbf{P}_A(u) + \mathbf{P}_{\text{Kernel}(A)}(v) \\ &= v + A(A^T A)^{-1} A^T (u - v). \end{aligned}$$

Here \mathbf{P}_A and $\mathbf{P}_{\text{Kernel}(A)}$ are projection operators to subspaces range of A and kernel of A respectively.

- (5) If not converge, $k = k + 1$, go back to step (2).

In step (2), since ψ in (3.23) is monotone increasing, x_i will have a unique solution. But because there is a tangent function in ψ in RSBN, one needs to implement a line search algorithm to solve it. We can see that if function ψ is piecewise polynomial, this method

is quite appealing, because a close form solution is available to the equation in step (2). This approach has been used in solving Huber's M-estimate, see Michelot and Bougeard (1994).

After getting u , the x can be solved via $u = Ax$. Recall matrix A is of full column rank.

4.5 Analysis

It is possible that the above mentioned algorithm converges slowly to the solution. We construct such an example. Assume the projection matrix associated with \mathbf{P}_A is, for $d < n$,

$$\begin{pmatrix} \mathbf{I}_d & 0 \\ 0 & 0 \end{pmatrix}_{n \times n},$$

where \mathbf{I}_d is a d by d eye matrix. The regression projects to the first d coordinates. Following the notations in Table ??, we have

$$\mu_i^{(k+1)} = \begin{cases} u_i, & 1 \leq i \leq d, \\ v_i, & 1 + d \leq i \leq n. \end{cases}$$

Restricted to $1 \leq i \leq d$, we have

$$|\mu_i^{(k+1)} - \mu_i^{(k)}| = |u_i - \mu_i^{(k)}| = |\psi(u_i - y_i)| \leq \frac{\lambda_1}{\delta} \tan \lambda_1,$$

where the last term is a constant, the second equality is based on the step 2. in the RSBN algorithm, and the inequality is based on (3.23). If u^0 is the solution of the RSBN, assuming that we started with an all zero vector $\mu^{(0)} = (0, 0, \dots, 0)^T$, the proposed proximal point algorithm takes at least

$$\frac{\max_{1 \leq i \leq d} |u_i^0|}{\lambda_1 / \delta \tan \lambda_1}$$

steps to converge. Note the number of steps can be large, if the maximum entry $\max_{1 \leq i \leq d} |u_i^0|$ is large.

The reason that the proximal point approach can be slow is that it does not take advantage of high degree smoothness of the objective function. For example, it does not use the second derivatives. More efficient numerical solution can be developed by taking advantage of the existence of second derivatives. Most of state-of-the-art optimization software will do so automatically. We propose one alternative in the next section.

5 Other Implementation: SNOPT and SQP

As an alternative, we use some state-of-the-art optimization software to solve the RSBN *directly*. In this research, we use a general-purpose optimization package—SNOPT. It is a software package developed in Gill, Murray, and Saunders (1998). It minimizes a linear or nonlinear function subject to bounds on the variables, as well as sparse linear or nonlinear constraints. It is suitable for large-scale linear and quadratic programming and for linearly constrained optimization, as well as for general nonlinear programs. In our case, in (2.2), we have linear constraints and a nonlinear but convex objective function.

SNOPT finds a solution that is *local optimal*. Ideally, any nonlinear functions should be smooth and users should provide gradients. In our case, since the objective function in (2.2) is convex, the *local optimal* solution will coincide with the global optimal solution. For RSBN, the gradients are given in (3.23).

SNOPT uses a sequential quadratic programming (SQP) algorithm that obtains a search direction from a sequence of quadratic programming subproblems. Each QP subproblem minimizes a quadratic model of a certain Lagrangian function subject to a linearization of the constraints. An augmented Lagrangian merit function is reduced along each search direction to ensure convergence from any starting point.

The source code for SNOPT is written in Fortran. In order to use it, a Fortran compiler is required. The numerical examples in the present paper are a result of combining some MATLAB programming, Unix shell programming, Fortran programming, and SNOPT.

6 Simulation

6.1 An Illustrative Example: Variable Star

In this section, we study a well-known data set in the time series analysis — magnitudes of a variable star at midnight on 600 successive nights. Bloomfield (1976) showed that it is a superposition of two ‘dominant’ sinusoid functions. We are taking a slightly different viewpoint. We assume that the underlying signal (denoted by \mathbf{s}) is a smooth signal residing in a low dimensional linear subspace. The observed magnitudes denoted by \mathbf{y} , $\mathbf{y} = (y_1, y_2, \dots, y_{600})^T$, are an approximation to s . In our case, \mathbf{y} is the rounded version of s : i.e., $y_i = [s_i + 0.5]$, where $[x]$ is the largest integer no larger than x . It is evident that the mapping from \mathbf{s} to \mathbf{y} is completely nonlinear. Let $\mathbf{y} = \mathbf{s} + \mathbf{n}$, where \mathbf{n} is the so-called noise sequence. Considering the source where the noise sequence is generated, the Gaussian assumption on the distribution of \mathbf{n} is not appropriate. We assume that the subspace, on which the signal resides, is known to us. In this case, we compare ordinary least square estimate, Huber’s estimate, and RSBN.

We consider the discrete cosine transform (DCT). The DCT with signal length n has the k th basis function:

$$c_k(i) = \begin{cases} \sqrt{1/n}, & k = 0, i = 1, 2, \dots, n; \\ \sqrt{2/n} \cos[(i - \frac{1}{2})k\frac{\pi}{n}], & k \neq 0, i = 1, 2, \dots, n. \end{cases}$$

The reasons of choosing DCT are: (a) DCT is a real analogous of the Fourier transform, which is widely adopted in representing cyclic signals; (b) there are fast numerical algorithms to implement DCT.

First, we study the original variable star data set. We find the subspace that contains most of the signal’s energy. This can be done by carrying out a DCT transform, then retaining the coefficients with the largest amplitudes. Later, we intentionally distort the

observation. Three different ways of projection are then compared. We illustrate the optimality of our method.

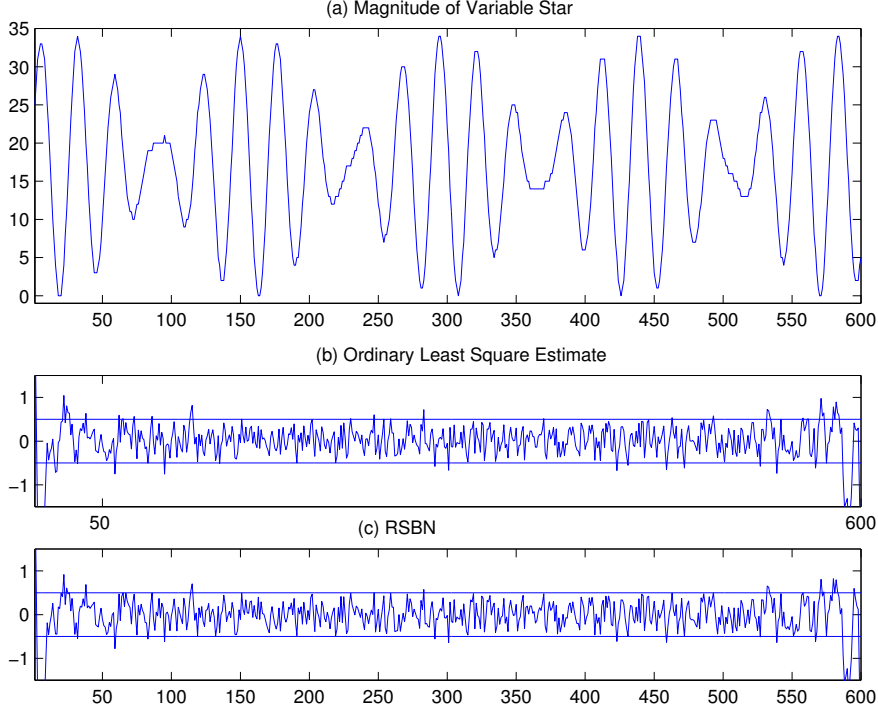


Fig. 4. Dataset Variable Star and deviations by the least square regression and RSBN.

Figure 4 (a) shows the magnitude vector \mathbf{y} of the variable star. We take a DCT of \mathbf{y} , keep the 10% of coefficients that have the 10% largest amplitudes (of coefficients). The associated 10% basis functions span the subspace that contains the largest possible proportion of the energy. We denote the subspace by A . The dimension of A is 60. Projecting the observation \mathbf{y} to A by the ordinary least square regression, we have $\mathbf{P}_{A,LS}(\mathbf{y}) = \hat{s}_{LS,1}$, where subscript ‘LS’ indicates least square estimate, and ‘1’ indicates for original observation \mathbf{y} . (In other words, vector $\hat{s}_{LS,1}$ minimizes $\|\mathbf{u} - \mathbf{y}\|_{\ell_2}^2$ among all vectors \mathbf{u} that are in the linear subspace spanned by columns of matrix A .) We then project the observation \mathbf{y} to A by using RSBN. We choose $\delta = 0.5$, $\lambda_1 = 0.46\pi$ and ρ is given in (3.20). We denote $\mathbf{P}_{A,RSBN}(\mathbf{y}) = \hat{s}_{RSBN,1}$, where subscript ‘RSBN’ indicates a RSBN estimate. Since the deviations, $\mathbf{y} - \hat{s}_{RSBN,1}$ (shown in Figure 4 (c)), are supposed to be round off errors, ideally they should be within the interval $[-0.5, 0.5]$. For the least square estimate, there are 70 deviations (shown in Figure 4 (b)) having amplitudes larger than 0.5, and 16 of

them having amplitudes larger than 1.0. For RSBN, there are 44 deviations having amplitudes larger than 0.5, and 15 of them having amplitude larger than 1.0. In this case, compared to the ordinary least square estimate, the RSBN has less deviations falling beyond the ideal interval $[-0.5, 0.5]$. Of course, at the same time, we should observe a loss in the mean square error, which is what a least square approach tries to minimize. The sum of squares of deviations in the least square estimate is 10.4337, and the one for RSBN is 10.6022.

Now we randomly pick up two positions in the variable star sequence. In particular, we choose position 224 and 446. Originally, $y_{224} = 15$ and $y_{446} = 16$. Suppose the decimal points in these numbers were somehow misspecified. The recorded values become $y'_{224} = 1.5$ and $y'_{446} = 160$. Without loss of generality, let \mathbf{y}' denote the new sequence. Figure 5 shows \mathbf{y}' .

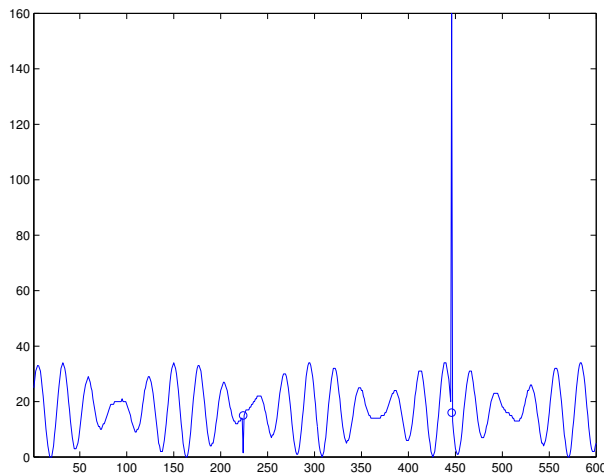


Fig. 5. The distorted variable star signal. Circles indicate the true values.

Recall $\mathbf{P}_{A,LS}$ and $\mathbf{P}_{A,RSBN}$ denote the projection operators to subspace A by least square estimate and RSBN, respectively. Let $\mathbf{P}_{A,H}$ denote a projection operator to A via Huber's M-estimate. Recall that a Huber's M-estimate is for $\Delta > 0$, choose

$$\rho(x) = \begin{cases} x^2, & |x| < \Delta, \\ 2\Delta|x| - \Delta^2, & |x| \geq \Delta, \end{cases}$$

in (2.2) (Huber, 1977, 1981). In Huber’s estimate, the function ρ is piecewise linear (outside a neighborhood of the origin) or quadratic (inside a neighborhood of the origin).

Consider the projections $\hat{s}_{LS,2} = \mathbf{P}_{A,LS}(\mathbf{y}')$, $\hat{s}_{RSBN,2} = \mathbf{P}_{A,RSBN}(\mathbf{y}')$, and $\hat{s}_{H,2} = \mathbf{P}_{A,H}(\mathbf{y}')$. The deviations $\mathbf{y} - \hat{s}_{LS,2}$, $\mathbf{y} - \hat{s}_{RSBN,2}$ and $\mathbf{y} - \hat{s}_{H,2}$ are plotted in Figure 6 (a), (b), and (c). Note these are the deviations from the estimates to the “original” signal sequence \mathbf{y} (not \mathbf{y}'). Table 1 shows some statistics on the performance of three different methods.

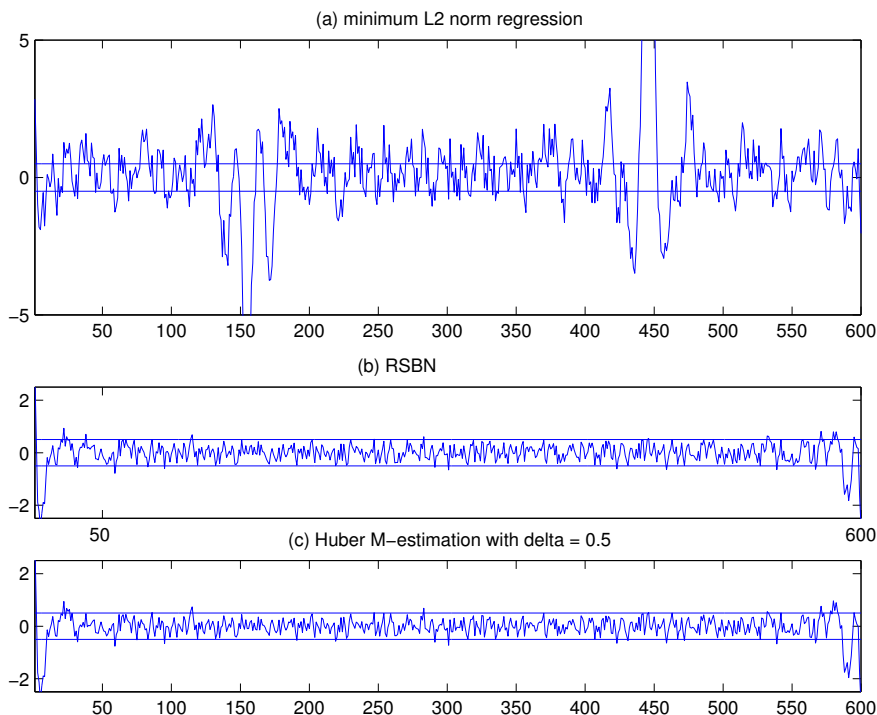


Fig. 6. Differences between the original variable star signal and the estimates from the distorted signal by three methods.

There are several phenomena noteworthy. First of all, the deviation of the least square estimate is significantly worse than the other two. This illustrates that least square estimate is not a robust method. Second, the performance of RSBN has almost no difference between the two cases: \mathbf{y} and \mathbf{y}' . In other words, $\hat{s}_{RSBN,2}$ is as close to \mathbf{y} as $\hat{s}_{RSBN,1}$ is. Third, RSBN performs nearly as well as the Huber’s M-estimate. RSBN is slightly better. It is not surprising that the performances of RSBN and Huber’s M-estimate are close, because the objective functions in (2.2) for these two are very close to each other.

Table 1

Some statistics for three regression methods for the distorted variable star data.

	ordinary least square	Huber's M-estimate	<i>RSBN</i>
Square root of sum of squares, $\ \mathbf{y} - \hat{s}_{*,2}\ _2$	47.2771	10.6658	10.6053
Number of amplitudes > 0.5	372	53	45
Number of amplitudes > 1.0	196	16	15

One commonality: they both take linear function outside an interval: $(-\delta, \delta)$ for *RSBN* and $(-\Delta, \Delta)$ for Huber's.

6.2 Comparison with Ordinary Least Square Estimate and Huber's Estimate

We compare three different regression methods: ordinary least square estimate, *RSBN*, and Huber's M-estimate. We demonstrate that for distorted Gaussian noises, *RSBN* does the best job.

6.2.1 Design of Simulation

We choose $m = 15, n = 600$. In each experiment, for the model in (2.1), A is generated by sampling each entry $(A_{ij}, 1 \leq i \leq n, 1 \leq j \leq m)$ from a standard Normal distribution ($N(0, 1)$), with the constraint that matrix A must have full column rank. If the generated matrix A does not have full column rank, the process is repeated instead of proceeding to the next step. The vector \mathbf{x} is generated as a standard Normal vector in \mathbf{R}^m , $\mathbf{x} \sim N(0, \mathbf{I}_m)$. The vector ε is generated as a standard Normal vector in \mathbf{R}^n , $\varepsilon \sim N(0, \mathbf{I}_n)$. The observation vector \mathbf{y} is a superposition: $\mathbf{y} = A\mathbf{x} + \varepsilon$.

Let $\text{span}(A)$ denote the linear subspace spanned by the column vectors in matrix A . Obviously, it has m degrees of freedom, $\dim(\text{span}(A)) = m$. Recall the operator $\mathbf{P}_{A,LS}$:

$\mathbf{R}^n \rightarrow \text{span}(A)$ is the projection operator from Euclidean space \mathbf{R}^n to the linear subspace $\text{span}(A)$. In other words,

$$\mathbf{P}_{A,LS}(\mathbf{y}) = \underset{\mathbf{u} \in \text{span}(A)}{\text{argmin}} \quad \|\mathbf{u} - \mathbf{y}\|_{\ell_2}^2.$$

Let $d_{LS,1}$ denote the deviation vector from the least square projection $\mathbf{P}_{A,LS}(\mathbf{y})$ to the true linear component $A\mathbf{x}$. Note here the first subscript “ LS ” indicates the least square method, and the second subscript “ 1 ” indicates the Gaussian noise vector (ε). We have

$$d_{LS,1} = \mathbf{P}_{A,LS}(\mathbf{y}) - A\mathbf{x} = \mathbf{P}_{A,LS}(\varepsilon). \quad (6.29)$$

We then distort the Gaussian vector ε . We randomly select 1% entries in ε , multiply them by 200 (value 200 is arbitrarily chosen). The new vector is denoted by ε' . Effectively, each entry of ε' follows a mixed normal distribution: $\varepsilon'_i \sim 0.99N(0, 1) + 0.01N(0, 200^2)$, $1 \leq i \leq n$. Denote $\mathbf{y}' = A\mathbf{x} + \varepsilon'$.

Recall previously mentioned notations, $\mathbf{P}_{A,RSBN} : \mathbf{R}^n \rightarrow \text{span}(A)$ and $\mathbf{P}_{A,H} : \mathbf{R}^n \rightarrow \text{span}(A)$ are projection operators by adopting RSBN and Huber’s M-estimate respectively. Let $d_{LS,2}$, $d_{RSBN,2}$ and $d_{Huber,2}$ denote the deviation vectors corresponding to the least square estimate, RSBN, and Huber’s M-estimate, respectively. (The first subscripts of the above d ’s indicate methods, and the second subscript “ 2 ” indicates distorted noise vector ε' .) We have

$$\begin{aligned} d_{LS,2} &= \mathbf{P}_{A,LS}(\mathbf{y}') - A\mathbf{x} = \mathbf{P}_{A,LS}(\varepsilon'); \\ d_{RSBN,2} &= \mathbf{P}_{A,RSBN}(\mathbf{y}') - A\mathbf{x} = \mathbf{P}_{A,RSBN}(\varepsilon'); \\ d_{Huber,2} &= \mathbf{P}_{A,Huber}(\mathbf{y}') - A\mathbf{x} = \mathbf{P}_{A,Huber}(\varepsilon'). \end{aligned}$$

We repeat the experiments for 1000 times. Each time, for the distorted noises, three methods lead to three deviation vectors: $d_{LS,2}$, $d_{RSBN,2}$ and $d_{Huber,2}$. Let $d_{LS,2}^{(i)}$, $d_{RSBN,2}^{(i)}$ and $d_{Huber,2}^{(i)}$ denote the deviation vectors we get in the i th experiment, we have totally 3000 n -D vectors: $d_{LS,2}^{(i)}$, $d_{RSBN,2}^{(i)}$, $d_{Huber,2}^{(i)}$, $i = 1, 2, \dots, 1000$.

The smaller the deviations are, the better the regression method is. In the multivariate situation, we need to quantify the smallness. We will report our comparison in Section 6.2.3.

6.2.2 Cut-off Value

To measure the robustness of different methods, it is nature to compare the deviation vectors $d_{LS,2}$, $d_{RSBN,2}$, and $d_{Huber,2}$ with deviation vector $d_{LS,1}$, because $d_{LS,1}$ is the deviation of an ideal method (least square estimation, or MLE) in the ideal situation (with Gaussian noises). We propose to study the number of deviations with amplitudes above a quantity τ : i.e., for $1 \leq i \leq 1000$,

$$\#\{j : |(d_{*,2}^{(i)})_j| > \tau, 1 \leq j \leq n\},$$

where $*$ can be LS, RSBN, or Huber. Here notation $\#$ stands for the cardinality of a finite set. The j th component of vector $d_{*,2}^{(i)}$ is denoted as $(d_{*,2}^{(i)})_j$. Value τ can be viewed as a quantile of random variable $\|d_{LS,1}\|_\infty$. The value τ will be called a *cut-off* value.

The following is to derive a reasonable value of τ . We study the distribution of random variable $\|d_{LS,1}\|_\infty$. Let $\|d_{LS,1}\|_2$ denote the ℓ_2 norm of the vector $d_{LS,1}$. We have

$$\|d_{LS,1}\|_\infty = \|d_{LS,1}\|_2 \cdot \frac{\|d_{LS,1}\|_\infty}{\|d_{LS,1}\|_2}.$$

We list three facts. For details, please refer to Appendix B.

- Random variables $\|d_{LS,1}\|_2$ and $\|d_{LS,1}\|_\infty/\|d_{LS,1}\|_2$ are independent;
- Random variable $\|d_{LS,1}\|_2^2$ satisfies the χ_m^2 distribution with m degrees of freedom. Recall m is the column rank of matrix A .
- Assume the projection $\mathbf{P}_{A,LS}$ related to model matrix A has eigenvalue decomposition

$$\mathbf{P}_{A,LS} = U^T \begin{pmatrix} \mathbf{I}_m \\ 0 \end{pmatrix} U,$$

where matrix U is orthogonal. Let

$$\mathbf{x} = U^T \begin{pmatrix} x_m \\ 0_{(n-m) \times 1} \end{pmatrix},$$

where vector x_m is Uniform on the unit sphere in \mathbf{R}^m , $\|x_m\|_2 = 1$, and vector $0_{(n-m) \times 1}$ is an all zero vector. Let $\rho_{\max, m} = \|\mathbf{x}\|_\infty$. The ratio $\|d_{LS,1}\|_\infty / \|d_{LS,1}\|_2$ has the same distribution as $\rho_{\max, m}$. The analytical solution to the probability density function of $\rho_{\max, m}$ could be too complicated to be useful though.

Based on the above three facts, we can find the distribution of $\|d_{LS,1}\|_\infty$ and the cut-off value through simulations. In this paper, we choose the cut-off value: $\tau = 1$. The related probability $P\{\|d_{LS,1}\|_\infty > \tau\}$ is approximately 3.1×10^{-4} , which is obtained through 100,000 times of simulations.

6.2.3 Simulation Results

Figure 7 illustrates the results from all the steps of one simulation.

- Figure 7 (1) shows the Gaussian noise vector ε . Each element of it satisfies distribution Normal(0, 1).
- Figure 7 (2) shows the deviation vector ($d_{LS,1}$) of the least square regression in the Gaussian noise (ε) case.
- Figure 7 (3) shows the distorted Gaussian noise vector ε' . The vector ε' is gotten by multiplying randomly picked six elements of vector ε by 200.
- Figure 7 (4) shows the deviation vector $d_{LS,2}$ from the least square regression with the distorted Gaussian noise ε' .
- Figure 7 (5) shows the corresponding deviation vector $d_{RSBN,2}$ from the RSBN.
- Figure 7 (6) shows the corresponding deviation vector $d_{Huber,2}$ from the Huber's estimate.

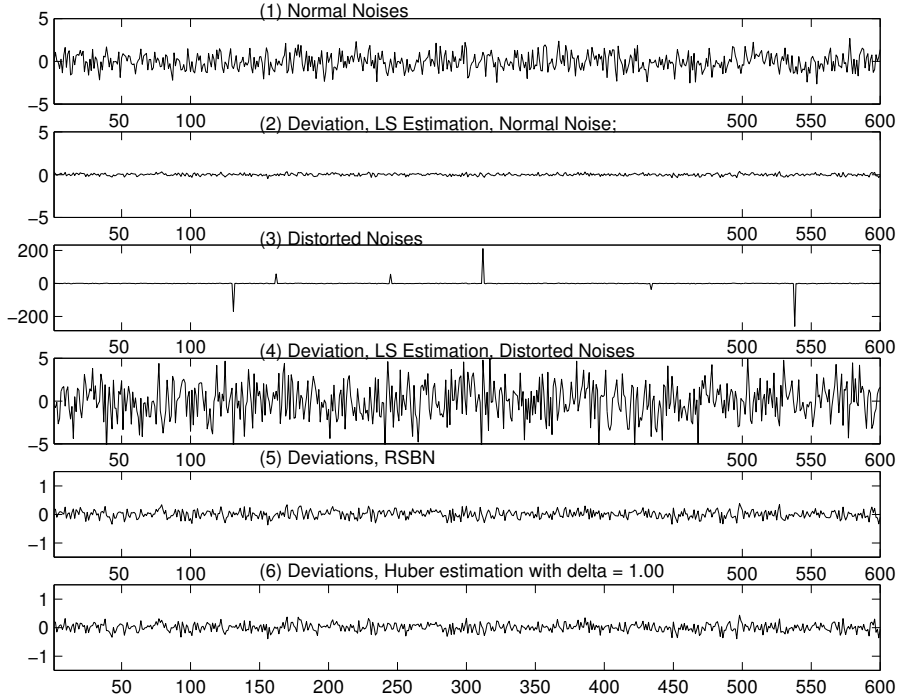


Fig. 7. Quantitative results in one simulation. See text for details.

Continued from Section 6.2.1, we get 3000 deviation vectors out of 1000 simulations: $d_{LS,2}^{(i)}$, $d_{RSBN,2}^{(i)}$, $d_{Huber,2}^{(i)}$, $i = 1, 2, \dots, 1000$.

We choose two ways to compare the three different methods. One is to study the relative ratio of the ℓ_2 norms of a pair of deviation vectors. The other is to count the number of amplitudes above the cut-off line (determined by the τ value developed in Section 6.2.2) in each deviation vector.

Figure 8 (a) gives a histogram of the ratios of the ℓ_2 norms of deviation vectors from the Huber's estimate and RSNB: $\|d_{Huber,2}^{(i)}\|_2^2 / \|d_{RSBN,2}^{(i)}\|_2^2$, $i = 1, 2, \dots, 1000$. We observe that most of them are above 1. This implies the RSNB tends to give smaller sum square of deviations than the Huber's estimate does. Figure 8 (b) shows a histogram of logarithm (base 10) of ratios corresponding to the least square estimate and the RSNB: $\log_{10}(\|d_{LS,2}^{(i)}\|_2^2 / \|d_{RSBN,2}^{(i)}\|_2^2)$, $i = 1, 2, \dots, 1000$. The reason to take logarithm is that some ratios can be extremely large. Obviously, the least square regression for non-Gaussian noise leads to much higher sum of squares of deviations than the RSNB does.

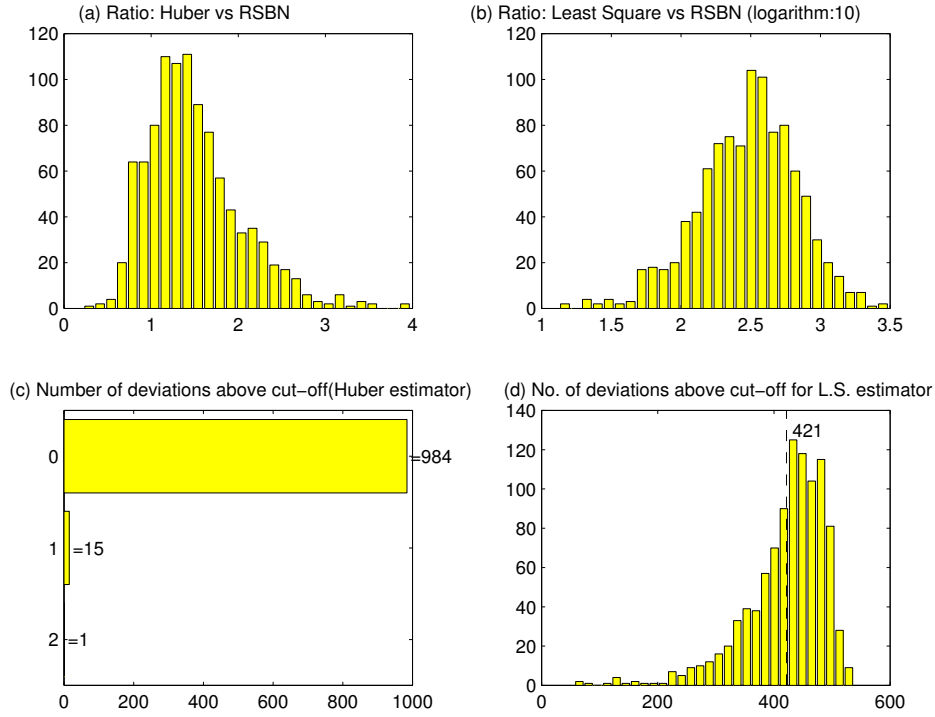


Fig. 8. (a) The histogram of the ratios between the Huber's estimate and RSBN: $\|d_{Huber,2}^{(i)}\|_2^2 / \|d_{RSBN,2}^{(i)}\|_2^2$, $i = 1, 2, \dots, 1000$; (b) The histogram of the logarithm ratios $\log_{10} \left(\|d_{L_2,2}^{(i)}\|_2^2 / \|d_{RSBN,2}^{(i)}\|_2^2 \right)$, $i = 1, 2, \dots, 1000$, for the least square regression and the RSBN; (c) For Huber's estimate, number of deviations whose amplitudes are above the cut-off; (d) For the ordinary least square regression, the histogram of the number of deviations whose amplitudes are above cut-off.

Define the numbers of amplitudes above the cut-off in the following way:

$$\Gamma_{*,2}^{(i)} = \#\{j : |(d_{*,2}^{(i)})_j| > 1, 1 \leq j \leq n\},$$

where the $*$ can be subscripts: LS, RSBN, or Huber. We observe that for all $1 \leq i \leq 1000$, $\Gamma_{RSBN,2}^{(i)} = 0$. This means the RSBN is very robust (in the sense that there is no outstanding deviation from the true signal). The Huber's estimate performs comparably. Figure 8 (c) gives a histogram of $\Gamma_{Huber,2}^{(i)}$, $i = 1, 2, \dots, 1000$. We observe that 15 out of 1000 of them have 1 deviation whose amplitude is larger than 1, and only 1 out of 1000 of them have 2 deviations whose amplitudes are greater than 1. Figure 8 (d) shows a histogram of $\Gamma_{LS,2}^{(i)}$, $i = 1, 2, \dots, 1000$. We can see that in most simulations, the number of

deviations with amplitudes above the cut-off 1 is large. The average number of deviations with amplitudes above the cut-off is 421, which is roughly 70% of the signal.

In this simulation, the RSBN outperforms the Huber's estimate, and the Huber's estimate outperforms the ordinary least square regression.

7 Discussion

7.1 A General Regression Formulation

Equation (2.2) is consistent with many approaches that exist in the literature.

- (1) If $\rho(x) = x^2$, (2.2) is the classical least square regression. The solution can be given by applying hat matrix: $\hat{x} = (A^T A)^{-1} A^T y$. We prefer this formulation if the residuals are normally distributed.
- (2) For $\Delta > 0$, we have

$$\rho(x) = \begin{cases} 0, & |x| < \Delta; \\ |x| - \Delta, & |x| \geq \Delta. \end{cases}$$

Formulation (2.2) is an ℓ_1 regression with a 'dead zone'. By adding some slack variables, (2.2) can be formulated as a linear programming problem. Readers can verify that the following linear programming problem is equivalent to the problem in (2.2).

$$\begin{aligned} & \text{minimize} && \sum_{i=1}^n t_i, \\ & \text{subject to} && -t_i - \Delta \leq y_i - a_i^T x, \quad i = 1, 2, \dots, n; \\ & && y_i - a_i^T x \leq t_i + \Delta, \quad i = 1, 2, \dots, n; \\ & && 0 \leq t_i, \quad i = 1, 2, \dots, n. \end{aligned}$$

The idea of adding a dead zone is to make the large residual relatively more important.

(3) If $\rho(x) = |x|$, (2.2) is the standard least ℓ_1 norm estimation (Dodge, 1987). It can be solved as a linear programming problem (Vanderbei, 1996). This can be viewed as a special case of the last problem: $\Delta = 0$. This formulation is interesting when the noises are Laplacian: i.e., the errors satisfy an exponential distribution. Li and Swetits (1998) established an analytical connection between Huber's estimate (with Δ being a tuning parameter) and the least ℓ_1 norm estimate (which was called linear ℓ_1 estimator in Li and Swetits (1998)). Their result is based on analyzing the solutions to the dual problems, and is inspiring.

7.2 Our Choice of Objective Function vs. Huber's Estimate

Our choice of objective function $\rho(\cdot)$ is rooted in (3.10). We present justification on why to choose such a functional solution as in (3.11). Because function $-\beta_1 \cdot v(x) - \beta_2$ in (3.10) is piecewise constant with discontinuity points $-\delta$ and δ , we consider a generic differential equation:

$$\frac{g''}{g} + C = 0, \quad (7.30)$$

where $C \in \mathbf{R}$ is a constant and $g = \sqrt{f_0}$. The general solution to the above equation, up to a constant, is:

- if $C = 0$, $g = x + c_1$,
- if $C > 0$, $g = \cos(x + c_2)$, and
- if $C < 0$, $g = \exp\{-\sqrt{-C}|x|\}$,

where c_1 and c_2 are constants. Since we want $g(\pm\infty) = 0$, we must assume $-\beta_1 \cdot v(x) - \beta_2 < 0$ outside interval $[-\delta, \delta]$, which leads to the only functional form that vanishes at infinities. Inside interval $[-\delta, \delta]$, we assumed $-\beta_1 \cdot v(x) - \beta_2 > 0$, which leads to the objective function in RSBN. If we choose to assume $-\beta_1 \cdot v(x) - \beta_2 = 0$, then we have $g(x) = x$, which eventually will lead to the Huber's estimate. Our numerical study seems to indicate that our choice leads to relatively more robust performance.

Historically, Huber's estimate is derived differently from our approach, see Lehmann (1991, Section 5.6). They consider an asymptotic minimax estimate among all cumulative distribution functions (c.d.f.) $F(x) = (1 - \varepsilon)G(x) + \varepsilon H(x)$ where constant ε and c.d.f. $G(x)$ are known, and c.d.f. $H(x)$ is unknown but satisfies some general conditions. When $G(x) = \Phi(x)$, which is the c.d.f. of the standard normal, the minimax estimate is the Huber's estimate. Our approach is strongly similar to theirs. However, it differs in the last few steps. We solved the minimax problem in a more general sense.

7.3 Other Theoretical Results

Some results that are related to estimators in regression are worth mentioning.

Researchers have explored the robustness of some regression approaches. For example, Ellis and Morgenthaler (1992) analyzed the 'leverage' and 'breakdown' in minimum ℓ_1 norm regression. The objective in that paper is different from ours: e.g., they do *not* consider asymptotic performance as we formulated and they do *not* consider a minimax estimate. However, their work is inspiring. A citation search of Ellis and Morgenthaler (1992) gives a good sense on what is known about the robustness of some estimators in regression.

In our formulation, we assume the independent noises. Other conditions regarding the regularity of the probability density function of the noises – e.g., the existence of the second derivative of the density, as well as some integrable conditions – are embedded in the derivation of the asymptotic minimaxity. Researchers have studied the condition for an M-estimate to be consistent. The Introduction of a recent article (Berlinet, Liese, and Vajda, 2000) provides a nice overview. Further citation search for the papers cited there gives a full spectrum of the results that are available. In this paper, we did not intend to address those issues. However, it will be an interesting future search to derive minimax M-estimate under weaker regularity conditions.

7.4 Convexity of Fisher Information $I(f)$

In our derivation, we implicitly used the result that Fisher information $I(f)$ is a convex function of f . We give a brief verification of such a convexity. Recall the function $f_t = (1-t)f_0 + tf_1$, $0 \leq t \leq 1$, which was defined in Section 3.3. We have

$$I(f_t) = \int \frac{(f'_t)^2}{f_t} d\mu(x).$$

One can verify that

$$\frac{\partial^2 I(f_t)}{\partial t^2} \Big|_{t=0} = 2 \int \frac{[(f'_1 - f'_0)f_0 - (f_1 - f_0)f'_0]^2}{f_0^3} \geq 0,$$

and the equality is achieved if and only if $f_1 = f_0$, which is *not* true. Hence functional $I(f)$ is strictly convex at every function f_0 .

7.5 Local Minimavity

We can only verify that our RSN estimate is minimax at a neighborhood of function f_0 . Reader can refer to Lemma 3.3. Proving that RSN is a minimax estimate globally (i.e., for all functions satisfying the ‘stochastically bounded noise’ condition) seems to be a difficult task. This problem has not been solved by our paper.

8 Conclusion

We derive an asymptotic minimax estimate in a general regression framework. Extensive numerical simulation demonstrates its advantage over ordinary least square estimate, as well as another robust estimate: Huber’s M-estimate.

An interesting insight of our result is to observe that the derived objective function should be in the form of the ℓ_1 norm outside a neighborhood of the origin. This coincides with many recent applications of ℓ_1 norm in problems such as variable selection. Even

though this paper does not exactly create any link, the connection between the ℓ_1 norm and the asymptotic minimaxity of RSN is certainly something that should be explored in the future.

A Details for Proving (3.21)

First of all, we have

$$I(f_0) = I(f_\theta)|_{\theta=0} = \int \left(\frac{\partial}{\partial \theta} f_\theta \right)^2 \frac{1}{f_\theta} dx |_{\theta=0}. \quad (\text{A.1})$$

From (3.11), we have

$$\begin{aligned} (\text{A.1}) &= \int_{-\delta}^{\delta} \left(2 \cdot c \cdot \cos \lambda_1 \frac{x}{\delta} \cdot \frac{\lambda_1}{\delta} \cdot \sin \frac{\lambda_1 x}{\delta} \right)^2 \frac{1}{c \left[\cos \lambda_1 \frac{x}{\delta} \right]^2} dx \\ &= 4c \frac{\lambda_1^2}{\delta^2} \int_{-\delta}^{\delta} \sin^2 \frac{\lambda_1 x}{\delta} dx + 2c \frac{4\lambda_2^2}{\delta^2} \cos^2 \lambda_1 \cdot \exp(2\lambda_2) \int_{\delta}^{+\infty} \exp\left(-2\lambda_2 \frac{x}{\delta}\right) dx \\ &= 4c \frac{\lambda_1^2}{\delta} \left(1 - \frac{1}{2\lambda_1} \sin 2\lambda_1 \right) + 4c \frac{\lambda_2}{\delta} \cos^2 \lambda_1. \end{aligned} \quad (\text{A.2})$$

From (3.16), we have

$$\begin{aligned} (\text{A.2}) &= 4c \frac{\lambda_1^2}{\delta} \stackrel{(3.18)}{=} 4 \frac{\lambda_1^2}{\delta^2} \frac{\alpha \lambda_2}{\cos^2 \lambda_1} \stackrel{(3.19)}{=} 4 \frac{\lambda_1^2}{\delta^2} \frac{\lambda_2 \cos \lambda_1}{\lambda_1 \cdot \sin \lambda_1 + \cos \lambda_1} \\ &\stackrel{(3.16)}{=} 4 \frac{\lambda_1^2}{\delta^2} \frac{\lambda_1 \cdot \sin \lambda_1}{\lambda_1 \cdot \sin \lambda_1 + \cos \lambda_1}. \end{aligned}$$

B Some Lemmas Regarding Cut-off Values in Section 6.2.2

Lemma 2.1 *Random variable $\|d_{LS,1}\|_2^2$ satisfies the χ_m^2 distribution with m degrees of freedom, where m is the column rank of matrix A .*

Proof. Based on (6.29),

$$d_{LS,1} = \mathbf{P}_{A,LS}(\varepsilon) = A(A^T A)^{-1} A^T \varepsilon.$$

Since $A(A^T A)^{-1} A^T$ is a projection matrix with the rank equals to m , it can be written as

$$A(A^T A)^{-1} A^T = O \cdot \begin{pmatrix} \mathbf{I}_m & 0 \\ 0 & 0 \end{pmatrix}_{n \times n} \cdot O^T,$$

where O is an orthogonal matrix. Hence

$$\|d_{LS,1}\|_2^2 = \left\| \begin{pmatrix} \mathbf{I}_m & 0 \\ 0 & 0 \end{pmatrix} \cdot O^T \cdot \varepsilon \right\|_2^2.$$

Since $O^T \cdot \varepsilon$ also satisfies standard Normal distribution in \mathbf{R}^n , i.e., $O^T \cdot \varepsilon \sim \text{Normal}(0, \mathbf{I}_n)$, we have

$$\|d_{LS,1}\|_2^2 \sim \chi_m^2.$$

□

Lemma 2.2 *The ratio $\|d_{LS,1}\|_\infty / \|d_{LS,1}\|_2$ has the same distribution as random variable $\rho_{\max, m}$, which was defined in Section 6.2.2.*

Proof. Let $\eta = d_{LS,1} / \|d_{LS,1}\|_2$. Recall

$$d_{LS,1} = \mathbf{P}_{A,LS}(\varepsilon) = U^T \begin{pmatrix} \mathbf{I}_m \\ 0 \end{pmatrix} U \cdot \varepsilon.$$

Because $\varepsilon \sim N(0, \mathbf{I}_n)$, we have $U \cdot \varepsilon \sim N(0, \mathbf{I}_n)$ as well. Hence

$$\begin{aligned} \frac{d_{LS,1}}{\|d_{LS,1}\|_2} &= \frac{U^T \begin{pmatrix} \mathbf{I}_m \\ 0 \end{pmatrix}_{n \times n} U \cdot \varepsilon}{\left\| \begin{pmatrix} \mathbf{I}_m \\ 0 \end{pmatrix}_{n \times n} U \cdot \varepsilon \right\|_2} \\ &= U^T \begin{pmatrix} \tilde{x}_m / \|\tilde{x}_m\|_2 \\ 0_{(n-m) \times 1} \end{pmatrix}, \end{aligned}$$

where $\tilde{x}_m = (\mathbf{I}_m \ 0)_{m \times n} U \cdot \varepsilon \sim N(0, \mathbf{I}_m)$. Based on the property of a normally distributed random vector, we have $x_m = \tilde{x}_m / \|\tilde{x}_m\|_2$ is uniformly distributed on the unit sphere $S^{m-1} \subset \mathbf{R}^m$. \square

Lemma 2.3 *Random variables $\|d_{LS,1}\|_2$ and $\|d_{LS,1}\|_\infty / \|d_{LS,1}\|_2$ are independent.*

Proof. Let $\eta = d_{LS,1} / \|d_{LS,1}\|_2$. From the previous lemma, the distribution of the random variable η is independent of the quantity $\|d_{LS,1}\|_2$. Hence $\|d_{LS,1}\|_\infty / \|d_{LS,1}\|_2 = \|\eta\|_\infty$ is independent of $\|d_{LS,1}\|_2$. \square

References

- [1] Berlinet, A., Liese, F., Vajda, I., 2000. Necessary and sufficient conditions for consistency of m-estimates in regression models with general errors. *Journal of Statistical Planning and Inference* 89 ((1-2)), 243–267.
- [2] Bloomfield, P., 1976. *Fourier Analysis of Time Series: An Introduction*. John Wiley & Sons.
- [3] Chen, S. S., Donoho, D. L., Saunders, M. A., 2001. Atomic decomposition by basis pursuit. *SIAM Rev.* 43 (1), 129–159, reprinted from *SIAM J. Sci. Comput.* 20 (1998), no. 1, 33–61.

- [4] Dodge, Y., 1987. *Statistical Data Analysis: Based on the L_1 -norm and Related Methods*. North-Holland.
- [5] Ellis, S. P., Morgenthaler, S., 1992. Leverage and breakdown in l_1 regression. *Journal of the American Statistical Association* 87, 143–148.
- [6] Gill, P. E., Murray, W., Saunders, M. A., 1998. *User's Guide for SNOPT 5.3: A Fortran Package for Large-Scale Nonlinear Programming*. Draft.
- [7] Hampel, F. R., Ronchetti, E. M., Rousseeuw, P. J., Stahel, W. A., 1986. *Robust Statistics: The Approach Based on Influence Functions*. Wiley Series in Pro. and Math. Sci.
- [8] Huber, P. J., 1977. *Robust Statistical Procedures*. Vol. 27. CBMS-NSF.
- [9] Huber, P. J., 1981. *Robust Statistics*. Wiley Series in Pro. and Math. Sci.
- [10] Lehmann, E. L., 1991. *Theory of Point Estimation*. Wadsworth & Brooks/Cole, Pacific Grove, CA.
- [11] Li, W., Swetits, J. J., 1998. The linear l_1 estimator and the huber m-estimator. *Siam J. Optim.* 8 (2), 457–475.
- [12] Michelot, C., Bougeard, M. L., 1994. Duality results and proximal solutions of the huber m-estimator problem. *Applied Mathematics & Optimization* 30, 203–221.
- [13] Spingarn, J. E., 1983. Partial inverse of a monotone operator. *Applied Mathematics and Optimization* 10, 247–265.
- [14] Tibshirani, R. J., 1996. Regression shrinkage and selection via the LASSO. *Journal of the Royal Statistical Society, Ser. B* 58, 267–288.
- [15] Vanderbei, R. J., 1996. *Linear Programming*. Kluwer Academic Publishers.