Towards a Lower Sample Complexity for Robust One-bit Compressed Sensing

Rongda Zhu RZHU4@ILLINOIS.EDU
Department of Computer Science, University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA

Quanquan Gu QG5W@VIRGINIA.EDU
Department of Systems and Information Engineering, University of Virginia, Charlottesville, VA 22904, USA

Abstract

In this paper, we propose a novel algorithm based on nonconvex sparsity-inducing penalty for one-bit compressed sensing. We prove that our algorithm has a sample complexity of $O(s/\epsilon^2)$ for strong signals, and $O(s\log d/\epsilon^2)$ for weak signals, where s is the number of nonzero entries in the signal vector, d is the signal dimension and ϵ is the recovery error. For general signals, the sample complexity of our algorithm lies between $O(s/\epsilon^2)$ and $O(s\log d/\epsilon^2)$. This is a remarkable improvement over the existing best sample complexity $O(s\log d/\epsilon^2)$. Furthermore, we show that our algorithm achieves exact support recovery with high probability for strong signals. Our theory is verified by extensive numerical experiments, which clearly illustrate the superiority of our algorithm for both approximate signal and support recovery in the noisy setting.

1. Introduction

Compressed sensing (Donoho, 2006; Candes & Tao, 2006) is a technique to design measurement matrices and recovery algorithms to estimate a sparse signal vector using a few linear measurements. Recently, one-bit compressed sensing (Boufounos & Baraniuk, 2008), which is a variant of conventional compressed sensing, has received increasing attention for its low computational cost and robustness to noise and non-linearity (Boufounos, 2010). In contrast to conventional compressed sensing, which uses real-valued measurement, one-bit compressed sensing uses one-bit measurement to recover the unknown signals. For example, suppose $\mathbf{x}^*$ is the unknown signal vector, and $\{\mathbf{u}_i\}_{i=1}^n$ is a set of measurement vectors. The sign of real-valued measurement is observed as follows:

$$y_i = \mathrm{sign}(\langle \mathbf{u}_i, \mathbf{x}^* \rangle), i = 1, 2, \ldots, n,$$

where y_i is the binary one-bit measurement. Since signs of real-valued measurements are used, scaling $\mathbf{x}^*$ will not make changes on the measurements. In other words, we cannot recover the norm of $\mathbf{x}^*$ from $\big\{(y_i, \mathbf{u}_i)\big\}_{i=1}^n$. For this reason, when studying approximate vector recovery in one-bit compressed sensing, we always assume that the original signal $\mathbf{x}^*$ is a unit vector, i.e., $\|\mathbf{x}^*\|_2 = 1$.

In general, there are two major tasks in one-bit compressed sensing: (1) support recovery (Gupta et al., 2010; Haupt & Baraniuk, 2011; Gopi et al., 2013), which recovers the support of the unknown signal vector $\mathbf{x}^*$; and (2) approximate signal vector recovery (Gopi et al., 2013; Jacques et al., 2013; Zhang et al., 2014), which aims at finding a unit vector $\widehat{\mathbf{x}}$ such that $\|\widehat{\mathbf{x}} - \mathbf{x}^*\|_2$ is small.

In this paper, we aim at presenting an algorithm, which is able to achieve both approximate signal recovery and support recovery with strong theoretical guarantees. At the core of our method is the nonconvex sparsity-inducing penalty. While nonconvex sparsity-inducing penalties have achieved great success in the statistics community (Fan & Li, 2001; Zhang, 2010; Wang et al., 2014; Gu et al., 2014), it is unclear whether nonconvex sparsity-inducing penalties are advantageous for one-bit compressed sensing. In our study, we show that the answer is in the affirmative. More specifically, the main contributions of this work are summarized as follows:

- We propose to incorporate sparsity-inducing penalty functions into one-bit compressed sensing, and derive an algorithm to efficiently solve the resulting problem. To the best of our knowledge, this is the first work on one-bit compressed sensing that utilizes nonconvex penalty functions.
- We prove that our proposed method improves sample complexity from previous best results $O(s\log d/\epsilon^2)$ to $O(s/\epsilon^2)$ for strong signals. And for general signals, our algorithm attains a sample complexity between

Proceedings of the 32^{nd} International Conference on Machine Learning, Lille, France, 2015. JMLR: W&CP volume 37.

$O(s \log d/\epsilon^2)$ and $O(s/\epsilon^2)$.

- We prove that our proposed method can exactly recover the support of the signal under mild magnitude assumptions on the signal.
- We verify the effectiveness of our method by thorough numerical experiments.

The remainder of this paper is organized as follows. We will review related work in Section 2, and then describe our method in Section 3. We present the main theoretical results in Section 4 and experiment results in Section 5. We conclude the paper in Section 6.

2. Related Work

One-bit compressed sensing was first introduced in (Boufounos & Baraniuk, 2008), with only the noiseless one-bit measurement considered. In particular, Boufounos & Baraniuk (2008) proposed an estimator by minimizing the ℓ_1 norm of a unit vector consistent with the measurements. Since the minimization is on the unit square, the problem is non-convex. To address this problem, Plan & Vershynin (2013b) proposed a convex formulation by putting the constraint on the ℓ_1 norm of real-valued measurement vector instead of the ℓ_2 norm of the signal. The sample complexity of this work is $O(s \log^2 d/\epsilon^5)$. Another convex estimator was proposed in (Plan & Vershynin, 2013a), which maximizes the dot product of the one-bit measurements of the real signal and the real-valued measurements of the recovered signal. This framework can recover both exactly and approximately sparse signals with noise, with sample complexity $O(s \log d/\epsilon^4)$. Currently, the best sample complexity result for vector recovery is achieved by (Zhang et al., 2014), where the authors proposed an efficient algorithm with close-from solution based on adding ℓ_1 regularization. In noisy and noiseless cases, the sample complexity of their work is $O(s \log d/\epsilon^2)$ for exactly sparse signals.

Another branch of one-bit compressed sensing is support recovery. Current best result in terms of sample complexity is $O(s \log d)$ in (Haupt & Baraniuk, 2011). However, their work depends on various specially designed measurement matrices, thus not universal. A universal support recovery method is proposed in (Gopi et al., 2013), where all signals can be recovered using a single measurement matrix. The sample complexity of their work is $O(s^2 \log d)$.

Most of the methods mentioned above use Gaussian measurements, and recently it is also extended to non-Gaussian measurements in (Ai et al., 2014), where the authors use sub-Gaussian measurements to recover both exactly and approximately sparse signals. There are also other extensions. For example, Movahed et al. (2014) considered recovery of signals with unknown and time-variant sparsity levels. Zeng & Figueiredo (2014) studied one-bit compressed sensing on piece-wise smoothing signals.

On the other hand, most of the existing studies only target one of the tasks in approximate vector recovery and support recovery. In this paper, we propose a method that can improve the previous best results for approximate vector recovery and achieve exact support recovery simultaneously.

3. The Proposed Method

In this section we will describe our method. Theoretical analysis of our method can be found in the next section.

3.1. Background

We will first briefly review the general framework of one-bit compressed sensing. As described in (Plan & Vershynin, 2013a), we assume y_i can be viewed as drawn independently with the expectation

$$\mathrm{E}(y_i|\mathbf{u}_i) = \theta\big(\langle \mathbf{u}_i, \mathbf{x}^* \rangle\big), i = 1, 2, \ldots, n,$$

where value domain of the function $\theta(z)$ is $[-1, 1]$. We define

$$\mathrm{E}[\theta(g)g] =: \gamma > 0, \tag{3.1}$$

where g is a standard Gaussian random variable, and γ measures the correlation between y_i and $\langle \mathbf{u}_i, \mathbf{x}^* \rangle$.

When these two are well correlated, γ will get a larger value. When y_i is equal to $\mathrm{sign}(\langle \mathbf{u}_i, \mathbf{x}^* \rangle)$ with no noise, γ will get maximal value $\sqrt{2/\pi}$. Since scaling of $\mathbf{x}^*$ does not influence the one-bit measurements and we cannot restore the scale, we assume $\mathbf{x}^*$ is a unit vector, i.e., $\|\mathbf{x}^*\|_2 = 1$.

3.2. Nonconvex Penalty Functions

We now introduce the decomposable nonconvex penalty functions we consider in our work, i.e., $\mathcal{G}_{\lambda,b}(\mathbf{x}) = \sum_{i=1}^{d} g_{\lambda,b}(x_i)$.

Typical nonconvex penalties include the smoothly clipped absolute deviation (SCAD) penalty (Fan & Li, 2001) and minimax concave penalty (MCP) (Zhang, 2010). For example, MCP is given by

$$g_{\lambda,b}(t) = \begin{cases} \lambda|t| - \dfrac{t^2}{2b}, & \text{if } |t| \le b\lambda, \\ \dfrac{b\lambda^2}{2}, & \text{if } |t| > b\lambda, \end{cases} \tag{3.2}$$

with fixed regularization parameters $b > 0, \lambda > 0$. A well-used property of the nonconvex penalties $g_{\lambda,b}(t)$ is that they can be formulated as the sum of a ℓ_1 penalty part and a concave part $h_{\lambda,b}(t) : g_{\lambda,b}(t) = \lambda|t| + h_{\lambda,b}(t)$.

Note that we do not require specific forms of $g_{\lambda,b}(t)$, such as MCP or SCAD. Generally, our work only depends on the following conditions on $g_{\lambda,b}(t)$ and $h_{\lambda,b}(t)$:

C1. $g'_{\lambda,b}(t) = 0$, for $|t| \geq \nu \geq 0$.
C2. $h'_{\lambda,b}(t)$ is monotone, and for $t' > t$, there is a constant $\zeta_- \geq 0$ such that

$$-\zeta_-(t'-t) \leq h'_{\lambda,b}(t') - h'_{\lambda,b}(t).$$

C3. $h_{\lambda,b}(0) = h'_{\lambda,b}(0) = 0$.
C4. $|h'_{\lambda,b}(t)| \leq \lambda$ for any t.

There are a lot of nonconvex penalty functions that hold the above properties. We can prove that MCP and SCAD are valid choices. For MCP, $\nu = b\lambda$ and $\zeta_- = 1/b$. Since we use MCP as our nonconvex penalty, we will use g, $\mathcal{G}$ and h, $\mathcal{H}$ to specifically denote MCP in (3.2) and its concave part for the rest of this paper.

3.3. One-bit Compressed Sensing with Nonconvex Penalty

Our estimator is any local optimal solution to the following optimization problem

$$\widehat{\mathbf{x}}_\tau = \underset{\|\mathbf{x}\|_2 \leq 1}{\operatorname{argmin}} -\frac{1}{n}\sum_{i=1}^{n} y_i\langle \mathbf{u}_i, \mathbf{x}\rangle + \mathcal{G}_{\lambda,b}(\mathbf{x}) + \frac{\tau}{2}\|\mathbf{x}\|_2^2, \tag{3.3}$$

where $\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_n \in \mathbb{R}^d$ are the rows of known measurement matrix $\mathbf{U} \in \mathbb{R}^{n\times d}$.

Now we describe our algorithm to efficiently compute our estimator by deriving the local minima of (3.3). We denote $\mathbf{v} = \mathbf{U}^\top \mathbf{y}/n \in \mathbb{R}^d$. We begin with the following lemma.

Lemma 3.1. *The solution to the following optimization problem*

$$\widehat{x} = \underset{x}{\operatorname{argmin}} \frac{1}{2}(x-y)^2 + g_{\lambda,b}(|x|)$$

is given by

- ***if*** $b > 1$

$$\widehat{x} = \begin{cases} \dfrac{S(y,\lambda)}{1-1/b}, & \text{if } |y| \leq b\lambda, \\ y, & \text{if } |y| > b\lambda, \end{cases} \tag{3.4}$$

- ***if*** $b \leq 1$

$$\widehat{x} = \begin{cases} 0, & \text{if } |y| \leq \sqrt{b}\lambda, \\ y, & \text{if } |y| > \sqrt{b}\lambda, \end{cases} \tag{3.5}$$

where $S(y,\lambda)$ is the soft-thresholding operator (Donoho et al., 1993) defined for $\lambda \geq 0$ by

$$S(y,\lambda) = \begin{cases} y-\lambda, & \text{if } y > \lambda, \\ 0, & \text{if } |y| \leq \lambda, \\ y+\lambda, & \text{if } y < -\lambda. \end{cases}$$

Proof. For $b > 1$, please see (Breheny & Huang, 2011). For $b \leq 1$, please see the longer version of this paper. □

We can quickly come up with a similar version of Lemma 3.1 with $\tau > 0$.

Lemma 3.2. *The solution to the following optimization problem*

$$\widehat{x} = \underset{x}{\operatorname{argmin}} \frac{1}{2}(x-y)^2 + g_{\lambda,b}(|x|) + \frac{\tau}{2}x^2$$

is given by

- ***if*** $b(1+\tau) > 1$

$$\widehat{x} = \begin{cases} \dfrac{S(y,\lambda)}{1+\tau-1/b}, & \text{if } |y| \leq b\lambda(1+\tau), \\ \dfrac{y}{1+\tau}, & \text{if } |y| > b\lambda(1+\tau). \end{cases} \tag{3.6}$$

- ***if*** $b(1+\tau) \leq 1$

$$\widehat{x} = \begin{cases} 0, & \text{if } |y| \leq \sqrt{b(1+\tau)}\lambda, \\ \dfrac{y}{1+\tau}, & \text{if } |y| > \sqrt{b(1+\tau)}\lambda. \end{cases} \tag{3.7}$$

For the omitted proof of lemmas and theorems in the rest of this work, please see the longer version of this paper.

Now we are ready to solve (3.3). For the sake of simplicity, we first consider the case where $\tau = 0$ to illustrate our method. We will show that the case where $\tau > 0$ can be solved in a similar way.

We consider the Lagrange function $f(\mu)$ of (3.3) given by

$$\begin{aligned} f(\mu) &= \min_{\mathbf{x}} -\mathbf{x}^\top\mathbf{v} + \mathcal{G}_{\lambda,b}(\mathbf{x}) + \mu(\|\mathbf{x}\|_2^2 - 1) \\ &= \min_{\mathbf{x}} 2\mu\left(\frac{1}{2}\|\mathbf{x} - \frac{\mathbf{v}}{2\mu}\|_2^2 + \frac{\mathcal{G}_{\lambda,b}(\mathbf{x})}{2\mu}\right) - \frac{\|\mathbf{v}\|_2^2}{4\mu} - \mu \\ &= 2\mu\left(\sum_i \min_{x_i} \frac{1}{2}\left(x_i - \frac{v_i}{2\mu}\right)^2 + g_{\lambda/(2\mu),2\mu b}(|x_i|)\right) \\ &\quad - \frac{\|\mathbf{v}\|_2^2}{4\mu} - \mu. \end{aligned} \tag{3.8}$$

Note that for the last step, given (3.2), we would easily get that $\frac{1}{2\mu}g_{\lambda,b}(|x_i|) = g_{\lambda/2\mu,2\mu b}(|x_i|)$. We will use μ^* to denote the dual optimal solution that maximizes $f(\mu)$.

According to Lemma 3.1, we need to consider two cases: (1) $2\mu b \leq 1$ and (2) $2\mu b > 1$. For the first case, i.e., $0 < \mu \leq 1/2b$, we have summarized our method in Algorithm 1. For $\mu > 1/2b$, our method is summarized in Algorithm 2. We will just sketch the outline here, and derivation and technical details of Algorithm 1 and 2 can be found in the longer version of this paper.

- $2\mu b \leq 1$: In this case, the solution to (3.8) comes from (3.5). Therefore, we need to compare the value of $|v_i/2\mu|$ and $\lambda\sqrt{b/2\mu}$, which is equivalent to comparing μ and $v_i^2/2b\lambda^2$, to decide the value of each term in the summation in (3.8). After sorting $|v_i|$ and dividing the feasible region into intervals, we will compute $f(\mu)$ and find μ^* within each interval, which has a close form solution as in Line 5 to 9 of Algorithm 1. Finally, among optimal solutions in each interval, we find μ_1^* that maximizes $f(\mu)$.

Algorithm 1 Find maximizer of $f(\mu)$ when $\mu \leq 1/2b$

1: **Input:** $\lambda, b, \mathbf{v}$
2: **Output:** μ_1^*
3: Initialize $f = f(1/2b), \mu_1^* = 1/2b$
4: $v_{(1)}, v_{(2)}, ..., v_{(d)}$ = **Sort**$(|v_1|, |v_2|, ..., |v_d|)$
5: $v_{(0)} = 0, v_{(d+1)} = \infty$
6: $l =$ **Find**$(v_{(l)} \leq 1/2b < v_{(l+1)})$
7: **for** i:=0 ... l **do**
8: **if** $\sqrt{\sum_{j=i}^{n} v_{(j)}^2}/2 \in (v_{(i)}^2/2b\lambda^2, v_{(i+1)}^2/2b\lambda^2]$ **then**
9: $\mu = \sqrt{\sum_{j=i}^{d} v_{(j)}^2}/2$
10: **else**
11: $\mu = v_{(i+1)}^2/2b\lambda^2$
12: **end if**
13: **if** $f(\mu) > f$ **and** $\mu < 1/2b$ **then**
14: $f = f(\mu),\ \mu_1^* = \mu$
15: **end if**
16: **end for**

- $2\mu b > 1$: In this case, the solution to (3.8) comes from (3.4). We do similar sorting and dividing operation, yet within each interval, we need to solve a simple optimization as in Line 8, Algorithm 2. Then we will find the final μ_2^* based on values from each interval.

After finding the optimal values of μ from the above two cases, we compare the objective function values of outputs of Algorithm 1 and 2 to get the final μ^*:

$$\mu^* = \underset{\mu \in \{\mu_1^*, \mu_2^*\}}{\operatorname{argmax}} f(\mu). \tag{3.9}$$

The optimal primal solution is

$$\widehat{\mathbf{x}} = \underset{\mathbf{x}}{\operatorname{argmin}} \frac{1}{2}\|\mathbf{x} - \frac{\mathbf{v}}{2\mu^*}\|_2^2 + \frac{\mathcal{G}_{\lambda,b}(\mathbf{x})}{2\mu^*}.$$

Algorithm 2 Find maximizer of $f(\mu)$ when $\mu > 1/2b$

1: **Input:** $\lambda, b, \mathbf{v}$
2: **Output:** μ_2^*
3: Initialize $f = f(1/2b), \mu_2^* = 1/2b$
4: $v_{(1)}, v_{(2)}, ..., v_{(d)}$ = **Sort**$(|v_1|, |v_2|, ..., |v_d|)$
5: $v_{(0)} = 0, v_{(d+1)} = \infty$
6: $l =$ **Find**$(v_{(l)} \leq 1/2b < v_{(l+1)})$
7: **for** i:=l ... n **do**
8: $S_1 = \sum_{j=i+1}^{n} v_{(j)}^2$
9: $S_2 = \sum_{j=l}^{i} (|v_{(j)}| - \lambda)^2$
10: $J(\mu) = \frac{S_1}{4\mu} + \frac{S_2}{2(2\mu - 1/b)} + \mu$
11: **if** $\mu_i = \operatorname{argmin}_\mu J(\mu) \in (|v_{(i)}|/2b\lambda, |v_{(i+1)}|/2b\lambda]$ **then**
12: $\mu = \mu_i$
13: **else**
14: $\mu = |v_{(i+1)}|/2b\lambda$
15: **end if**
16: **if** $f(\mu) > f$ **and** $\mu > 1/2b$ **then**
17: $f = f(\mu),\ \mu_2^* = \mu$
18: **end if**
19: **end for**

By Lemma 3.1, we would finally get our estimator as follows:

- ***if*** $2\mu^* b > 1$

$$\widehat{x}_i = \begin{cases} \dfrac{S(v_i, \lambda)}{2\mu^* - 1/b}, & \text{if } |v_i| \leq 2\mu^*\lambda b, \\ \dfrac{v_i}{2\mu^*}, & \text{if } |v_i| > 2\mu^*\lambda b. \end{cases}$$

- ***if*** $2\mu^* b \leq 1$

$$\widehat{x}_i = \begin{cases} 0, & \text{if } |v_i| \leq \sqrt{2\mu^* b}\lambda, \\ \dfrac{v_i}{2\mu^*}, & \text{if } |v_i| > \sqrt{2\mu^* b}\lambda. \end{cases}$$

For the case $\tau > 0$, we have a similar Lagrange function $f(\mu')$ with $\mu' = \mu + \tau/2$. The optimization of $f(\mu')$ is in a similar manner, and we omit it here.

Note that although our algorithm is fairly involved, it only involves sorting and analytic form calculation. So it is still very efficient.

Remark 3.3. *When $\tau > \zeta_-$, the estimator in (3.3) is actually strongly convex. The output of our algorithm is the global optimal solution for (3.3). When $\tau = 0$, the estimator in (3.3) is nonconvex and our algorithm will output a local minimum in the primal.*

4. Main Results

We will prove that under a reasonable assumption on the elements of the true signal $\mathbf{x}^*$, our estimator will have ora-

cle property, i.e., identical to the oracle estimator, with high probability. This indicates exact support recovery. We will also show the advantage of our method in terms of sample complexity.

4.1. Oracle Property and Sample Complexity of Our Estimator for Strong Signals

In this section, we will introduce the advantage of our estimator for strong signals, which consists of two parts, the oracle property and improved sample complexity of $O(s/\epsilon^2)$. The definition of the oracle estimator $\widehat{\mathbf{x}}_O$ is given by

$$\widehat{\mathbf{x}}_O = \underset{\mathrm{supp}(\mathbf{x})\subset S, \|\mathbf{x}\|_2\leq 1}{\operatorname{argmin}} \mathcal{L}_O(\mathbf{x}), \tag{4.1}$$

where $\mathcal{L}_O(\mathbf{x}) = -1/n\sum_{i=1}^n y_i\langle \mathbf{u}_i, \mathbf{x}\rangle$. Note that the oracle estimator does not have ℓ_2 regularizer.

In the rest of this paper, we define the following shorthand notations

$$\mathcal{H}_{\lambda,b}(\mathbf{x}) = \sum_{i=1}^d h_{\lambda,b}(x_i) = \mathcal{G}_{\lambda,b}(\mathbf{x}) - \lambda\|\mathbf{x}\|_1,$$
$$\mathcal{L}(\mathbf{x}) = \mathcal{L}_O(\mathbf{x}) + \frac{\tau}{2}\|\mathbf{x}\|_2^2 = -\frac{1}{n}\mathbf{y}^\top\mathbf{U}\mathbf{x} + \frac{\tau}{2}\|\mathbf{x}\|_2^2,$$

and

$$\widetilde{\mathcal{L}}_\lambda(\mathbf{x}) = \mathcal{L}(\mathbf{x}) + \mathcal{H}_{\lambda,b}(\mathbf{x}) = -\frac{1}{n}\mathbf{y}^\top\mathbf{U}\mathbf{x} + \frac{\tau}{2}\|\mathbf{x}\|_2^2 + \mathcal{H}_{\lambda,b}(\mathbf{x}).$$

Before we lay out the main result, we first present two lemmas, which are central to prove the main result. First, we have the following important property for the oracle estimator.

Lemma 4.1. *If $\tau \leq \|\mathbf{v}_S\|_2$ where $\mathbf{v} = -1/n\sum_{i=1}^n y_i\mathbf{u}_i$ and S is the support of $\mathbf{x}^*$. The following optimization problem*

$$\widehat{\mathbf{x}} = \underset{\mathrm{supp}(\mathbf{x})\subset S, \|\mathbf{x}\|_2\leq 1}{\operatorname{argmin}} -\frac{1}{n}\sum_{i=1}^n y_i\langle \mathbf{u}_i, \mathbf{x}\rangle + \frac{\tau}{2}\|\mathbf{x}\|_2^2, \tag{4.2}$$

has the same solution as the oracle estimator in (4.1).

Second, the following lemma reveals the relation between the real signal and the measurements.

Lemma 4.2. *With a probability at least $1 - 1/d$, we have*

$$\left\|\frac{\mathbf{U}_S^\top\mathbf{y}}{n} - \gamma\mathbf{x}_S^*\right\|_2 \leq C\sqrt{\frac{s}{n}}, \tag{4.3}$$

where C is a universal constant and S is the support of $\mathbf{x}^*$.

Equipped with Lemma 4.1 and Lemma 4.2, we have the following theorem establishing the oracle property and sample complexity of our estimator for strong signals.

Theorem 4.3. *Assume that we have the nonconvex penalty $\mathcal{G}_\lambda(\mathbf{x}) = \sum_{i=1}^d g_{\lambda,b}(x_i)$ that satisfies conditions C1 and C2. If the true signal $\mathbf{x}^*$ satisfies the magnitude condition $\min_{j\in S}|x_j^*| \geq \nu + \|\widehat{\mathbf{x}}_O - \mathbf{x}^*\|_2$, for our estimator $\widehat{\mathbf{x}}_\tau$ with regularization parameter $\lambda = C\sqrt{\log d/n} + |\gamma - \tau|$ and $\zeta_- < \tau \leq \|\mathbf{v}_S\|_2$ as in Lemma 4.1, we have*

(1) $\widehat{\mathbf{x}}_\tau = \widehat{\mathbf{x}}_O$;

(2) With a probability of at least $1 - 1/d$,

$$\|\widehat{\mathbf{x}}_\tau - \mathbf{x}^*\|_2 \leq \frac{C}{\gamma}\sqrt{\frac{s}{n}},$$

where C is a universal constant.

In Theorem 4.3, the dependence on s is suboptimal, because we only obtain the ℓ_2 norm based estimator error bound. In order to get rid of s, we need ℓ_∞ norm based estimator error bound, which requires a much stronger condition namely irrepresentable condition (Wainwright, 2009).

Theorem 4.3 indicates that our estimator will be identical to oracle estimator under a magnitude assumption, while requiring no oracle information a priori. This will lead to exact support recovery directly. It is worth noting that the ℓ_2 regularizer in (3.3) is essential to achieve the oracle property, the estimator in (3.3) with $\tau > \zeta_-$ is identical to the oracle estimator in (4.1). In particular, the ℓ_2 regularizer makes the objective function in (3.3) strongly convex, based on which we can prove the estimator with ℓ_2 regularizer in (3.3) is identical to the oracle estimator in (4.1). Note that the oracle estimator in (4.1) does not have ℓ_2 regularizer. Therefore, the l2 regularizer in (3.3) does not introduce any extra approximation error. Note also that for Theorem 4.5, we do not need the ℓ_2 regularizer ($\tau = 0$). Now we analyze the error bound of oracle estimator, which is also the error bound of our estimator. We will also show that the magnitude assumption is actually a weak assumption.

Moreover, we can see that the recovery error of our method for strong signals is just $O(\sqrt{s/n})$, indicating a sample complexity of $O(s/\epsilon^2)$, which is a significant improvement from previous best result $O(s\log d/\epsilon^2)$.

In addition, we have $\|\widehat{\mathbf{x}}_O - \mathbf{x}^*\|_2 \leq C/\gamma\sqrt{s/n}$ with high probability, which does not depend on the magnitude assumption. Therefore, we will only need

$$\min_{j\in S}|x_j^*| \geq \nu + C/\gamma\sqrt{s/n} \tag{4.4}$$

to get $\widehat{\mathbf{x}}_\tau = \widehat{\mathbf{x}}_O$ with probability at least $1 - 1/d$. This is a weak assumption, since one-bit measurements can be acquired at very high rates. When n is very large, the right-hand side of (4.4) will converge to a constant ν. Note again

that for the oracle estimator, the error bound is always of the order of $O(\sqrt{s/n})$, which does not depend on the magnitude assumption.

4.2. Sample Complexity of Our Estimator for General Signals

We now turn to the case of general signals, where the magnitude assumption does not hold necessarily. In this case, we consider our estimator in (3.3) with $\tau = 0$, i.e., $\widehat{\mathbf{x}}_{\tau=0}$. Note that when $\tau = 0$, the problem in (3.3) is not convex. However, as we will see later, our theory applies to any local optimal solution to (3.3). In other words, it is sufficient to get a local optimal solution to (3.3) as our estimator. We start with the following lemma, which characterizes the curvature of the loss function in the ball $\|\mathbf{x}\|_2 \leq 1$.

Lemma 4.4. *For any* $\mathbf{x}$ *where* $\|\mathbf{x}\|_2 \leq 1$*, we have*

$$\frac{\langle \mathbb{E}[\mathbf{U}^\top \mathbf{y}]/n, \mathbf{x}^* - \mathbf{x}\rangle}{\gamma} \geq \frac{1}{2}\|\mathbf{x}^* - \mathbf{x}\|_2^2.$$

Proof. We have

$$\begin{aligned}\langle \mathbb{E}[\mathbf{U}^\top \mathbf{y}]/n, \mathbf{x}^* - \mathbf{x}\rangle &= \langle \gamma \mathbf{x}^*, \mathbf{x}^* - \mathbf{x}\rangle \\ &= \langle \gamma \mathbf{x}^* - \gamma \mathbf{x} + \gamma \mathbf{x}, \mathbf{x}^* - \mathbf{x}\rangle \\ &= \gamma \|\mathbf{x}^* - \mathbf{x}\|_2^2 + \gamma \langle \mathbf{x}, \mathbf{x}^* - \mathbf{x}\rangle. \end{aligned} \tag{4.5}$$

On the other hand, we have

$$\begin{aligned}\langle \mathbf{x}, \mathbf{x}^* - \mathbf{x}\rangle &= \mathbf{x}^\top \mathbf{x}^* - \|\mathbf{x}\|_2^2 \geq \mathbf{x}^\top \mathbf{x}^* - \frac{1}{2} - \frac{1}{2}\|\mathbf{x}\|_2^2 \\ &= \mathbf{x}^\top \mathbf{x}^* - \frac{1}{2}\|\mathbf{x}^*\|_2^2 - \frac{1}{2}\|\mathbf{x}\|_2^2 \\ &= -\frac{1}{2}\|\mathbf{x} - \mathbf{x}^*\|_2^2. \end{aligned} \tag{4.6}$$

Substituting (4.6) into (4.5), we obtain

$$\langle \mathbb{E}[\mathbf{U}^\top \mathbf{y}]/n, \mathbf{x}^* - \mathbf{x}\rangle \geq \frac{\gamma}{2}\|\mathbf{x} - \mathbf{x}^*\|_2^2,$$

which completes the proof. □

We are now ready to present the following theorem, which bounds the error of our estimator for general signals.

Theorem 4.5. *Suppose the nonconvex penalty* $\mathcal{G}_{\lambda,b}(\mathbf{x}) = \sum_{i=1}^d g_{\lambda,b}(x_i)$ *satisfies conditions C2, C3 and C4. For any local optimal solution* $\widehat{\mathbf{x}}_{\tau=0}$ *to* (3.3) *with* $\tau = 0$*,* $\lambda = C\sqrt{\frac{\log d}{n}}$ *and* $\zeta_- < \frac{\gamma}{2}$*, we have with probability at least* $1 - 1/d$ *that*

$$\|\widehat{\mathbf{x}}_{\tau=0} - \mathbf{x}^*\|_2 \leq \underbrace{\frac{2C}{\gamma - 2\zeta_-}\sqrt{\frac{s_1}{n}}}_{S_1:|\mathbf{x}_i^*|\geq\nu} + \underbrace{\frac{6C\sqrt{s_2}}{\gamma - 2\zeta_-}\sqrt{\frac{\log d}{n}}}_{S_2:0<|\mathbf{x}_i^*|<\nu},$$

where C *is a universal constant.*

From Theorem 4.5, we can see that for strong signals, we have $|\mathbf{x}_i^*| \geq \nu$ for all $i \in S$, thus $s_2 = 0$. Then our recovery error is just $O(\sqrt{s/n})$, which is equivalent to a sample complexity of $O(s/\epsilon^2)$. In the worst case, $|\mathbf{x}_i^*| < \nu$ for all $i \in S$, thus $s_2 = s$, and our recovery error is $O(\sqrt{s \log d/n})$. This yields the worst sample complexity of $O(s \log d/\epsilon^2)$.

When $\tau = 0$, our algorithm will output a local minimum in the primal. Note the proof of Theorem 4.5 relies only on the first order necessary condition for local minima. Therefore, any local minima to the optimization problem (3.3) with $\tau = 0$ enjoys the rate in Theorem 4.5. Since the output of our algorithm in this case ($\tau = 0$) is a local minimum in the primal, it enjoys the theoretical guarantee in Theorem 4.5.

5. Experiments

In this section, we will verify our theoretical results on synthetic datasets. For each recovery task, we will tune C and b by cross validation and select λ and τ according to Theorem 4.3 for strong signals and Theorem 4.5 for general signals. For each parameter setting, we present the average results of 100 trials of our method and four other methods:

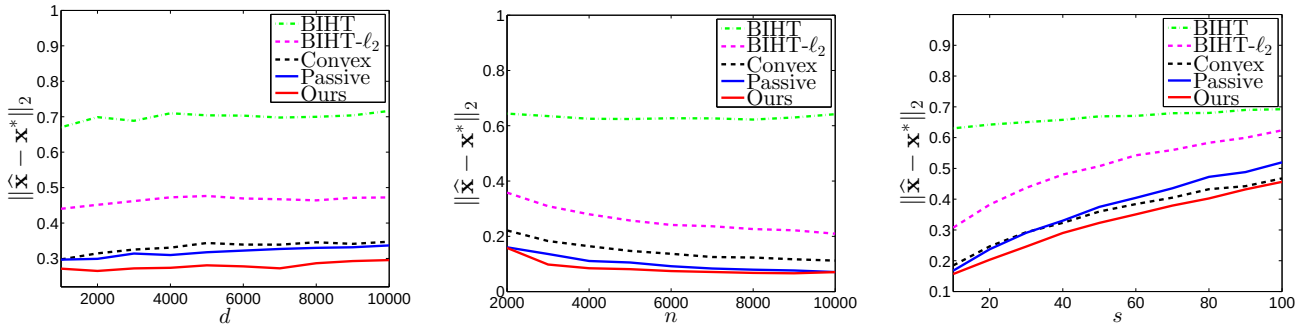
- Passive: the passive algorithm proposed in (Zhang et al., 2014), the best previous result on sample complexity.
- Convex: the convex programming approach proposed in (Plan & Vershynin, 2013a).
- BIHT and BIHT-ℓ_2 proposed in (Jacques et al., 2013)

5.1. Approximate Vector Recovery for General Signals

In this subsection, we will show our experimental results on general signals, i.e., no magnitude assumption guaranteed. We will randomly select their support and draw the values of nonzero elements from a standard normal distribution. The elements in the matrix $\mathbf{U}$ are also drawn from standard normal distribution. We choose the noisy setting in (Plan & Vershynin, 2013a) by flipping the signs of measurements with a probability of 0.1.

Figure 1(a) shows the recovery error against the dimensionality of signals d. We can see that our proposed method outperforms all the other algorithms with a remarkable margin. As the dimensionality of signal d goes up, the recovery error grows slowly, because the dependency on d is logarithmic by Theorem 4.5. We can also see that in this noisy setting, the more vulnerable BIHT and BIHT-ℓ_2 consistently perform worse than the other methods.

Figure 1(b) shows the recovery error against the number of measurements n. Our method consistently achieves the best performance. The passive algorithm also performs reasonably well, but our method outperforms it in a wide range of n.



(a) $s = 10, n = 1000$. (b) $s = 10, d = 1000$. (c) $d = 1000, n = 3000$.

Figure 1. Recovery error for general signals

Figure 1(c) shows the recovery error against the sparsity of signals s. We can see that for all the algorithms except BIHT, the error goes up quickly when s becomes larger. Our algorithm is still consistently the best among all. Note that the dependency on s is not logarithmic, therefore, the error grows much faster than the case of varying d. We choose number of measurements $n = 3000$ here, which is larger than the signal dimension d. This is practical in one-bit compressed sensing, because the one-bit measurements can be generated at very high rates. To sum up, our method can improve recovery accuracy in different parameter settings even with noise.

5.2. Approximate Vector Recovery for Strong Signals

In this subsection, we present results of our recovery algorithm for strong signals. We will first generate unit sparse signals with random support, and set all nonzero entries to $1/\sqrt{s}$. Noise is added in the same way with section 5.1.

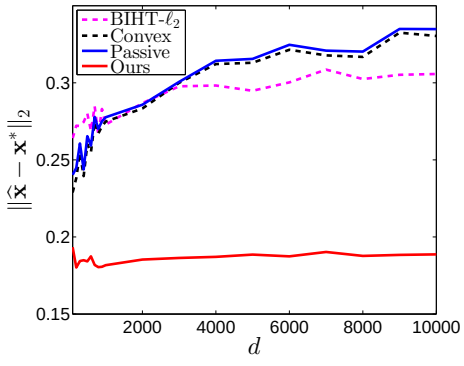


Figure 2. Recovery error of strong signals against d when $s = 10$, $n = 1000$.

Figure 2 shows the recovery error of strong signals. According to Theorem 4.3, our error rate does not depend on dimensionality d, which is verified by the results. Our recovery error stays on the same level, while the errors of all the other algorithms go up with increasing d. Note that the error of BIHT is much higher than the other algorithms. For better illustration and scaling the behavior of the other methods, we omit it in the figure here.

5.3. Support Recovery

We are now going to investigate the problem of support recovery. According to Theorem 4.3, our estimator enjoys oracle property for strong signals. We generate the signals in the same way as section 5.2 and present the F_1 score of support recovery in different d and n settings. F_1 score is defined as the harmonic mean of precision and recall.

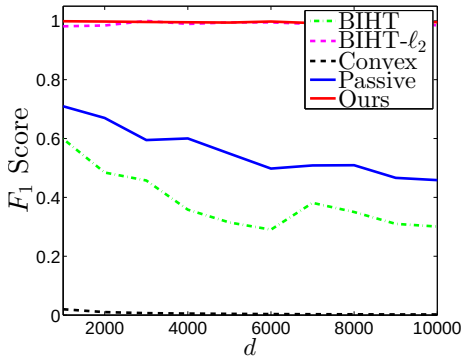
$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$
$$F_1 = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}},$$

where

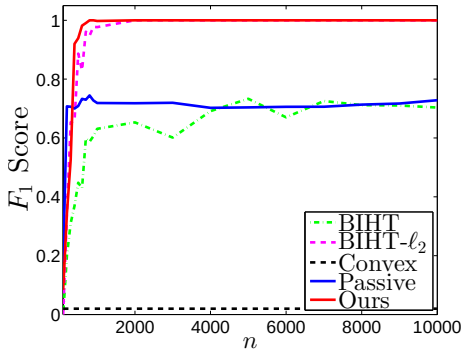
$$\text{TP} = \sum_{i=1}^{d} \mathbf{1}(\widehat{\mathbf{x}}_i \neq 0, \mathbf{x}_i^* \neq 0), \text{FP} = \sum_{i=1}^{d} \mathbf{1}(\widehat{\mathbf{x}}_i \neq 0, \mathbf{x}_i^* = 0)$$
$$\text{FN} = \sum_{i=1}^{d} \mathbf{1}(\widehat{\mathbf{x}}_i = 0, \mathbf{x}_i^* \neq 0).$$

Note that our method is different from best previous work on support recovery. We do not need to construct specific measurement matrix as (Gopi et al., 2013; Haupt & Baraniuk, 2011), nor do we depend on dynamic range or adaption of the measurement process as (Gupta et al., 2010). Therefore, their methods are not directly comparable with ours.

Figure 3(a) shows the F_1 score against signal dimension d. We can see that as the assumption in Theorem 4.3 is satisfied, our algorithm can achieve exact support recovery with very high probability. Our method and BIHT-ℓ_2 out-



(a) $s = 10, n = 1000$.



(b) $s = 10, d = 1000$.

Figure 3. F_1 score for support reovery

perform the other algorithms with notable margins. In addition, Theorem 4.3 indicates that the support recovery of our method does not depend on d, which is also validated by the experiments. While for the other algorithms, the performance of the passive algorithm drops significantly as d goes up; BIHT is not effective either, nor can it achieve a stable performance. Note that for the convex optimization method, there is no ℓ_0 constraints on the signal. Therefore, most of the entries in the estimator are nonzero, resulting in very low precision. This explains the observation that convex optimization method always have a F1-score close to zero.

In Figure 3(b), we plot the F_1 score against number of measurements n. For the same reason, the convex optimization method still suffers very low F_1 score close to 0. For the other four methods, when there are not enough measurements, they perform poorly on support recovery. As the number of measurements goes up, the passive algorithm is the fastest to boost the performance. However, the F_1 score will stop increasing around 0.7 in spite of the increase of measurements. For BIHT, the performance is less stable, but F_1 score will still converge around 0.7 with increasing measurements. Compared with the passive algorithm, our algorithm needs a bit more measurements to converge in terms of F_1 score. Moreover, when n is larger than 500, our algorithm can achieve very good performance, almost recover the support with probability 1. BIHT-ℓ_2 has a similar behavior as our algorithm with enough measurements, but our method requires fewer measurements.

5.4. Oracle Property

We will further study the oracle property of our estimator. We plot the difference between proposed estimator and the oracle estimator in (4.1). By Theorem 4.3, the two should be the same with high probability. In Figure 4, we can see

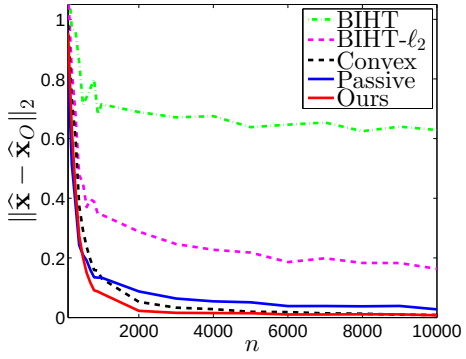


Figure 4. Difference between estimators and oracle estimators against n when $s = 10$, $d = 1000$.

that when the number of measurements goes up, the difference between our estimator and oracle estimator converges to zero very quickly. For BIHT and BIHT-ℓ_2, the differences are large; for the passive algorithm, the difference is still discernible, and the support recovery is not satisfying; for the convex optimization algorithm, although the norm of the difference is converging, it cannot recover the support. Therefore, our estimator is the only one that enjoys oracle property.

6. Conclusions

In this paper, we proposed a novel and effective method based on nonconvex penalty functions for one-bit compressed sensing. Compared with existing methods, our method improves the sample complexity significantly, and achieves excellent performance on support recovery. We also showed that our method is robust to noise by numerical experiments.

Acknowledgments

We would like to thank the anonymous reviewers for their helpful comments. Research was sponsored by Quanquan Gu's startup funding at Department of Systems and Information Engineering, University of Virginia.

References

Ai, Albert, Lapanowski, Alex, Plan, Yaniv, and Vershynin, Roman. One-bit compressed sensing with non-gaussian measurements. *Linear Algebra and its Applications*, 441 (0):222 – 239, 2014. ISSN 0024-3795. Special Issue on Sparse Approximate Solution of Linear Systems.

Boufounos, P. T. and Baraniuk, R. G. 1-bit compressive sensing. In *Proc. Conf. Inform. Science and Systems (CISS)*, Princeton, NJ, March 19-21 2008.

Boufounos, P.T. Reconstruction of sparse signals from distorted randomized measurements. In *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*, pp. 3998–4001, March 2010.

Breheny, Patrick and Huang, Jian. Coordinate descent algorithms for nonconvex penalized regression, with applications to biological feature selection. *The Annals of Applied Statistics*, 5(1):232–253, 03 2011.

Candes, E.J. and Tao, T. Near-optimal signal recovery from random projections: Universal encoding strategies? *Information Theory, IEEE Transactions on*, 52(12):5406–5425, Dec 2006. ISSN 0018-9448.

Donoho, D., Johnstone, I., and Johnstone, Iain M. Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, 81: 425–455, 1993.

Donoho, D.L. Compressed sensing. *Information Theory, IEEE Transactions on*, 52(4):1289–1306, April 2006. ISSN 0018-9448.

Fan, J. and Li, R. Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association*, 96(456):1348–1360, 2001.

Gopi, Sivakant, Netrapalli, Praneeth, Jain, Prateek, and Nori, Aditya V. One-bit compressed sensing: Provable support and vector recovery. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*, pp. 154–162, 2013.

Gu, Quanquan, Wang, Zhaoran, and Liu, Han. Sparse pca with oracle property. In *Advances in Neural Information Processing Systems*, pp. 1529–1537, 2014.

Gupta, A., Nowak, R., and Recht, B. Sample complexity for 1-bit compressed sensing and sparse classification. In *Information Theory Proceedings (ISIT), 2010 IEEE International Symposium on*, pp. 1553–1557, June 2010.

Haupt, J. and Baraniuk, R. Robust support recovery using sparse compressive sensing matrices. In *Information Sciences and Systems (CISS), 2011 45th Annual Conference on*, pp. 1–6, March 2011.

Jacques, L., Laska, J. N., Boufounos, P. T., and Baraniuk, R. G. Robust 1-bit compressive sensing via binary stable embeddings of sparse vectors. *IEEE Trans. Info. Theory*, 59(4), April 2013.

Movahed, A., Panahi, A., and Reed, M.C. Recovering signals with variable sparsity levels from the noisy 1-bit compressive measurements. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pp. 6454–6458, May 2014.

Plan, Y. and Vershynin, R. Robust 1-bit compressed sensing and sparse logistic regression: A convex programming approach. *Information Theory, IEEE Transactions on*, 59(1):482–494, Jan 2013a. ISSN 0018-9448.

Plan, Yaniv and Vershynin, Roman. One-bit compressed sensing by linear programming. *Communications on Pure and Applied Mathematics*, 66(8):1275–1297, 2013b. ISSN 1097-0312.

Wainwright, Martin J. Sharp thresholds for high-dimensional and noisy sparsity recovery using-constrained quadratic programming (lasso). *Information Theory, IEEE Transactions on*, 55(5):2183–2202, 2009.

Wang, Zhaoran, Liu, Han, and Zhang, Tong. Optimal computational and statistical rates of convergence for sparse nonconvex learning problems. *The Annals of Statistics*, 42(6):2164–2201, 12 2014.

Zeng, Xiangrong and Figueiredo, M.A.T. Robust binary fused compressive sensing using adaptive outlier pursuit. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pp. 7674–7678, May 2014.

Zhang, Cun-Hui. Nearly unbiased variable selection under minimax concave penalty. *Ann. Statist.*, 38(2):894–942, 2010.

Zhang, Lijun, Yi, Jinfeng, and Jin, Rong. Efficient algorithms for robust one-bit compressive sensing. In Jebara, Tony and Xing, Eric P. (eds.), *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pp. 820–828. JMLR Workshop and Conference Proceedings, 2014.