Pinball Loss Minimization for One-bit Compressive Sensing

Xiaolin Huang, Lei Shi, Ming Yan, and Johan A.K. Suykens,

***Abstract*—The one-bit quantization can be implemented by one single comparator, which operates at low power and a high rate. Hence one-bit compressive sensing (*1bit-CS*) becomes very attractive in signal processing. When the measurements are corrupted by noise during signal acquisition and transmission, 1bit-CS is usually modeled as minimizing a loss function with a sparsity constraint. The existing loss functions include the hinge loss and the linear loss. Though 1bit-CS can be regarded as a binary classification problem because a one-bit measurement only provides the sign information, the choice of the hinge loss over the linear loss in binary classification is not true for 1bit-CS. Many experiments show that the linear loss performs better than the hinge loss for 1bit-CS. Motivated by this observation, we consider the pinball loss, which provides a bridge between the hinge loss and the linear loss. Using this bridge, two 1bit-CS models and two corresponding algorithms are proposed. Pinball loss iterative hard thresholding improves the performance of the binary iterative hard theresholding proposed in [6] and is suitable for the case when the sparsity of the true signal is given. Elastic-net pinball support vector machine generalizes the passive model proposed in [11] and is suitable for the case when the sparsity of the true signal is not given. A fast dual coordinate ascent algorithm is proposed to solve the elastic-net pinball support vector machine problem, and its convergence is proved. The numerical experiments demonstrate that the pinball loss, as a trade-off between the hinge loss and the linear loss, improves the existing 1bit-CS models with better performances.**

***Index Terms*—compressive sensing, one-bit, classification, pinball loss, dual coordinate ascent**

I. Introduction

Manuscript received 2015;

This work was partially supported by • EU: The research leading to these results has received funding from the European Research Council under the European Union's Seventh Framework Programme (FP7/2007-2013) / ERC AdG A-DATADRIVE-B (290923). This paper reflects only the authors' views, the Union is not liable for any use that may be made of the contained information. • Research Council KUL: GOA/10/09 MaNet, CoE PFV/10/002 (OPTEC), BIL12/11T; PhD/Postdoc grants. • Flemish Government: FWO: projects: G.0377.12 (Structured systems), G.088114N (Tensor based data similarity); PhD/Postdoc grants IWT: projects: SBO POM (100031); PhD/Postdoc grants iMinds Medical Information Technologies SBO 2014. • Belgian Federal Science Policy Office: IUAP P7/19 (DYSCO, Dynamical systems, control and optimization, 2012-2017). L. Shi is supported by the National Natural Science Foundation of China Project No. 11201079) and the Joint Research Fund by National Natural Science Foundation of China and Research Grants Council of Hong Kong (Project No. 11461161006 and Project No. CityU 104012). M. Yan is supported by the National Natural Science Foundation of United States (Nos. DMS-0748839, DMS-1317602).

X. Huang and J.A.K. Suykens are with Department of Electrical Engineering (ESAT-STADIUS), KU Leuven, Kasteelpark Arenberg 10, B-3001, Leuven, Belgium. L. Shi is with Shanghai Key Laboratory for Contemporary Applied Mathematics and School of Mathematical Sciences, Fudan University, Shanghai, P.R. China. M. Yan is with Department of Mathematics, University of California, Los Angeles, CA, USA. (e-mails: huangxl06@mails.tsinghua.edu.cn, leishi@fudan.edu.cn, yanm@math.ucla.edu, johan.suykens@esat.kuleuven.be.)

IN analog-to-digital conversions and succeeding signal processing stages, quantization is an important issue. The extreme quantization scheme is that we only acquire one-bit for each measurement. This scheme only needs a single comparator and has many benefits in hardware implementation such as low power and a high rate. Suppose that we have a linear measurement system $\mathbf{u} \in \mathbb{R}^n$ for a signal $\mathbf{x} \in \mathbb{R}^n$. Then the analog measurement is $\mathbf{u}^T\mathbf{x}$, and the one-bit quantized observation is only its sign, i.e., $y = \mathrm{sgn}(\mathbf{u}^T\mathbf{x})$. We set the sign of a non-negative number as 1 and that of a negative number as -1. Then the signal recovery problem related to one-bit measurements can be formulated as finding a signal $\mathbf{x}$ from the signs of a set of measurements, i.e., $\{\mathbf{u}_i, y_i\}_{i=1}^m$ with

$$y_i = \mathrm{sgn}\left(\mathbf{u}_i^T\mathbf{x}\right).$$

Let $\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_m]$ and $\mathbf{y} = [y_1, y_2, \ldots, y_m]^T$ denote the measurement systems and the measurements respectively. It is easy to notice that signals with the same direction but different magnitudes have the same one-bit measurements with the same measurement systems, i.e., the magnitude of the signal is lost in this quantization. Therefore, we have to make an additional assumption on the magnitude of $\mathbf{x}$. Without loss of generality, we assume $\|\mathbf{x}\|_2 = 1$. Then the meaning of one-bit signal recovery can be explained as finding the subset of the unit sphere $\|\mathbf{x}\|_2 = 1$ partitioned by random hyperplanes. In general, when the number of hyperplanes becomes larger, the feasible set becomes smaller, and the recovery result becomes more accurate.

However, there may still be infinite many points in the subset, and we need additional assumptions on the signal to make it unique. The compressive sensing (CS, [1], [2], [3]) tells us that if the signal is sparse, we may exactly recover the signal with much fewer measurements than the dimension of the signal [4], [5], [6]. This technique has been successfully applied in many fields. However, quantization is rarely considered in these applications.

Motivated by the advantages of one-bit quantization and CS, *one-bit compressive sensing (1bit-CS)* is proposed in [4] and has attracted many attentions in recent years. 1bit-CS tries to recover a sparse signal from the signs of a small number of measurements. Here the number of measurements can be larger than the dimension of the signal, which is different from regular CS. Same as in regular CS problems, the fundamental assumption for 1bit-CS is that the true signal is sparse, i.e., only a few components of the signal are non-zero. Then, 1bit-

CS is to find the sparsest solution in the feasible set, i.e.,

$$\begin{aligned} \min_{\mathbf{x}\in\mathbb{R}^n} \quad & \|\mathbf{x}\|_0 \\ \text{s.t.} \quad & y_i = \operatorname{sgn}(\mathbf{u}_i^T\mathbf{x}), \quad \forall i = 1, 2, \ldots, m, \\ & \|\mathbf{x}\|_2 = 1, \end{aligned} \tag{1}$$

where $\|\cdot\|_0$ counts the number of non-zero components. This problem is non-convex because of the ℓ_0-norm in the objective and the constraint $\|\mathbf{x}\|_2 = 1$. There are several algorithms that approximately solve (1) or its variants. See [4], [5], [7], [8].

In (1), we require that $y_i = \operatorname{sgn}(\mathbf{u}_i^T\mathbf{x})$ holds for all measurements with the assumption that there is no noise in the measurements. However, in real applications, there is always noise in the measuring process, i.e., $y_i = \operatorname{sgn}(\mathbf{u}_i^T\mathbf{x} + \varepsilon_i)$ with $\varepsilon_i \neq 0$. When the noise is small and $\operatorname{sgn}(\mathbf{u}_i^T\mathbf{x} + \varepsilon_i) = \operatorname{sgn}(\mathbf{u}_i^T\mathbf{x})$, we can still recover the true signal accurately. The robustness to small noise is one of the advantages of 1bit-CS. However, when the noise ε_i is large, we may have $\operatorname{sgn}(\mathbf{u}_i^T\mathbf{x} + \varepsilon_i) \neq \operatorname{sgn}(\mathbf{u}_i^T\mathbf{x})$. In addition, there could be sign flips on components of $\mathbf{y}$ during the transmission. Note that sign changes because of noise happen with a higher probability when the magnitudes of true analog measurements are small, while sign flips during the transmission happen randomly among the measurements. With this difference in mind, the methods to deal with these two types of sign changes will also be different.

With noise or/and sign flips, the feasible set of (1) excludes the true signal and can become empty. To deal with noise and sign flips, soft loss functions are used to replace the hard constraint, and it leads to robust 1bit-CS models. The first robust model is given by [6]. It utilizes the following hinge loss to measure the sign changes,

$$L_{\text{hinge}}(t) = \max\{0, t\}.$$

In the same paper, the squared hinge loss is also considered. The attempt in [9] considers the following *linear loss*,

$$L_{\text{linear}}(t) = t.$$

Via minimizing the hinge or the linear loss, some robust 1bit-CS models and corresponding algorithms are proposed in [6], [9], [10], [11] and so on. These models will be reviewed in Section II. With these robust models, 1bit-CS becomes more attractive. For example, it is shown in [12] and [13] that under some conditions, signal recovery based on one-bit measurements is even better than conventional methods for nonlinear distortions and heavy noise.

In 1bit-CS, we only have sign information $y_i \in \{-1, +1\}$, and hence recovering $\mathbf{x}$ can be regarded as a binary classification problem. In the binary classification field, the hinge loss is widely used, e.g., it is the loss function used in the classical support vector machine (SVM, [14]). In [15], [16] and other literature, it is shown that the hinge loss enjoys many good properties for classification such as classification-calibration, Bayes consistency, and so on. In traditional classification tasks, the linear loss is rarely considered. Recently, it is found in [17] that applying the linear loss in SVM is equal to the classical kernel rule [18], which enjoys computational effectiveness yet lacks of accuracy in many tasks. However, according to the experiments in [9] and [11], the linear loss is quite suitable for 1bit-CS, compared with the hinge loss.

This unusual phenomena that the linear loss performs better than the hinge loss motives us to investigate the properties of 1bit-CS. We will apply a *pinball loss* to establish recovery models for 1bit-CS. Statistically, the pinball loss is closely related to the concept of quantile; see [19], [20], [21] for regression, and [22] for classification. In this paper, we use the following definition for the pinball loss:

$$L_\tau(t) = \begin{cases} t, & t \geq 0, \\ -\tau t, & t < 0. \end{cases} \tag{2}$$

The pinball loss is characterized by the parameter τ, and it is convex when $\tau \geq -1$. The hinge loss and the linear loss can be viewed as particular pinball loss functions with $\tau = 0$ and $\tau = -1$, respectively. In other words, $L_\tau(t)$ provides a bridge from the hinge loss to the linear loss. The hinge loss is a good choice for regular classification tasks; and the linear loss shows good performance in 1bit-CS. Hence, it is expected that a suitable trade-off between them can achieve better performance in 1bit-CS.

In this paper, we will discuss two models with the pinball loss minimization. First, based on the model given by [6], we propose a new model consisting of the pinball loss minimization, a ℓ_0-norm constraint, and the ℓ_2-norm unit sphere constraint. This problem is non-convex because of the ℓ_0 norm and ℓ_2 norm constraints. In order to solve this problem, *pinball iterative hard thresholding (PIHT)* is established and evaluated by numerical experiments. Second, we propose a convex model which contains the pinball loss, a ℓ_1-norm regularization term, and the ℓ_2-norm ball constraint. This model considers both the ℓ_1-norm and the ℓ_2-norm. So we name it as *elastic-net pin-SVM (ep-SVM)*. When $\tau = -1$, it reduces to the model given by [11]. To effectively solve ep-SVM, its dual problem is derived, and a dual coordinate ascent algorithm is given. This algorithm is shown to converge to a global optimum, and its effectiveness is illustrated by numerical experiments.

The rest of this paper is organized as follows. A brief review on the existing 1bit-CS methods is given in Section II. Section III introduces the pinball loss and proposes a pinball loss model with a ℓ_0-norm constraint for 1bit-CS. In Section IV, the elastic-net pin-SVM is discussed, and an algorithm is provided to solve it. Both proposed methods are evaluated on numerical experiments in Section V, showing the performance of the pinball loss in 1bit-CS. A conclusion is given to end this paper in Section VI.

II. Review on 1bit-CS Models

1bit-CS was introduced in 2008 by [4], and since then, it has attracted lots of attentions. Since the original model (1) is hard to minimize because of the ℓ_0-norm, which is nonsmooth and non-convex. One alternative way is to minimize the convex hull of ℓ_0, i.e, the ℓ_1-norm, and obtain the following 1bit-CS

model:

$$\begin{aligned} \min_{\mathbf{x}\in\mathbb{R}^n} \quad & \|\mathbf{x}\|_1 \\ \text{s.t.} \quad & y_i(\mathbf{u}_i^T\mathbf{x}) \geq 0,\ \forall i = 1,2,\ldots,m, \\ & \|\mathbf{x}\|_2 = 1. \end{aligned} \tag{3}$$

This model is given by [4], and an efficient heuristic is established in [5].

Due to the fact that the unit sphere is non-convex, (3) is still a non-convex problem. In order to pursue the convexity, the non-convex sphere constraint $\|\mathbf{x}\|_2 = 1$ is replaced by a convex constraint on the measurements in [23], and a convex model is established as follows:

$$\begin{aligned} \min_{\mathbf{x}\in\mathbb{R}^n} \quad & \|\mathbf{x}\|_1 \\ \text{s.t.} \quad & y_i(\mathbf{u}_i^T\mathbf{x}) \geq 0,\ \forall i = 1,2,\ldots,m, \\ & \|\mathbf{U}^T\mathbf{x}\|_1 = s, \end{aligned} \tag{4}$$

where s is a given positive constant. Note that (4) can be reformulated as a linear programming problem because the second constraint $\|\mathbf{U}^T\mathbf{x}\|_1 = s$ becomes $\sum_{i=1}^m y_i(\mathbf{u}_i^T\mathbf{x}) = s$ if the first constraint is satisfied. However, the solution of (4) is not necessarily located on the unit sphere, hence one needs to project the solution onto the unit sphere. In fact, the solution is independent of s after projected onto the unit sphere.

As we mentioned before, these models work only when there is no noise or the noise is too small to change the binary measurements, i.e., there is no sign changes in $\mathbf{y}$. In real applications, noise in the measurements is unavoidable, and there could be sign flips on $\mathbf{y}$ during the transmission. Noise or/and sign flips can make 1bit-CS problems (3) and (4) infeasible, and even feasible, the true signal is not in the feasible set. In other words, the related classification problem is non-separable, and even separable, the classifier is not accurate. To deal with noise and sign flips, one can use a soft loss function instead of the hard constraint. Since models with soft loss functions can tolerate the existence of noise and sign flips, they are called robust 1bit-CS models. In [6], the following robust model is introduced:

$$\begin{aligned} \min_{\mathbf{x}\in\mathbb{R}^n} \quad & \frac{1}{m}\sum_{i=1}^m \max\left\{0, -y_i(\mathbf{u}_i^T\mathbf{x})\right\} \\ \text{s.t.} \quad & \|\mathbf{x}\|_2 = 1, \\ & \|\mathbf{x}\|_0 = K, \end{aligned} \tag{5}$$

where K is the number of non-zero components in the true signal. *Binary iterative hard thresholding with a one-sided ℓ_1-norm* (BIHT) is proposed to solve it approximately. The one-sided ℓ_1-norm is related to the hinge loss function in classical L1-SVM [14], whose statistical property in classification has been well studied and understood in [15], [16], [24] and [25]. Similar to the link between L1-SVM and L2-SVM, (5) can be modified into the following problem via replacing the one-sided ℓ_1-norm with a one-sided ℓ_2-norm:

$$\begin{aligned} \min_{\mathbf{x}\in\mathbb{R}^n} \quad & \frac{1}{m}\sum_{i=1}^m \max\left\{0, -y_i(\mathbf{u}_i^T\mathbf{x})\right\}^2 \\ \text{s.t.} \quad & \|\mathbf{x}\|_2 = 1, \\ & \|\mathbf{x}\|_0 = K, \end{aligned} \tag{6}$$

for which binary iterative hard thresholding with a one-sided ℓ_2-norm (BIHT-ℓ_2, [6]) is proposed. Modifications of BIHT/BIHT-ℓ_2 for sign flips are proposed by [10] to improve their robustness to sign flips. However, this modification can not improve their robustness to sign changes because of noise in the measuring process. There are several ways to deal with sign changes because of noise, e.g., [26] uses maximum likelihood estimation; [27] uses a logistic function.

Note that both problems (5) and (6) are non-convex, and the algorithms BIHT/BIHT-ℓ_2 only approximately solve the problems. The convex model for robust 1bit-CS using the linear loss proposed in [9] is:

$$\begin{aligned} \min_{\mathbf{x}\in\mathbb{R}^n} \quad & -\frac{1}{m}\sum_{i=1}^m y_i(\mathbf{u}_i^T\mathbf{x}) \\ \text{s.t.} \quad & \|\mathbf{x}\|_2 \leq 1, \\ & \|\mathbf{x}\|_1 \leq s, \end{aligned} \tag{7}$$

where s is a given positive constant. The unit sphere constraint $\|\mathbf{x}\|_2 = 1$ is relaxed to the unit ball constraint $\|\mathbf{x}\|_2 \leq 1$, and the sparsity constraint $\|\mathbf{x}\|_0 \leq K$ is replaced by the ℓ_1 constraint $\|\mathbf{x}\|_1 \leq s$. Moreover, the one-sided ℓ_1-norm is replaced by a linear loss to avoid the trivial zero solution, and minimizing the linear loss can be explained as maximizing the correlation between y_i and $\mathbf{u}_i^T\mathbf{x}$. One can equivalently put the ℓ_1-norm in the objective function instead of in the constraint. The corresponding problem is given by [11]:

$$\begin{aligned} \min_{\mathbf{x}\in\mathbb{R}^n} \quad & \mu\|\mathbf{x}\|_1 - \frac{1}{m}\sum_{i=1}^m y_i(\mathbf{u}_i^T\mathbf{x}) \\ \text{s.t.} \quad & \|\mathbf{x}\|_2 \leq 1, \end{aligned} \tag{8}$$

where μ is the regularization parameter for the ℓ_1-norm. In the rest of this paper, we call (7) *Plan's model* and (8) *the passive model*. The latter comes from the name of the algorithm for (8) in [11]. Both problems (7) and (8) are convex, and there is a closed-form solution for (8).

III. The Pinball Loss for 1bit-CS

A. Pinball loss

In robust 1bit-CS models, the loss function plays an important role. According to the experiments in [11], Plan's model and the passive model, which both minimize the linear loss, perform much better than BIHT/BIHT-ℓ_2. However, the linear loss is quite rare in other classification tasks. To the best of our knowledge, among the existing classification methods, only the classical kernel rule [18], which enjoys computational effectiveness yet has bad classification accuracy generally, could be regarded as a support vector machine with the linear loss. This connection is recently discussed in [17].

To improve the performance from the one-sided ℓ_1-norm and the linear loss, we in this paper consider the pinball loss, which is defined in (2). Note that there are other equivalent formulations to define the pinball loss in [19] and [20]. The parameter τ is a key parameter for the pinball loss, and the one-sided ℓ_1-norm and the linear loss correspond to the cases $\tau = 0$ and $\tau = -1$, respectively.

If $\tau \geq -1$, $L_\tau(t)$ is convex. Thus, according to Theorem 2 of [16], one can verify that it is classification-calibrated, i.e., the function which minimizes the risk induced from $L_\tau(u)$ has the same sign as the Bayes rule. Furthermore, if τ is non-negative, it is proved in [22] that minimizing the pinball loss results in the Bayes rule. However, when $\tau \in [-1, 0)$, the pinball loss is not consistent to the Bayes rule because $L_\tau(u)$ with a negative τ is not lower-bounded. Thus, in most classification problems, the performance of a negative τ is not good, especially when τ is around -1. However the experiments on 1bit-CS conflict with the common sense: $\tau = -1$ leads to much better results than $\tau = 0$, implying that 1bit-CS has some special properties and motivating us to investigate the pinball loss with $\tau \in [-1, 0]$.

B. Pinball iterative hard thresholding

At this section, we replace the one-sided ℓ_1 norm in (5) with the pinball loss and expect that a better performance could be achieved. Specifically, we establish the following model,

$$\begin{aligned} \min_{\mathbf{x}\in\mathbb{R}^n} \quad & \frac{1}{m}\sum_{i=1}^{m} L_\tau\left(c - y_i(\mathbf{u}_i^T\mathbf{x})\right) \\ \text{s.t.} \quad & \|\mathbf{x}\|_2 = 1, \\ & \|\mathbf{x}\|_0 = K. \end{aligned} \tag{9}$$

Besides that a different loss function is considered, we also consider a bias term $c > 0$ in the loss. One can change c based on the measurement systems and even choose different c's for different measurements. However, for the sake of simplicity, we choose the same c for all measurements.

Minimization of a classification loss related to $c - y_i(\mathbf{u}_i^T\mathbf{x})$ makes data locate on the half-planes $y_i(\mathbf{u}_i^T\mathbf{x}) \geq c$. The margin between the two half-planes is given by $2c/\|\mathbf{x}\|_2$. In 1bit-CS, $\|\mathbf{x}\|_2$ is fixed to be 1, that means the margin is $2c$. Pursuing a large margin between two classes is helpful, especially when the data are noise-corrupted. In most SVM classifiers, c is set to be one. In BIHT, $c = 0$ and the loss function becomes the one-sided ℓ_1-norm of $-y_i(\mathbf{u}_i^T\mathbf{x})$. We will show the effect of c in robust 1bit-CS after introducing the algorithm for (9).

Replacing the subgradient of the one-sided ℓ_1 norm in BIHT with that of the pinball loss, we obtain *Pinball Iterative Hard Thresholding (PIHT)* for (9). The algorithm is summarized in Algorithm 1, where η_K stands for the best K-term approximation used in BIHT [4].

It is not hard to verify that $\mathbf{U}\mathbf{g}^l$ gives a subgradient of $\sum_{i=1}^m L_\tau\left(c - y_i(\mathbf{u}_i^T\mathbf{x}^l)\right)$, which is parallel to Lemma 5 in [6]. Same as BIHT, the convergence of PIHT cannot be guaranteed neither. The user needs to give a maximum number of iterations. Though BIHT lacks of convergence analysis, it shows good performance in noiseless 1bit-CS and is widely applied; see, e.g., [28], [29], [30].

We in this stage give a simple example to investigate the performance of the pinball loss for different τ and c values.

Experiment 1. We randomly generate a 1000-dimensional 10-sparse vector $\bar{\mathbf{x}}$, i.e., there are 10 non-zero components in $\bar{\mathbf{x}}$. The non-zero components are randomly selected and their values follow the standard Gaussian distribution. We take 500 binary measurements with $\mathbf{u}_i$ drawing from the standard Gaussian distribution as well. Here, we consider the noise-free case and 10% of the measurements are flipped.

Algorithm 1: Pinball Iterative Hard Thresholding (PIHT)

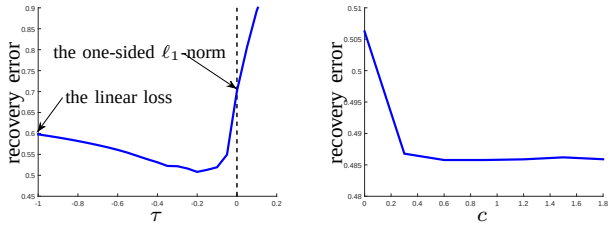
Set $\mathbf{x}^0, K, l_{\max}$, $\alpha > 0$, and $l := 0$
repeat
  Calculate $\mathbf{g}^l$ as

$$g_i^l = \begin{cases} -y_i, & y_i(\mathbf{u}_i^T\mathbf{x}^l) \leq c, \\ \tau y_i, & y_i(\mathbf{u}_i^T\mathbf{x}^l) > c; \end{cases} \tag{10}$$

  Update $\mathbf{a}^{l+1} = \mathbf{x}^l - \alpha\mathbf{U}\mathbf{g}^l$;
  Calculate $\mathbf{x}^{l+1} = \eta_K(\mathbf{a}^{l+1})$;
  $l := l + 1$;
until $l > l_{\max}$;
Return

$$\mathbf{x} = \mathbf{x}^l/\|\mathbf{x}^l\|_2.$$

Algorithm 1 is used to recover the signal, and the result is denoted as $\hat{\mathbf{x}}$. The step-size α is chosen as suggested in [6] and fixed. The average recovery error $\|\hat{\mathbf{x}} - \bar{\mathbf{x}}\|_2$ of 100 runs is used to measure the recovery performance. In Fig.1, the average recovery errors for different τ and c values are plotted. The performance corresponding to the one-sided ℓ_1-norm and the linear loss is marked. Generally, we can conclude that PIHT with a suitable negative τ improves the performance of BIHT. The performance of PIHT is not very sensitive to c when $c > 0.3$, and we suggest $c = 1$, which coincides with the loss used in most SVM classifiers.



(a) (b)

Fig. 1. Average recovery error of PIHT for different τ and c values. In this experiment, there is no noise in the measurements but 10% of the signs are flipped: (a) We set the bias term $c = 0$ and test different τ values; (b) We set $\tau = -0.2$ and evaluate the performance of different c values.

C. Properties of pinball loss and possible extensions

From Fig.1, we observe that a significantly better performance could be achieved by PIHT with a negative τ. We do not claim which τ value is the best, but one can find that the recovery performance is not monotonous with respect to τ. Generally, the pinball loss with a negative τ performs better than the hinge loss, i.e., the pinball loss with $\tau = 0$. This conflicts with the observation from many other classification tasks and motivates us to investigate special properties of 1bit-CS.

Consider $\mathbf{g}^l$ calculated by (10) for $\mathbf{x}^l$. When $\tau = 0$, i.e., the hinge loss is minimized, g_i^l is not zero only for the

observations satisfying $y_i(\mathbf{u}_i^T\mathbf{x}^l) \leq c$, i.e., these observations are not strictly correctly classified. Here we consider 1bit-CS as a binary classification problem, and by strictly, we mean that the analog measurement is not near zero. Because the hinge loss minimization is to minimize the summation of the distances to the decision boundary for the measurements that are not strictly correctly classified, the measurements that are strictly correctly classified do not contribute in the optimal solution. If we let $c = 0$ as in (5), then the true signal is optimal when there is no noise or sign flips because the objective value is lower bounded by 0, and the objective value for the true signal is 0. When there are sign changes in the measurements, then the objective value for the true signal is not 0 any more, and only the measurements with inconsistent signs, i.e., the sign of y_i is different from that of the $\mathbf{u}_i^T\mathbf{x}$, contribute to the optimal solution for (5). Thus many measurements are useless in determining the optimal solution.

The idea behind the linear loss and PIHT is to draw information from not only the incorrectly classified data, but also from the correctly classified ones. For example, when $\tau < 0$, $\mathbf{g}^l$ calculated by (10) encourages a larger $y_i(\mathbf{u}_i^T\mathbf{x}^l)$ when $y_i(\mathbf{u}_i^T\mathbf{x}^l) > c$ as well. Following this way, all measurements contribute to the final result, and the influence of sign flips and noise is weakened.

If we can detect the measurements with sign changes accurately, we can remove or replace them with the opposite values. In [10], *Adaptive Outlier Pursuit (AOP)* is designed to detect the sign flips during the transmission. Via adaptively detecting the measurements with sign flips, the performance of BIHT for 1bit-CS is significantly improved. We can also combine AOP and PIHT to improve the performance of PIHT. The new method is called AOP-PIHT. Because during the iterations, AOP detects the sign flips more and more accurately, the effect of τ should be decreased. We heuristically set $\tau = 0.95^{l_{\text{out}}}\tau_0$ in AOP-PIHT, where τ_0 is the initial τ value for the pinball loss and l_{out} is the counter of the outer loop in AOP-PIHT. We compare the performances of BIHT, PIHT, AOP-BIHT, and AOP-PIHT in the following two experiments.

Experiment 2. We randomly generate a 1000-dimensional 15-sparse vector $\bar{\mathbf{x}}$ in the same way as in Experiment 1. We compare PIHT, BIHT, AOP-BIHT, and AOP-PIHT for recovering the signal with different numbers of binary measurements ($m = 200, 300, \ldots, 1500$). Again there is no noise in the measurements but 10% of the signs are flipped. The average recovery errors of these four methods are shown in Fig.2.

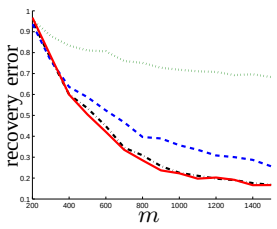


Fig. 2. The performance of BIHT (green dotted line), PIHT (blue dashed line), AOP-BIHT (black dot-dashed line), and AOP-PIHT (red solid line) for different numbers of measurements.

From Fig. 2, we can see that PIHT has a better performance than BIHT for all m before AOP is applied and both algorithms have very similar performances after AOP is applied. AOP is able to improve the performances of both algorithms, because it is able to detect most sign flips, and after the sign flips are corrected, the measurements are more accurate. However, before AOP is applied, PIHT is more robust than BIHT.

As mentioned in the previous sections, there are mainly two different sources of sign changes. Though AOP is able to detect random sign flips, sign changes because of noise are more difficult to detect. The next experiment shows that PIHT is able to deal with sign changes mainly because of noise. In the following experiment, the performances of BIHT, PIHT, AOP-BIHT, and AOP-PIHT are evaluated for two cases. Firstly, we consider the case with noise only, and then we consider the case with both noise and sign flips.

Experiment 3. We randomly generate a 1000-dimensional 15-sparse vector $\bar{\mathbf{x}}$ in the same way as in Experiment 1. Then we take 800 analog measurements and add noise with different Signal to Noise Ratio (SNR) values ($r_n = 1, \ldots, 50$) before the quantization. First we consider the case without sign flips, then we flip 10% of the measurements after the quantization. In Fig.3, the average recovery errors of BIHT, PIHT, AOP-BIHT, and AOP-PIHT are displayed by green dotted, blue dashed, black dot-dashed, and red solid line, respectively.

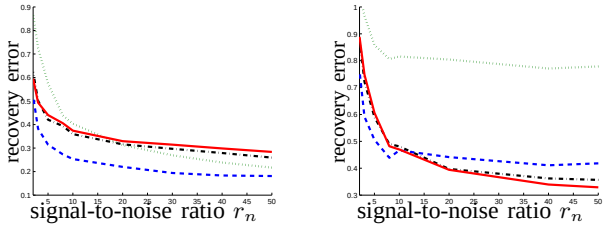


Fig. 3. The performances of BIHT (green dotted line), PIHT (blue dashed line), AOP-BIHT (black dot-dashed line), and AOP-PIHT (red solid line) for noisy data with different levels of noise. (a) no sign flip; (b) 10% of the measurements are flipped.

When noise is the main source of sign changes, e.g., no sign flips in Fig.3(a) and low SNRs in Fig.3(b), PIHT has the best performance among these four algorithms. AOP-BIHT and AOP-PIHT have similar performances, and their performances are worse than that of PIHT, i.e., AOP reduces the performance of PIHT when noise is the main source of sign changes. However, when sign changes happen mainly because of random sign flips, e.g., Fig.2 and high SNRs in Fig.3(b), AOP-PIHT has a better performance than PIHT. It confirms that the two different sources of sign changes have to consider differently and AOP is only suitable when the number of random sign flips is large.

The main purpose of this paper is to introduce the pinball loss for 1bit-CS. We leave modifications based on the pinball loss for the further work. In general, many advanced techniques for BIHT are also applicable for PIHT. Since minimizing the pinball loss instead of the hinge loss could improve

performances, one can naturally expect that modifications on PIHT, e.g., AOP-PIHT, will achieve better performances.

IV. Elastic-net Pin-SVM

A. Primal problem

In the above section, we replaced the one-sided ℓ_1 norm in BIHT with the pinball loss and established PIHT for robust 1bit-CS. Numerical experiments illustrate that PIHT performs better than BIHT. However, problem (9) that PIHT solves is non-convex, and there is no guarantee that PIHT converges to the global optimum of (9). In this section, we propose a convex model using the pinball loss and derive an algorithm to solve it. Specifically, the convex problem is

$$\begin{aligned} \min_{\mathbf{x}\in\mathbb{R}^n} \quad & \mu\|\mathbf{x}\|_1 + \frac{1}{m}\sum_{i=1}^{m} L_\tau(c - y_i(\mathbf{u}_i^T\mathbf{x})) \\ \text{s.t.} \quad & \|\mathbf{x}\|_2 \le 1. \end{aligned} \tag{11}$$

Here μ is a parameter to balance the regularization term $\|\mathbf{x}\|_1$ and the data loss term. We call (11) as an *elastic-net pin-SVM (ep-SVM)* because it involves both ℓ_1-norm and ℓ_2-norm.

When $\tau = -1$, the pinball loss becomes the linear loss, and (11) reduces to the passive model (8), for which there is a closed-form solution [11]. According to the experience on other classification tasks and the performance in Fig.1, we expect that a suitably selected τ may improve the performance. However, for all τ greater than -1, analytic solutions are not available and we need an efficient algorithm. We will introduce a coordinate ascent method to solve (11).

B. Dual problem

In order to obtain the dual problem of (11), we reformulate (11) into the following problem

$$\begin{aligned} \min_{\mathbf{x},\mathbf{e},\mathbf{z}} \quad & P(\mathbf{x},\mathbf{e},\mathbf{z}) := \mu\|\mathbf{e}\|_1 + \frac{1}{m}\sum_{i=1}^{m} L_\tau(z_i) + \iota(\mathbf{x}) \\ \text{s.t.} \quad & \mathbf{x} = \mathbf{e}, \\ & c - y_i(\mathbf{u}_i^T\mathbf{x}) = z_i,\ i = 1,2,\ldots,m, \end{aligned} \tag{12}$$

where $\iota(\mathbf{x})$ is the indicator function defined as

$$\iota(\mathbf{x}) = \begin{cases} 0, & \text{if } \|\mathbf{x}\|_2 \le 1, \\ +\infty, & \text{else.} \end{cases}$$

The corresponding Lagrangian is

$$\begin{aligned} \mathcal{L}(\mathbf{x},\mathbf{e},\mathbf{z},\beta,\xi) =& \mu\|\mathbf{e}\|_1 + \frac{1}{m}\sum_{i=1}^{m} L_\tau(z_i) + \iota(\mathbf{x}) \\ & + \beta^T(\mathbf{x}-\mathbf{e}) + \sum_{i=1}^{m}\xi_i(c - y_i(\mathbf{u}_i^T\mathbf{x}) - z_i). \end{aligned}$$

Then we can minimize $\mathcal{L}$ with respect to the primal variables $\{\mathbf{x},\mathbf{e},\mathbf{z}\}$ and obtain the dual problem of (12) as below,

$$\begin{aligned} \max_{\beta,\xi} \quad & D(\beta,\xi) := c\sum_{i=1}^{m}\xi_i - \left\|\sum_{i=1}^{m}\xi_i y_i\mathbf{u}_i - \beta\right\|_2 \\ \text{s.t.} \quad & \|\beta\|_\infty \le \mu, \\ & -\frac{\tau}{m} \le \xi_i \le \frac{1}{m},\ i = 1,2,\ldots,m. \end{aligned} \tag{13}$$

Assume that we solve the dual problem and obtain optimal β^* and ξ^*, then we can find the optimal $\mathbf{x}^*$ for (11) as follows:

1) If $\sum_{i=1}^m \xi_i^* y_i\mathbf{u}_i - \beta^* \neq \mathbf{0}$, i.e., $\|\sum_{i=1}^m \xi_i^* y_i\mathbf{u}_i\|_\infty > \mu$, the optimal $\mathbf{x}^*$ can be obtained as

$$\mathbf{x}^* = \left(\sum_{i=1}^{m}\xi_i^* y_i\mathbf{u}_i - \beta^*\right) \Big/ \left\|\sum_{i=1}^{m}\xi_i^* y_i\mathbf{u}_i - \beta^*\right\|_2.$$

2) If $\sum_{i=1}^m \xi_i^* y_i\mathbf{u}_i - \beta^* = \mathbf{0}$, i.e., $\|\sum_{i=1}^m \xi_i^* y_i\mathbf{u}_i\|_\infty \le \mu$. The optimal $\mathbf{x}^*$ may not be unique, and all $\mathbf{x}^*$ that satisfies

$$\begin{aligned} \|\mathbf{x}^*\|_2 &\le 1, && \text{(14a)} \\ x_j^* &= 0, \quad \text{if } |\beta_j^*| < \mu, && \text{(14b)} \\ x_j^* &\ge 0, \quad \text{if } \beta_j^* = \mu, && \text{(14c)} \\ x_j^* &\le 0, \quad \text{if } \beta_j^* = -\mu, && \text{(14d)} \\ c - y_i(\mathbf{u}_i^T\mathbf{x}^*) &\ge 0, \quad \text{if } \xi_i^* = 1/m, && \text{(14e)} \\ c - y_i(\mathbf{u}_i^T\mathbf{x}^*) &\le 0, \quad \text{if } \xi_i^* = -\tau/m, && \text{(14f)} \\ c - y_i(\mathbf{u}_i^T\mathbf{x}^*) &= 0, \quad \text{if } \xi_i^* \in (-\tau/m, 1/m), && \text{(14g)} \end{aligned}$$

are optimal.

Remark 1: If $\left\|\sum_{i=1}^m \frac{1}{m} y_i\mathbf{u}_i\right\|_\infty \le \mu$, we have $\xi_i^* = 1/m$. Then it is easy to check that $\mathbf{x}^* = \mathbf{0}$ is optimal. This generalizes the result for the passive model [11, Lemma 1]. When $\tau = -1$, there is no constraints (14e)-(14g), and any $\mathbf{x}^*$ satisfying (14a)-(14d) is optimal.

Let us define two hypercubes

$$\begin{aligned} \mathcal{A} &= \left\{\mathbf{z} = \sum\nolimits_{i=1}^m \xi_i y_i\mathbf{u}_i \,\middle|\, -\frac{\tau}{m} \le \xi_i \le \frac{1}{m},\ \forall i\right\}, \\ \mathcal{B} &= \left\{\mathbf{z} \,\middle|\, -\mu \le z_i \le \mu,\ \forall i\right\}. \end{aligned}$$

If $\mathcal{A}\bigcap\mathcal{B} = \emptyset$, then the optimal $\mathbf{x}^*$ will always be on the unit sphere. Even if $\mathcal{A}\bigcap\mathcal{B} \neq \emptyset$, we may still have $\|\sum_{i=1}^m \xi_i^* y_i\mathbf{u}_i\|_\infty > \mu$ when $c > 0$, and the optimal $\mathbf{x}^*$ is on the unit sphere. However, if $c = 0$, the optimal dual objective is 0 when $\mathcal{A}\cap\mathcal{B} \neq \emptyset$, and the the primal objective becomes zero when $\mathbf{x} = \mathbf{0}$, so $\mathbf{x}^* = \mathbf{0}$ is optimal for the primal problem.

In order to get the optimal $\mathbf{x}^*$ on the unit sphere, we can choose smaller μ because smaller μ implies smaller $\mathcal{B}$ and $\mathcal{A}\cap\mathcal{B}$ becomes smaller.

C. Dual coordinate ascent algorithm

In the dual problem (13), the constraints are separable and we can apply the coordinate ascent method efficiently. The subproblems are:

1) β_j-subproblem: $D(\beta,\xi)$ is separable with respect to β, so β_j can be computed in parallel via

$$\tilde{\beta}_j = \max\left\{-\mu, \min\left\{\mu, \left(\sum_{i=1}^{m}\xi_i y_i\mathbf{u}_i\right)_j\right\}\right\}. \tag{15}$$

2) ξ_i-subproblem: Let us consider $\tilde{\xi}_i = \xi_i + d_i$. Then it becomes the following optimization problem on d_i:

$$\max_{-\frac{\tau}{m} \le \xi_i + d_i \le \frac{1}{m}} cd_i - \left\|y_i\mathbf{u}_i d_i + \sum_{i=1}^{m}\xi_i y_i\mathbf{u}_i - \beta\right\|_2. \tag{16}$$

Denote $\mathbf{w} = \sum_{i=1}^m \xi_i y_i \mathbf{u}_i - \beta$. Problem (16) becomes

$$\max_{-\frac{\tau}{m} \le \xi_i + d_i \le \frac{1}{m}} cd_i - \sqrt{\|\mathbf{u}_i\|_2^2 d_i^2 + 2y_i \mathbf{u}_i^T \mathbf{w} d_i + \|\mathbf{w}\|_2^2}$$

The optimal solution d_i^* can be calculated analytically as following:

- If $\|\mathbf{u}_i\|_2 \le c$, the objective function is non-increasing. We have that $d_i^* = 1/m - \xi_i$ is optimal, i.e., $\tilde{\xi}_i = 1/m$.
- If $\|\mathbf{u}_i\|_2 > c$, we have

$$d_i^* = \max\left\{-\frac{\tau}{m} - \xi_i, \min\left\{\frac{1}{m} - \xi_i, \bar{d}_i\right\}\right\}, \quad (17)$$

where

$$\bar{d}_i = \frac{-B_d + \sqrt{B_d^2 - 4A_d C_d}}{2A_d} \quad (18)$$

with

$$\begin{aligned} A_d &= \|\mathbf{u}_i\|_2^2(\|\mathbf{u}_i\|_2^2 - c^2), \\ B_d &= 2(\|\mathbf{u}_i\|_2^2 - c^2) y_i \mathbf{u}_i^T \mathbf{w}, \\ C_d &= (\mathbf{u}_i^T \mathbf{w})^2 - c^2 \|\mathbf{w}\|_2^2. \end{aligned}$$

Summarizing the previous discussion, we give the dual coordinate ascent method for (11) in Algorithm 2.

Algorithm 2: Dual coordinate ascent for ep-SVM

```
Set l = 0, β^0 = 0_{n×1}, ξ^0 = -(τ/m) 1_{m×1};
Calculate w = Σ_{i=1}^m ξ_i y_i u_i - β
repeat
    for i = 1, 2, ..., m do
        if c ≥ ||u_i||_2 then
            d_i^* = 1/m - ξ_i^l;
        else
            Calculate d̄_i by (18) and d_i^* by (17);
        end
        if d_i^* ≠ 0 then
            w = w + y_i u_i d_i^*;
            ξ_i^{l+1} = ξ_i^l + d_i^*;
        end
    end
    Calculate β^{l+1} by (15);
    l := l + 1;
until ξ^l = ξ^{l-1} ;
if ||w||_2 > 0 then
    x = w / ||w||_2;
else
    Find x that satisfies (14);
end
```

When $\tau = -1$, the algorithm ends in one iteration because $\xi_i = 1/m$ for all i. Thus, there is an analytical solution for the passive model. For other τ values, there is no analytical solution. However, the next theorem states that the output $\mathbf{x}^*$ of Algorithm 2 is optimal for (11).

Theorem 1: The dual coordinate ascent for ep-SVM (Algorithm 2) converges to an optimal solution of (11).

Proof: The optimality condition for (11) is

- If $\|\mathbf{x}\|_2 = 1$, there exists ξ satisfying

$$\xi_i \begin{cases} = 1/m & \text{if } c - y_i(\mathbf{u}_i^T \mathbf{x}) > 0 \\ \in [-\tau/m, 1/m] & \text{if } c - y_i(\mathbf{u}_i^T \mathbf{x}) = 0 \\ = -\tau/m & \text{if } c - y_i(\mathbf{u}_i^T \mathbf{x}) < 0 \end{cases} \quad (19)$$

  such that $\mu\|\mathbf{x}\|_1 - \sum_{i=1}^m \xi_i y_i(\mathbf{u}_i^T \mathbf{x}) \ge 0$.
- If $\|\mathbf{x}\|_2 < 1$, there exists ξ satisfying

$$\xi_i \begin{cases} = 1/m & \text{if } c - y_i(\mathbf{u}_i^T \mathbf{x}) > 0 \\ \in [-\tau/m, 1/m] & \text{if } c - y_i(\mathbf{u}_i^T \mathbf{x}) = 0 \\ = -\tau/m & \text{if } c - y_i(\mathbf{u}_i^T \mathbf{x}) < 0 \end{cases} \quad (20)$$

  such that $\mu\|\mathbf{x}\|_1 - \sum_{i=1}^m \xi_i y_i(\mathbf{u}_i^T \mathbf{x}) = 0$.

Then we show that the output of Algorithm 2 satisfies the optimality condition.

Case 1 ($\mathbf{w} \neq \mathbf{0}$): We have $\|\mathbf{x}^*\| = 1$ and the algorithm shows that $\{\beta_j^*\}$ and $\{\xi_i^*\}$ are coordinate maximum of (13). Thus we have

- If $\xi_i^* = 1/m$, we have $y_i(\mathbf{u}_i^T \mathbf{x}^*) \le c$.
- If $\xi_i^* = -\tau/m$, we have $y_i(\mathbf{u}_i^T \mathbf{x}^*) \ge c$.
- If $\xi_i^* \in (-\tau/m, 1/m)$, we have $y_i(\mathbf{u}_i^T \mathbf{x}^*) = c$.
- If $|\left(\sum_{i=1}^m \xi_i^* y_i \mathbf{u}_i\right)_j| < \mu$, we have $x_j^* = 0$.
- If $\left(\sum_{i=1}^m \xi_i^* y_i \mathbf{u}_i\right)_j \ge \mu$, we have $x_j^* \ge 0$.
- If $\left(\sum_{i=1}^m \xi_i^* y_i \mathbf{u}_i\right)_j \le -\mu$, we have $x_j^* \le 0$.

Therefore, ξ^* satisfy (19), and we have

$$\sum\nolimits_{i=1}^m \xi_i^* y_i(\mathbf{u}_i^T \mathbf{x}^*) = \left(\sum\nolimits_{i=1}^m \xi_i^* y_i \mathbf{u}_i\right)^T \mathbf{x}^* \ge \mu\|\mathbf{x}^*\|_1,$$

which means that $\mathbf{x}^*$ is optimal.

Case 2 ($\mathbf{w} = \mathbf{0}$): We have $\|\mathbf{x}^*\| \le 1$, and $\{\beta_j^*\}$ and $\{\xi_i^*\}$ being coordinate maximum of (13) tells us that

- If $\xi_i^* = 1/m$, we have $\|y_i \mathbf{u}_i\|_2 \le c$.
- If $\xi_i^* = -\tau/m$, we have $\|y_i \mathbf{u}_i\|_2 \ge c$.
- If $\xi_i^* \in (-\tau/m, 1/m)$, we have $\|y_i \mathbf{u}_i\|_2 = c$.
- If $|\left(\sum_{i=1}^m \xi_i^* y_i \mathbf{u}_i\right)_j| < \mu$, i.e., $|\beta_j^*| < \mu$, we have $x_j^* = 0$.
- If $\left(\sum_{i=1}^m \xi_i^* y_i \mathbf{u}_i\right)_j = \mu$, i.e., $\beta_j^* = \mu$, we have $x_j^* \ge 0$.
- If $\left(\sum_{i=1}^m \xi_i^* y_i \mathbf{u}_i\right)_j = -\mu$, i.e., $\beta_j^* = -\mu$, we have $x_j^* \le 0$.

Therefore, for any $\mathbf{x}^*$ satisfying (14), ξ^* satisfies (20), and we have

$$\sum\nolimits_{i=1}^m \xi_i^* y_i(\mathbf{u}_i^T \mathbf{x}^*) = \left(\sum\nolimits_{i=1}^m \xi_i^* y_i \mathbf{u}_i\right)^T \mathbf{x}^* = \mu\|\mathbf{x}^*\|_1,$$

which means that $\mathbf{x}^*$ is optimal. ■

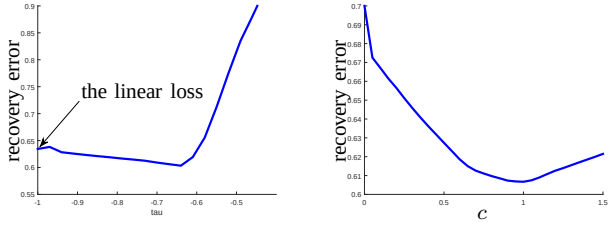
Remark 2: Both the proof of Theorem 1 and Algorithm 2 suggest that if $c \ge \|\mathbf{u}_i\|_2$ for all i, then $\xi_i^* = 1/m$, and ep-SVM reduces to the passive model no matter what τ is. It happens because $c - y_i(\mathbf{u}_i^T \mathbf{x}) \ge 0$ if $\|\mathbf{x}\|_2 \le 1$. Therefore, we choose c to be much smaller than most $\|\mathbf{u}_i\|_2$. In all the experiments in this paper, $\mathbf{u}_i$ has the same dimension ($n = 1000$), and they are generated in the same way. So we choose the same c.

In practice, we can set a maximum number of iterations $l_{\max}$ and choose $\|\xi^l - \xi^{l-1}\|_\infty < \delta$ as the stopping criterion. Here δ is a small positive number. In the following experiments, we set $l_{\max} = 100$ and $\delta = (1 + \tau)/(10m)$.

D. Selection of τ, c, and μ

Though the passive model has an analytical solution, the linear loss is not a good classification loss in regular classification problems and 1bit-CS. Thus we choose the pinball loss with $\tau > -1$. In order to evaluate the improvement using ep-SVM with different τ values, we consider an experiment similar to Experiment 1.

Experiment 4. We randomly generate a 1000-dimensional 15-sparse vector $\bar{\mathbf{x}}$ in the same way as in Experiment 1. Then we take 300 binary measurements and flip 10% of them. Ep-SVM (11) with different τ and c values are evaluated. For the regularization coefficient μ, we choose $\mu = \sqrt{\frac{\log n}{m}}$, as suggested in [11]. The experiments are repeated 100 times and the average recovery error is plotted in Fig.4.



(a) (b)

Fig. 4. Average recovery error of ep-SVM for different τ and c values. (a) Set the bias term $c = 1$ and evaluate the performances of different τ values; (b) Set $\tau = -0.7$ and evaluate the performances of different c values.

This experiment has similar results as Experiment 1. The recovery error is not monotonous with respect to τ, and a suitable τ value, e.g., $\tau = -0.7$ in this experiment, leads to a better result. The performances of different c values in Fig.4(b) suggest that $c = 1$ is a good choice for this measurement system.

It is possible that different μ values are suitable for different τ values. In the following, we set $\mu = C\sqrt{\frac{\log n}{m}}$ and consider the performances with different C and τ values. The corresponding average recovery error is shown by a contour map in Fig.5. The curves represent the level sets, and the colors stand for the recovery error. Generally, when $\tau = -1$, the suitable C is around 1, which is also the suggestion of [11]. If a larger τ is used, the corresponding suitable C is smaller. The relationship between τ and μ is problem-dependent. For practical use, we suggest $\tau = -0.5$ and $\mu = 0.7\sqrt{\frac{\log n}{m}}$ for ep-SVM. In the numerical experiments, we will evaluate other parameter values as well.

V. Numerical Experiments

A. Known sparsity

In the previous sections, we introduced the pinball loss for robust 1bit-CS and established two models and two corresponding algorithms. Several simple experiments illustrate that the pinball loss minimization helps us improve the recovery performance for 1bit-CS. In this section, we further evaluate the performance of the pinball loss in more experiments with different noise levels and different numbers of measurements.

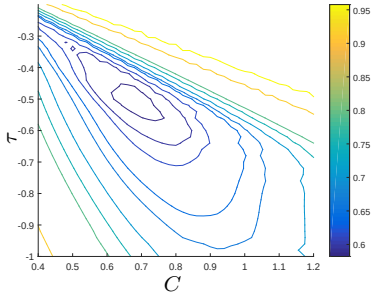
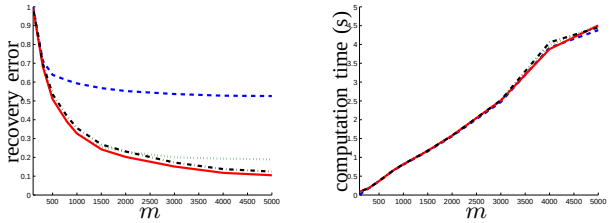


Fig. 5. Contour map of the average recovery error of ep-SVM for different τ and μ values, where $\mu = C\sqrt{\frac{\log n}{m}}$. The experiment parameters are as the same as those in Fig.4.

To highlight the main purpose of this paper, i.e., using a new loss function for 1bit-CS, we do not consider advanced techniques such as AOP. As shown in Fig.2, suitably applying those techniques for pinball loss minimization can further improve the performance.

First, assume that the sparsity is known in advance. Then (5) and (9) are applicable to recover the signal. We solve them by BIHT[1] and Algorithm 1, respectively. Note that there is no stopping criterion for both BIHT and PIHT. We set the maximum number of iterations to 500 for both of them. Though the experiment shown in Fig.1 implies that PIHT with $\tau = -0.2$ is a good choice. We also evaluate the performance for $\tau = -0.1$ and -0.4. The data are generated following the same way of Experiments 1–4: we have m one-bit measurements and try to recover a n dimensional signal with K-sparsity. The sign flip ratio is r_f and the SNR in measurements is r_n. All the results below is the average value for repeating 100 times. The experiments are done with Matlab 2013a on Core i5-1.80GHz, 4.0GB. The source code for Algorithm 1 and Algorithm 2 can be found on the authors' homepages.

To test the performance of BIHT and PIHT for different numbers of measurements, we select $n = 1000$, $r_f = 10\%$, $r_n = \infty$, $K = 20$ and vary m from 100 to 5000. Fig.6 displays the performances for BIHT and PIHT with different τ values. Compared with BIHT, using a negative τ improves the performance significantly with similar computational time. The good performance of $\tau = -0.2$ is again confirmed.



(a) (b)

Fig. 6. The performances of BIHT (blue dashed line) and PIHT with $\tau = -0.1$ (green dotted line), $\tau = -0.2$ (red solid line), and $\tau = -0.4$ (black dot-dashed line). In this experiment, $n = 1000, r_f = 10\%, K = 20$, and $r_n = \infty$: (a) recovery error vs. m; (b) computational time vs. m.

[1]http://perso.uclouvain.be/laurent.jacques/index.php/Main/BIHTDemo

In the previous experiment, we assumed that there is no noise and considered only sign flips. Next we consider the performance of PIHT for different SNRs with a fixed sign flip ratio. The average recovery error is shown by Fig.7. The performances again suggest $\tau = -0.2$ for this measurement system.

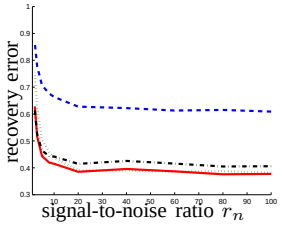


Fig. 7. The recovery error for different noise-levels with $n = 1000, m = 800, r_f = 10\%$, and $K = 20$. The result of BIHT is shown by blue dashed line, PIHT with $\tau = -0.1$ is shown by green dotted line, PIHT with $\tau = -0.2$ by red solid line, and PIHT with $\tau = -0.4$ by black dot-dashed line.

When the sparsity of the true signal is not known in advance yet we have an estimation of the sparsity, we may still apply BIHT and PIHT with this estimation. However, the performance will be reduced, as shown in the next experiment. In order to test the performances of PIHT with different estimations of the sparsity of the signal, we fix the sparsity of the true signal as 20, but use different K values in BIHT and PIHT. In Fig.8, we can observe that if the estimation on the sparsity is accurate, PIHT gives good recovery performance. But if the gap between the estimation and the real sparsity is large, the performance of PIHT becomes bad. This experiment shows that an accurate estimation is necessary for PIHT.

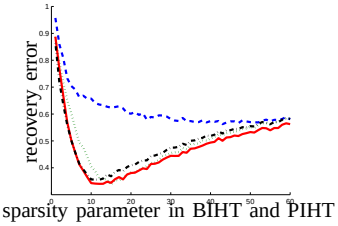


Fig. 8. The recovery error for different sparsity estimations. The sparsity of the true signal is 20, other parameters for this experiment are $n = 1000, m = 800, r_n = \infty$, and $r_f = 10\%$. The result of BIHT is shown by blue dashed line, PIHT with $\tau = -0.1$ is shown by green dotted line, PIHT with $\tau = -0.2$ by red solid line, and PIHT with $\tau = -0.4$ by black dot-dashed line.

B. Unknown sparsity

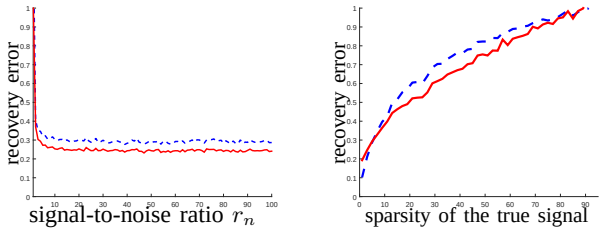
In general, the sparsity of the true signal is not known or the true signal is only approximately sparse. In these cases, the performance of PIHT is reduced if the estimation is not correct, and we can consider Plan's model, the passive model, or the proposed ep-SVM. Plan's model and the passive model use the same loss function, and they have similar recovery errors, according to the numerical study in [11]. Since there is an efficient algorithm for the passive model, we in this paper only compare ep-SVM with the passive model.

In the passive model and ep-SVM, there is a regularization coefficient μ. As suggested by [11], we set $\mu = \sqrt{\frac{\log n}{m}}$ for the passive model. For ep-SVM, the suitable μ value depends on τ, as illustrated by Fig.5. Heuristically, we let $\mu = C\sqrt{\frac{\log n}{m}}$, where C is given below,

τ	-0.4	-0.5	-0.7	-0.9	-1.0
C	0.6	0.7	0.8	0.9	1.0

Let $n = 1000, K = 20, r_n = \infty, r_f = 10\%$, and vary the number of measurements from 200 to 2000. The average recovery errors and the computational time for different τ values are listed in Table I. The smallest average recovery error for each m is highlighted in bold font. The results imply that $\tau = -0,5, \mu = 0.7\sqrt{\frac{\log n}{m}}$ is a promising choice. Though there is no analytic solution when $\tau > -1$, the computational time in Table I shows that Algorithm 2 is very fast. Furthermore, we test the performance of Algorithm 2 for noise-corrupted data ($r_n = 10$), and the corresponding results are reported in Table II.

From the results listed in Table I and II, we observe that minimizing the pinball loss can improve the accuracy of the linear loss. The computational time is monotonically increasing with respect to τ, and generally Algorithm 2 is efficient. In practice, we suggest $\tau = -0.5$ and $\mu = 0.7\sqrt{\frac{\log n}{m}}$ for ep-SVM. In the following, we use this parameter set for different sparsity levels and different SNRs. The results are shown in Fig.9, from which one can observe the improvement of pinball loss minimization.



(a) (b)

Fig. 9. The performances of the passive model (blue dashed line) and ep-SVM (red solid line). (a) recovery accuracy vs. SNR ($n = 1000, m = 2000, K = 20, r_f = 10\%$); (b) recovery accuracy vs. sparsity of the true signal ($n = 1000, m = 500, r_n = 20, r_f = 10\%$).

VI. Conclusion

In 1bit-CS, one recovers a signal from a set of sign measurements. It can also be regarded as a binary classification problem, for which the hinge loss enjoys many good properties. However, the linear loss performs better than the hinge loss in robust 1bit-CS. Thus, a trade-off between them is expected to share their good properties and improve the recovery performance for 1bit-CS. We introduce the pinball loss, which is a trade-off between the hinge loss and the linear loss. We proposed two model for minimizing the pinball loss and the two algorithms to solve them. PIHT improves the performance of the BIHT proposed in [6] and is suitable for the case when the sparsity of the true signal is given. Ep-SVM generalizes the passive model proposed in [11] and is

	m	200	350	500	650	800	1100	1400	1700	2000
$\tau = -0.40$	error	0.889	0.599	0.498	0.437	0.398	0.338	0.292	0.260	0.245
	time (ms)	74.8	366	375	374	381	418	513	562	634
$\tau = -0.50$	error	0.850	**0.582**	**0.495**	**0.430**	**0.390**	**0.329**	**0.287**	**0.251**	**0.235**
	time (ms)	141	298	295	303	332	386	471	579	629
$\tau = -0.70$	error	**0.796**	0.602	0.518	0.450	0.405	0.344	0.301	0.269	0.242
	time (ms)	135	169	176	205	249	304	385	414	456
$\tau = -0.90$	error	0.820	0.633	0.540	0.479	0.430	0.368	0.324	0.289	0.258
	time (ms)	63.0	86.8	108	131	152	193	248	278	311
$\tau = -1.00$	error	0.837	0.657	0.558	0.504	0.451	0.392	0.345	0.309	0.274
(passive algorithm)	time (ms)	5.20	6.23	7.90	10.0	10.5	13.9	13.5	15.1	17.7

TABLE I
RECOVERY ERROR AND COMPUTATIONAL TIME OF EP-SVM

	m	200	350	500	650	800	1100	1400	1700	2000
$\tau = -0.40$	error	0.949	0.676	0.549	0.471	0.418	0.361	0.315	0.282	0.252
	time (ms)	50.6	297	418	409	446	467	537	591	709
$\tau = -0.50$	error	0.906	**0.648**	**0.541**	**0.460**	**0.404**	**0.348**	**0.305**	**0.274**	**0.243**
	time (ms)	56.7	317	341	332	375	425	503	561	637
$\tau = -0.70$	error	**0.821**	0.663	0.563	0.479	0.418	0.361	0.320	0.285	0.252
	time (ms)	146	169	185	235	267	332	405	438	470
$\tau = -0.90$	error	0.840	0.689	0.596	0.507	0.442	0.384	0.343	0.305	0.269
	time (ms)	66.5	90.9	111	137	172	198	243	275	304
$\tau = -1.00$	error	0.855	0.707	0.622	0.534	0.462	0.405	0.364	0.324	0.287
(passive algorithm)	time (ms)	2.50	4.60	6.15	7.57	8.91	11.5	13.4	16.5	19.0

TABLE II
RECOVERY ERROR AND COMPUTATIONAL TIME OF EP-SVM FOR NOISE-CORRUPTED DATA ($r_n = 10$).

suitable for the case when the sparsity of the true signal is not given. A fast dual coordinate ascent algorithm is proposed to solve ep-SVM, and its convergence is proved. The numerical experiments demonstrate that the pinball loss, as a trade-off between the hinge loss and the linear loss, improves the existing 1bit-CS models with better performances. In the future, we will investigate other advanced methods based on the pinball loss minimization. The related statistical properties in view of learning are also interesting for study.

REFERENCES

[1] E. J. Candès and T. Tao, "Decoding by linear programming," *IEEE Transactions on Information Theory*, vol. 51, no. 12, pp. 4203–4215, 2005.

[2] D. L. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[3] Y. C. Eldar and G. Kutyniok, *Compressed Sensing: Theory and Applications*. Cambridge University Press, Cambridge, 2012.

[4] P. T. Boufounos and R. G. Baraniuk, "1-bit compressive sensing," in *Proceedings the 42nd Annual Conference on Information Sciences and Systems (CISS)*, 2008, pp. 16–21.

[5] J. N. Laska, Z. Wen, W. Yin, and R. G. Baraniuk, "Trust, but verify: Fast and accurate signal recovery from 1-bit compressive measurements," *IEEE Transactions on Signal Processing*, vol. 59, no. 11, pp. 5289–5301, 2011.

[6] L. Jacques, J. N. Laska, P. T. Boufounos, and R. G. Baraniuk, "Robust 1-bit compressive sensing via binary stable embeddings of sparse vectors," *IEEE Transactions on Information Theory*, vol. 59, no. 4, pp. 2082–2102, 2013.

[7] P. T. Boufounos, "Greedy sparse signal reconstruction from sign measurements," in *Proceedings of the 43rd Asilomar Conference on Signals, Systems and Computers*, 2009, pp. 1305–1309.

[8] Y. Xu, Y. Kabashima, and L. Zdeborová, "Bayesian signal reconstruction for 1-bit compressed sensing," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2014, no. 11, P11015, 2014.

[9] Y. Plan and R. Vershynin, "Robust 1-bit compressed sensing and sparse logistic regression: A convex programming approach," *IEEE Transactions on Information Theory*, vol. 59, no. 1, pp. 482–494, 2013.

[10] M. Yan, Y. Yang, and S. Osher, "Robust 1-bit compressive sensing using adaptive outlier pursuit," *IEEE Transactions on Signal Processing*, vol. 60, no. 7, pp. 3868–3875, 2012.

[11] L. Zhang, J. Yi, and R. Jin, "Efficient algorithms for robust one-bit compressive sensing," in *Proceedings of the 31st International Conference on Machine Learning (ICML)*, 2014, pp. 820–828.

[12] P. T. Boufounos, "Reconstruction of sparse signals from distorted randomized measurements," in *Proceedings of 2010 IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP)*, pp. 3998–4001.

[13] J. N. Laska and R. G. Baraniuk, "Regime change: Bit-depth versus measurement-rate in compressive sensing," *IEEE Transactions on Signal Processing*, vol. 60, no. 7, pp. 3496–3505, 2012.

[14] V. Vapnik, *Statistical Learning Theory*. Wiley, New York, 1998.

[15] T. Zhang, "Statistical analysis of some multi-category large margin classification methods," *The Journal of Machine Learning Research*, vol. 5, pp. 1225–1251, 2004.

[16] P. L. Bartlett, M. I. Jordan, and J. D. McAuliffe, "Convexity, classification, and risk bounds," *Journal of the American Statistical Association*, vol. 101, no. 473, pp. 138–156, 2006.

[17] X. Huang, L. Shi, and J.A.K. Suykens, "Solution path for pin-SVM classifiers with positive and negative τ values," *Internal Report 14-123, ESAT-SISTA, KU Leuven*, 2014.

[18] L. Györfi, L. Devroye, and G. Lugosi, *A Probabilistic Theory of Pattern Recognition*, Springer, New York, 1996.

[19] R. Koenker, *Quantile Regression*. Cambridge University Press, Cambridge, 2005.

[20] A. Christmann and I. Steinwart, "How SVMs can estimate quantiles and the median," in *Advances in Neural Information Processing Systems*, 2007, pp. 305–312.

[21] I. Steinwart and A. Christmann, "Estimating conditional quantiles with the help of the pinball loss," *Bernoulli*, vol. 17, no. 1, pp. 211–225, 2011.

[22] X. Huang, L. Shi, and J.A.K. Suykens, "Support vector machine classifier with pinball loss," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 5, pp. 984–997, 2014.

[23] Y. Plan and R. Vershynin, "One-bit compressed sensing by linear programming," *Communications on Pure and Applied Mathematics*, vol. 66, no. 8, pp. 1275–1297, 2013.

[24] F. Cucker and D. X. Zhou, *Learning Theory: an Approximation Theory Viewpoint*. Cambridge University Press, Cambrige, 2007.

[25] I. Steinwart and A. Christmann, *Support Vector Machines*. Springer, New York, 2008.

[26] S. Bahmani, P. T. Boufounos, and B. Raj, "Robust 1-bit compressive sensing via gradient support pursuit," *arXiv preprint arXiv:1304.6627*, 2013.

[27] J. Fang, Y. Shen, H. Li, and Z. Ren, "Sparse signal recovery from one-

bit quantized data: an iterative reweighted algorithm," *Signal Processing*, vol. 102, no. 0, pp. 201 – 206, 2014.

[28] A. Movahed, A. Panahi, and G. Durisi, "A robust RFPI-based 1-bit compressive sensing reconstruction algorithm," in *Proceedings of 2012 IEEE Information Theory Workshop (ITW)*, 2012, pp. 567–571.

[29] D. Lee, T. Sasaki, T. Yamada, K. Akabane, Y. Yamaguchi, and K. Uehara, "Spectrum sensing for networked system using 1-bit compressed sensing with partial random circulant measurement matrices," in *Proceedings of the 75th IEEE Vehicular Technology Conference (VTC Spring)*, 2012, pp. 1–5.

[30] C. E. Luo, "A low power self-capacitive touch sensing analog front end with sparse multi-touch detection," in *Proceedings of 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 3007–3011.