Efficient Sparse Recovery via Adaptive Non-Convex Regularizers with Oracle Property

Ming Lin *
Department of Automation
Tsinghua University
Beijing, P.R. China 100084

Rong Jin
Computer Science and Engineering
Michigan State University
East Lansing, MI, USA 48823

Changshui Zhang
Department of Automation
Tsinghua University
Beijing, P.R. China 100084

Abstract

The main shortcoming of sparse recovery with a convex regularizer is that it is a biased estimator and therefore will result in a suboptimal performance in many cases. Recent studies have shown, both theoretically and empirically, that non-convex regularizer is able to overcome the biased estimation problem. Although multiple algorithms have been developed for sparse recovery with non-convex regularization, they are either computationally demanding or not equipped with the desired properties (i.e. optimal recovery error, selection consistency and oracle property). In this work, we develop an algorithm for efficient sparse recovery based on proximal gradient descent. The key feature of the proposed algorithm is introducing adaptive non-convex regularizers whose shrinking threshold vary over iterations. The algorithm is compatible with most popular non-convex regularizers, achieves a geometric convergence rate for the recovery error, is selection consistent, and most importantly has the oracle property. Based on the proposed framework, we suggest to use a so–called ACCQ regularizer, which is equivalent to zero proximal projection gap adaptive hard-thresholding. Experiments with both synthetic data sets and real images verify both the efficiency and effectiveness of the proposed method compared to the state-of-the-art methods for sparse recovery.

1 INTRODUCTION

Inspired by the seminal work of compressive sensing (Candès et al., 2006), numerous algorithms have been developed to recover a sparse vector from its linear low dimensional measurement. Most of these algorithms can be classified into two categories: greedy methods and optimization–based methods. Greedy methods aggressively select the support set as they recover the target sparse vector ((Tropp and Wright, 2010) and references therein). Although they are computationally efficient, the greedy methods are usually sensitive to noise especially when the target signal is not exactly sparse. Optimization–based methods, on the other hand, are known to be more robust to noise but at the price of a higher computational cost (Zou and Hastie, 2005; Rosenbaum and Tsybakov, 2010; Xu et al., 2010).

Most optimization–based methods cast sparse recovery into convex optimization problems. The most well known algorithms in this category are LASSO (Tibshirani, 1996; Efron et al., 2004) and Dantzig selector (Candes and Tao, 2007). A main drawback of convex optimization based methods for sparse recovery is that they are biased estimators, i.e. the solutions found by the convex optimization based methods do not have the *oracle property* (Fan and Li, 2001; Zhang and Zhang, 2011), which is sometime referred to as *Lasso bias* (Zhang and Zhang, 2011). We note that there are two different versions of oracle property used in the literature: the asymptotical one that examines the oracle property with the number of measurements going to infinity (Fan and Li, 2001) and the finite sample one (Zhang and Zhang, 2011) that examines the oracle property with a finite number of measurements. In this study, we will use the finite sample version of oracle property.

It was first suggested in (Fan and Li, 2001) that the Lasso bias can be corrected by a non–convex regularizer. Several theory and algorithms have been developed recently for sparse recovery using concave regularizers (Zhang and Zhang, 2011; Zhang, 2012; Loh and Wainwright, 2013), and their effectiveness for sparse recovery has been verified empirically by several recent studies (Xiang et al., 2013; Gong et al., 2013a; Ochs et al., 2013). Despite the appealing result, it remains to be challenging as how to efficiently solve the optimization problem with non-convex regular-

*Ming Lin and Changshui Zhang are from State Key Laboratory on Intelligent Technology and Systems, Tsinghua National Laboratory for Informatin Science and Technology(TNList), Department of Automation, Tsinghua University, Beijing , P.R. China 100084.

izer.

Zhang (2012) proposed a multi-stage algorithm that relaxes a non–convex optimization problem into a sequence of convex optimization problems with weighted ℓ_1 regularizers. Besides the recovery error, the author also showed in (Zhang, 2012) that the solution found by multi-stage algorithm satisfies the oracle property when the non-zero entries in the target sparse vector are sufficiently large. The main shortcoming of the multi-stage algorithm is its potentially high computational cost as it needs to solve a sequence of ℓ_1 regularized optimization problems. (Zhaoran Wang, 2013) relax this problem by computing approximate solution at each stage, but still require multiple stages thus is not very efficient. Several proximal gradient descent methods have been proposed for non-convex regularizers (Gong et al., 2013b; Loh and Wainwright, 2013) that enjoy higher computational efficiency than the multi-stage algorithm. However, it is unclear if the solutions found by these algorithms will be unbiased estimators and have the oracle property, the key reason for using a non-convex regularizer.

In this work, we propose an algorithm for sparse recovery using adaptive non-convex regularizer to develop efficient algorithms with all the desired properties above. The proposed framework, on one hand, enjoys the high computational efficiency and achieves a linear convergence rate in the recovery error as some of the proximal gradient descent methods do. On the other hand, like the multi-stage algorithm, the proposed framewrok is able to find a sparse solution with optimal recovery error, selection consistency, and oracle property under appropriate conditions. The key feature introduced by the proposed framework is introducing *adaptive concave regularizers* whose shrinking-threshold vary over iterations. It is the introduction of the adaptive concave regularizer that allows us to effectively remove the noise and identify the support set, leading to high computational efficiency and a solution with optimal recovery error and oracle property.

Although the proposed algorithm is compatible with most popular non-convex regularizers, via a more deep examination, we find that the type of non-convex regularizer is not important at all. What really matters is the so–called *proximal projection gap* that will be defined later. This gap determines the bias of regularizer in sparse estimation. Based on this discovery, we propose to use a so–called ACCQ regularizer, whose proximal projection gap is zero. From optimization viewpoint, the ACCQ regularizer is equivalent to one kind of hard-thresholding algorithms with adaptive threshold. The ACCQ regularizer is the only regularizer whose projection gap is zero, thus is surperior than other alternatives.

The rest of this paper is organized as following. Section 2 reviews the related work. Section 3 describes the proposed algorithm for sparse recovery. Section 4 analyzes theoretical properties. Experimental results with both synthesized and real data sets are summarized in Section 5. Section 6 encloses our study with open questions.

2 RELATED WORK

We briefly review the related work on sparse recovery, with focus on non-convex regularizer. More complete references on the related subject can be found in (Tropp and Wright, 2010), (Davenport et al., 2011) and (Zhang and Zhang, 2011).

Most sparse recovery methods are based on ℓ_1 regularization. The most well algorithm is LASSO (Tibshirani, 1996; Efron et al., 2004). Numerous algorithms have been develop to solve LASSO related optimization problem efficiently (Beck and Teboulle, 2009; Foucart, 2012). It has been shown that ℓ_1 regularization can be solved efficiently with a linear convergence (up to stochastic tolerance) (Xiao and Zhang, 2012; Agarwal et al., 2012). A main problem with LASSO is that it is a biased estimator. In particular, LASSO is unable to perfectly recover the solution of oracle Least Square Estimation (LSE), a property that is usually referred to as *oracle property*. We emphasize that the LASSO bias is not an artifact of analysis, and it does show up noticeably in the recovery error, according to our empirical study as well as others (Zhang, 2012; Zou, 2006). It was pointed out in (Fan and Li, 2001; Zhang and Zhang, 2011) that LASSO bias also exists in other convex regularizers.

Multiple non-convex regularizers have been proposed to address the bias of convex regularizers, including Geman Penalty (GP) (Geman and Yang, 1995; Trzasko and Manduca, 2009), SCAD (Fan and Li, 2001), Log Sum Penalty (LSP) (Candes et al., 2008), ℓ_q norm (Foucart and Lai, 2009), Minimax Concave Penalty (MCP) (Zhang, 2010a), Capped–ℓ_1 norm (Loh and Wainwright, 2013). Various algorithms have been developed to find local optimal for non-convex regularizers((Zhang and Zhang, 2011) and references therein). It is however unclear if the local solutions found by these algorithms have the desired properties (i.e. the optimal recovery error and the oracle property). Only a handful algorithms that achieve the desired properties, including the multi-stage algorithm (Zhang, 2010b), adaptive LASSO (Zou, 2006) that can be shown as a special case of multi-stage algorithm and achieve the asymptotical oracle property, and the forward and backward regression scheme (FOBA) (Zhang, 2011), based on adaptive regularization, also finds the solutions with all the desired properties. FOBA is guaranteed to terminiate within $O(s)$ iterations, where s is the sparsity. The main limitation of FOBA is that it is unclear if the oracle property recovery error of their algorithm can achieve a linear convergence rate. Each iteration of FOBA is an optimization problem therefore is not efficient. Ji Liu (2013) propose an variant of FOBA but

they still suffer the same problem. The FOBA and its variants make different assumptions that is complementary to our analysis. Although the multi-stage algorithm achieves a linear convergence, it requires to solve a weighted ℓ_1 regularization problem that can be computationally costly when the dimension of data is high.

3 SPARSE RECOVERY BY ADAPTIVE NON-CONVEX REGULARIZER

In this section, we first introduce the background materials and notations for sparse recovery. We then present our algorithm and its main theoretical property. The sketch of proofs is given at the end of this section.

3.1 BACKGROUNDS AND NOTATIONS

Let A be an $n \times d$ design matrix and $\mathbf{y} \in \mathbb{R}^n$ be response vector satisfying

$$\mathbf{y} = A\mathbf{x}_* + \mathbf{z} \,, \tag{1}$$

where $\mathbf{x}_* \in \mathbb{R}^d$ is the s-sparse vector to be recovered and $\mathbf{z} \in \mathbb{R}^n$ is a noise vector. We assume that each element $[A^\top \mathbf{z}]_i$ follows a subgaussian distribution with $\|[A^\top \mathbf{z}]_i\|_{\psi_2} \leq \sigma\sqrt{n}, i = 1, \ldots, d$, where σ indicates the noise level in $\mathbf{z}$ and $\|\cdot\|_{\psi_2}$ is Orlicz norm (Koltchinskii, 2011). Using the property of subgaussianianity, we have, with a high probability $(1 - d^{-3})$,

$$\|[A\mathbf{z}]_i\|_\infty \leq 2\sigma\sqrt{n \log d} \tag{2}$$

For the rest of the paper, we will simply assume condition (2) holds.

Following (Candes and Tao, 2005), we assume A satisfies *Restricted Isometric Property* (R.I.P.) defined as follows.

Definition 1 (*Restricted Isometric Property*)**.** *A matrix A satifies δ_s–R.I.P. condition, if there exits a possitive constant δ_s such that for all s-sparse vector* $\mathbf{x}$*,*

$$(1 - \delta_s)\|\boldsymbol{x}\|_2^2 \leq \frac{1}{n}\|A\boldsymbol{x}\|_2^2 \leq (1 + \delta_s)\|\boldsymbol{x}\|_2^2 \,. \tag{3}$$

A is called $\delta_{s,s}$–restricted orthogonal, if there exists a possitive constant $\delta_{s,s}$ such that for any two s–sparse vector $\mathbf{u}, \mathbf{v}$ *whose support sets are disjoint,*

$$\frac{1}{n}|\langle A\mathbf{u}, A\mathbf{v}\rangle| \leq \delta_{s,s}\|\mathbf{u}\|_2\|\mathbf{v}\|_2 \,. \tag{4}$$

Small δ_s and $\delta_{s,s}$ indicate that A is approximately isometric on sparse subspace and any two set of s columns are approximately orthogonal if they are disjoint. In the rest of this paper, we will say A is δ–R.I.P. when both (3) and (4) are satisfied with $\delta = \max\{\delta_{2s}, \delta_{s,s}\}$.

Algorithm 1 Proximal Gradient Descent With Adaptive Capped Concave Quadratic (ACCQ) Regularizer

Input: the size of target vector $R \geq \|\mathbf{x}_*\|_2$, design matrix A, measurements $\mathbf{y}$, threshold θ, shrinking parameter $q \in (0, 1)$, and number of iterations T

1: **Initialization:** $\mathbf{x}_1 = 0$
2: **for** $t = 1$ to T **do**
3: Compute τ_t by $\tau_t = Rq^{t-1} + \theta$
4: Compute $\widehat{\mathbf{x}}_{t+1}$ by $\widehat{\mathbf{x}}_{t+1} = \mathbf{x}_t - \nabla L(\mathbf{x}_t)$
5: Update $\mathbf{x}_{t+1}$ using (8)
6: **end for**

Output: $\mathbf{x}_{T+1}$

There are other alternative conditions for spare recovery which are more general than R.I.P, such as restricted eigenvalue condition (Bickel et al., 2009). A complete list of conditions for sparse recovery and their comparison can be found (Van De Geer and Bühlmann, 2009). We choose R.I.P condition due to its simplicity, and extension to more general cases will be studied in the future. From now on we will assume that A is δ–R.I.P.. In experiments we generate A from random Gaussian distribution, which is widely known to obey the R.I.P. with a high probability.

To recover the sparse vector $\mathbf{x}_*$, a common approach is to minimize the regularized empirical loss

$$\min_{\mathbf{x}} \frac{1}{2n}\|\mathbf{y} - A\mathbf{x}\|_2^2 + \Omega(\mathbf{x}) \,, \tag{5}$$

where $\Omega(\mathbf{x})$ is a regularizer that controls the sparsity of the solution. To remove the LASSO bias, a non-convex regularizer is used for $\Omega(\mathbf{x})$, leading to a non-convex optimization problem that is not only difficult to solve numerically and but also challenging to analyze the theoretical properties for the found solution.

The following notation will be used throughout this paper. For a vector $\mathbf{x}$, we denote by $[\mathbf{x}]_i$ the i-the entry of $\mathbf{x}$, by $|\mathbf{x}|_i$ the absolute value of $[\mathbf{x}]_i$, by $[\mathbf{x}]_A$ the subvector of $\mathbf{x}$ that only includes the elements in the index set $A \subseteq [d]$, and by $\lambda_{\min}(\mathbf{x})$ the minimum absolute value of the non-zero entries in $\mathbf{x}$. We will use $\text{supp}(\mathbf{x})$ for the support set for a vector $\mathbf{x}$. We will use $\|\mathbf{x}\|_2$, $\|\mathbf{x}\|_1$, and $\|\mathbf{x}\|_\infty$ to represent the ℓ_2, ℓ_1, and ℓ_∞ norm of vector $\mathbf{x}$. We will denote by S_* the support set of $\mathbf{x}_*$ and by S_t the support set for $\mathbf{x}_t$.

3.2 PROXIMAL GRADIENT DESCENT USING ADAPTIVE CAPPED CONCAVE QUADRATIC (ACCQ) REGULARIZER

The proposed framework essentially follows the proximal gradient descent method that has been widely used in convex optimization. At each iteration, we first obtain an aux-

iliary solution $\widehat{\mathbf{x}}_t$ by

$$\begin{aligned} \widehat{\mathbf{x}}_t &= \mathbf{x}_t - \nabla L(\mathbf{x}_t) \\ \text{where} \quad L(\mathbf{x}) &= \frac{1}{2n}\|\mathbf{y} - A\mathbf{x}\|_2^2 \end{aligned} . \tag{6}$$

The updated solution $\mathbf{x}_{t+1}$ is then given by

$$\mathbf{x}_{t+1} = \arg\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{x} - \widehat{\mathbf{x}}_t\|_2^2 + \Omega_t(\mathbf{x}) \tag{7}$$

where the regularizer Ω_t has a subscript t and therefore varies from trial to trial.

To ensure Eq. (7) leading to an unbiased sparse estimation, we make some assumptions on the adaptive regularizer $\Omega_t(\mathbf{x})$. The type of $\Omega_t(\mathbf{x})$ is not important at all. We only care about its shrinking strategy in step Eq. (7).

Assumption 1. *Define variables τ_t and constant α. The adaptive non-convex regularizer $\Omega_t(\mathbf{x})$ in Eq. (7) shrinks $\hat{\mathbf{x}}_t$ in the following way:*

- *For $|\hat{\mathbf{x}}_t|_i < \tau_t$, $[\mathbf{x}_{t+1}]_i = 0$.*
- *For $|\hat{\mathbf{x}}_t|_i > \tau_t + \alpha$, $[\mathbf{x}_{t+1}]_i = [\hat{\mathbf{x}}_t]_i$.*
- *For $\tau_t \leq |\hat{\mathbf{x}}_t|_i \leq \tau_t + \alpha$, $0 < [\mathbf{x}_{t+1}]_i < [\hat{\mathbf{x}}_t]_i$.*

The τ_t is a threshold parameter that adaptively shrinks over iterations. We will describe the updating rule of τ_t later in Eq. (9). α is called *proximal projection gap*. In Assumption 1, if the intermedia solution $|\hat{\mathbf{x}}_t|_i$ is outside $[\tau_t, \tau_t + \alpha]$, the proximal projection of Ω_t is equivalent to hard-thresholding. Otherwise $|\hat{\mathbf{x}}_t|_i$ is projected onto $(0, |\hat{\mathbf{x}}_t|_i)$, whose value depends on the specific regularizer being used. The hard-thresholding is the key to unbias estimation, which is only possible when using non-convex regularizer. The proximal projection gap α reflects the non-convexity of $\Omega_t(\mathbf{x})$ in the proximal projection. For convex regularizer, α is infinity by definition because they are soft-thresholding methods that never do hard-thresholding. For a particular non-convex regularizer, we natually prefer small α to avoid involving bias as much as possible.

The shrinking strategy of non-convex regularizer is very similar to greedy algorithms. The following concavity assumption distinguish non-convex methods from greedy methods, which at the same time build a bridge of the two realms.

Assumption 2. *$\Omega_t(\mathbf{x})$ is concave in $\mathbf{x}$:*

$$\Omega_t(\mathbf{x}_1) - \Omega_t(\mathbf{x}_2) \leq \langle \partial\Omega_t(\mathbf{x}_2), \mathbf{x}_1 - \mathbf{x}_2 \rangle \ , \|\partial\Omega_t(\mathbf{x})\| \leq \tau_t \ ,$$

where $\partial\Omega_t(\mathbf{x})$ is the subgradient.

Most popular static non-convex regularizer and their adaptive variants fit Assumption 1 and Assumption 2, with different τ_t and α. We list a few of them in Table 1. We notice that in Table 1, most regularizers' α is not zero. Although our theory could deal with non-zero α, we natually hope there is a regularizer that doesn't need to suffer this gap α, which results in a better estimation. Inspired by this observation, we introduce *Adaptive Capped Concave Quadratic* regularizer (ACCQ) , defined in the last row of Table 1. Clearly, this regularizer is the only hard-thresholding regularizer whose proximal projection gap is zero. It is proximal projection is given by :

$$[\mathbf{x}_{t+1}]_i = \begin{cases} [\widehat{\mathbf{x}}_t]_i & |\widehat{\mathbf{x}}_t|_i \geq \tau_t \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

This specific adaptive hard-thresholding strategy allows us to correct the LASSO bias and consequentially achieve the oracle property when the signals in the target vectors are strong.

In the rest of this paper, we only focus on ACCQ and its recovery properties. For regularizers with $\alpha > 0$, all of the following theorems hold true except an extra α in the upper bound. Therefore they are always suboptimal compared with ACCQ.

Threshold parameter τ_t is determined by the following equation

$$\tau_t = Rq^{t-1} + \theta \ . \tag{9}$$

- Threshold parameter $\theta > 0$ determines the lower bound for τ_t. It is introduced to ensure that the regularization is strong enough to overcome the noise.
- Shrinking parameter $q \in (0, 1)$ controls the speed of shrinkage. The idea of using a shrinking regularizer is motivated by a simple observation: as we go through the iteration t, we expect the solution $\mathbf{x}_t$ will approach the target vector $\mathbf{x}_*$ with a smaller error. As a result, only a smaller regularization is needed to overcome the noise caused by the recovery error. We note that similar shrinking strategy has been used in sparse recovery with the ℓ_1 regularizer (Xiao and Zhang, 2012).

Algorithm 1 gives the details for the proposed algorithm.

Remark 1 In practice, the settings of q and θ is robust, as suggested by our theorems and experiments. We suggest to set $0.9 \leq q < 1$ and $\theta \in [O(\sigma/\sqrt{n}), \lambda_{\min}(\mathbf{x}_*)]$. For example, $q = 0.95$, $\theta = 0.005$ usually satisfies our assumption and works well in practice.

Remark 2 It is interesting to compare Algorithm 1 with greedy hard-thresholding algorithms like GraDeS. GraDeS keeps exactly s entries at each iteration, even if the smallest s-th entry contains large noise. The proposed algorithm gradually collects entries according to their magnitude and current estimation uncertainty. It doesn't keep entries that contain large noise, so at the beginning of each iteration, it will keep less than s entries. Another greedy algorithm is OMP, which greedy select entries then keep them as support set. OMP must ensure that it always collect right entry

Table 1: Adaptive Regularizers And Proximal Projection Gap

Name	$\Omega_t(\mathbf{x}_i)$	Gap α
adaptive ℓ_1 norm	$\tau_t\lambda\lvert\mathbf{x}_i\rvert$	∞
adaptive capped ℓ_1 norm	$\tau_t\lambda\min(\lvert\mathbf{x}_i\rvert,\theta) \quad \theta>0$	$\tau_t(\theta-\lambda)$
adaptive MCP	$\tau_t\lambda\int_0^{\lvert\mathbf{x}_i\rvert}\min(1,\frac{[\theta\lambda-\mathbf{x}_i]_+}{(\theta-1)\lambda})\,\mathrm{d}x \quad \theta\geq 2$	$\tau_t(\theta-1)\lambda$
ACCQ	$\Omega_t^{\mathrm{CCQ}}(\mathbf{x})=\sum_{i=1}^d \begin{cases} -\frac{1}{2}(\lvert\mathbf{x}\rvert_i-\tau_t)^2+\frac{1}{2}\tau_t^2 & \lvert\mathbf{x}\rvert_i<\tau_t \\ \frac{1}{2}\tau_t^2 & \text{otherwise}\end{cases}$	0

at each iteration, otherwise it will fail definitely. The propose algorithm adaptively throw out entries that is collected in the previous iterations. This strategy is clealy much more robst against noise.

3.3 MAIN THEORETICAL RESULTS

The following theorem shows that the recovery error for Algorithm 1 is reduced exponentially, and all the intermediate solutions are $2s$-sparse.

Theorem 1. *Assume $\mathbf{x}_*$ is s-sparse, and $6\delta<1$. Set parameter q, and θ in Algorithm 1 as*

$$q=\max\left(3\delta, 2\sqrt{\frac{\delta}{1-2\delta}}\right),\ \theta=2\sigma\sqrt{\frac{\log d}{n}} \tag{10}$$

Let $\mathbf{x}_1,\ldots,\mathbf{x}_T$ be the sequence of solutions output from algorithm 1. Let S_t be the support set for $\mathbf{x}_t$ and let S_* be the support set for $\mathbf{x}_*$. We have

$$|S_t\setminus S_*|\leq s,\ \|\mathbf{x}_t-\mathbf{x}_*\|_2\leq Rq^{t-1}+\frac{4\sigma}{1-q}\sqrt{\frac{s\log d}{n}} \tag{11}$$

First, as revealed by Theorem 1, the recovery error will converge geometrically to $O(\sigma\sqrt{s\log d/n})$, which has shown to be minimax optimal in (Raskutti et al., 2009). Second, as indicated in (11), all the intermediate solutions are at most $2s$-sparse, making it computational appealing as our algorithm only needs to maintain a vector of no more than $2s$ non-zero entries. This is similar to iterative hard thresholding algorithms (e.g. (Garg and Khandekar, 2009)). Finally, the linear convergence takes place when $\delta\leq 1/6$, which worse than some other conditions on δ (e.g. (Garg and Khandekar, 2009)). We believe that this condition is improvable by more carefully tuning the inequalities in our analysis via similar techniques demonstrated in (Garg and Khandekar, 2009).

Next, we will show that the solution found by Algorithm 1 is selection consistent and has the oracle property if $\lambda_{\min}(\mathbf{x}_*)$, the smallest absolute value for the non-zero entries in $\mathbf{x}_*$, is larger than $O(\sigma\sqrt{\log d/n})$. To this end, we first define the solution of Least Square Oracle (LSE) estimation.

Definition 2 (Least Square Oracle (LSE))**.** *The Least Square Oracle estimation is the least square estimation of $\mathbf{x}_*$ by assuming that the support set S_* of $\mathbf{x}_*$ is provided. The LSE solution $\mathbf{x}_o$ is given by*

$$\mathbf{x}_o=\mathbf{x}_{S_*}=(A_{S_*}^\top A_{S_*})^{-1}A_{S_*}^\top\mathbf{y}.$$

Definition 3 (Selection Consistent and Oracle Property)**.** *An estimator is selection consistent if the estimated solution $\widehat{\mathbf{x}}$ satisfies supp$(\widehat{\mathbf{x}})$ = supp$(\mathbf{x}_*)$, and has the oracle property if $\widehat{\mathbf{x}}=\mathbf{x}_o$.*

We note that since the solution of oracle LSE $\mathbf{x}_o$ is obtained without using any regularizer, the oracle property (i.e. $\widehat{\mathbf{x}}=\mathbf{x}_o$) essentially ensures that the sparse recovery algorithm will not be biased by the regularizer, of course under the assumption that $\lambda_{\min}(\mathbf{x}_*)$ is sufficiently large. We note that the early definition of oracle property (e.g. (Fan and Li, 2001)) requires $\widehat{\mathbf{x}}_{S_*}-\mathbf{x}_o$ converge to a Gaussian random vector when the number of measurements n goes to infinity, which is weaker than the finite sample version defined above.

Theorem 2. *Assume $6\delta<1$ and with q and θ set in (10). Define*

$$t_0=\log_2\left(\frac{\max(R,1)}{\sigma}\sqrt{\frac{\log d}{n}}\right)$$

Then, we have $S_t=S_*$ for $t>t_0$ (i.e. selection consistency) and

$$\|\mathbf{x}_t-\mathbf{x}_o\|_2\leq\left(\frac{3\delta}{1-3\delta}\right)^{(t-t_0)/2}\frac{8\sigma}{1-q}\sqrt{\frac{s\log d}{n}}$$

if

$$\lambda_{\min}(\mathbf{x}_*)\geq\frac{4\sigma}{1-\delta}\sqrt{2\frac{\log d}{n}} \tag{12}$$

and

$$s\leq\frac{1}{50\delta}(1-q)^2(2-\frac{1}{1-\delta}) \tag{13}$$

As revealed by the above theorem, $\mathbf{x}_o$, the solution of oracle LSE, can be perfectly recovered by Algorithm 1 with sufficiently large number of iterations, provided (i) the non-zero entries in $\mathbf{x}_*$ are sufficiently large, and (ii) the RIP constant δ is sufficiently small.

Remark 1 Eq. (13) is essentially a particular form of *Generalized Uncertainty Principle* (GUP) (Candès et al., 2006). GUP claims that for any compress sensing methods, the number s of non-zeros elements in $\mathbf{x}_*$ couldn't be larger than half of the number n of frequence sampling, i.e., $s \leq 0.5n$. Otherwise there is no algorithm could recover the sparse signal. In Eq. (13), generally speaking, $\delta = O(1/n)$. So Eq. (13) is claiming that s should be smaller than

$$s \leq cn \ ,$$

where c is a constant. The constant c given by Eq. (13) is slightly loose than $1/2$ which is proven to be optimal in GUP. We believe this could be refined by carefully tuning the inequalities in our analysis.

Remark 2 $\lambda_{\min}(\mathbf{x}_*)$ in Eq. (12) doesn't contain $\sqrt{s}$ thanks to the non-convexity of ACCQ. For convex regularizer, $\lambda_{\min}(\mathbf{x}_*) = O(\sqrt{s}\sigma)$ therefore is significantly suboptimal compared to Eq. (12).

Remark 3 If the proximal projection gap $\alpha > 0$, Eq. (12) should be modified as :

$$\lambda_{\min}(\mathbf{x}_*) \geq O(\sqrt{\frac{\log d}{n}} + \alpha) \ .$$

This is inferior than $\alpha = 0$. For $\alpha > 0$, the proof is similar. We will explore $\alpha > 0$ in the journal version of this paper.

3.4 PROOF SKETCH

In this analysis, we provide a sketch of proof for Theorem 1. The proof for Theorem 2 mostly follows the analysis of Theorem 1 by carefully exploiting the property of non-convex regularizer. All the detailed proofs can be found in the supplementary document.

The first step toward the proof for Theorem 1 is to show that the solution $\mathbf{x}_{t+1}$ will be sparse if $\mathbf{x}_t$ is sparse, which is revealed by the following theorem.

Theorem 3. *Assume $|S_*| \leq s$, and $|S_t \setminus S_*| \leq s$. Then, if we set*

$$\tau_t \geq \frac{3\delta}{\sqrt{s}}\|\mathbf{x}_t - \mathbf{x}_*\|_2 + 2\sigma\sqrt{\frac{\log d}{n}},$$

we have $|S_{t+1} \setminus S_*| \leq s$.

In the second step, we show in the theorem below that the recovery error will be reduced exponentially, up to the stochastic tolerance (i.e. $O(\sigma\sqrt{s\log d/n})$).

Theorem 4. *Assume $|S_*| \leq s$, $|S_t \setminus S_*| \leq s$, and $\|\mathbf{x}_t - \mathbf{x}_*\|_2 \leq \Delta_t$. Then, by setting*

$$\tau_t = \frac{3\delta}{\sqrt{s}}\Delta_t + 2\sigma\sqrt{\frac{\log d}{n}},$$

we have

$$\|\mathbf{x}_{t+1} - \mathbf{x}_*\|_2 \leq q\Delta_t + 4\sigma\sqrt{\frac{s\log d}{n}}$$

where

$$q = \max\left(3\delta, 2\sqrt{\frac{\delta}{1-2\delta}}\right)$$

With the above two theorems, we will show Theorem 1 by induction. Since $\|\mathbf{x}_1 - \mathbf{x}_*\|_2 = \|\mathbf{x}_*\|_2 \leq R$, Theorem 1 holds for $t = 1$. Let's assume that it holds for $\mathbf{x}_t$. Using Theorem 3, we have that $\mathbf{x}_{t+1}$ is a $2s$ sparse vector with $|S_{t+1} \setminus S_*| \leq s$. Using Theorem 4, we have

$$\begin{aligned}\|\mathbf{x}_{t+1} - \mathbf{x}_*\|_2 &\leq q\Delta_t + 4\sigma\sqrt{\frac{s\log d}{n}} \\ &\leq q^t R + \frac{4\sigma}{1-q}\sqrt{\frac{s\log d}{n}}\end{aligned}$$

4 EXPERIMENTS

Since there are thounsands of papers discussing compress sensing, it it impossible to compare every method published in this paper. We mainly select methods with theoretical gaurantees and provable geometrical convergence rate. We compare Algorithm 1 (ACCQ) with five baseline methods:

- the greedy hard thresholding method (GraDeS) (Garg and Khandekar, 2009),
- Multiple Stage Capped ℓ_1–norm method (MSCL1) (Zhang, 2012),
- non–convex proximal gradient descent with MCP (P–MCP) (Loh and Wainwright, 2013),
- the GIST method (Gong et al., 2013b), and
- Homotopy LASSO (LASSO) (Xiao and Zhang, 2012).

GraDeS is a greedy hard thresholding method with a geometrical convergence rate. We set $\gamma = 3$ in GraDeS as recommended in (Garg and Khandekar, 2009). Homotopy LASSO is a LASSO solver with geometrical convergence rate. We tune the regularizer parameter λ in Homotopy LASSO in set $\{1, 0.1, 10^{-2}, 10^{-3}, 10^{-4}\}$, and report the best performance. MSCL1, GIST and P–MCP are based on non–convex regularizers. The threshold parameter θ in MSCL1 is set to be $\theta = \lambda_{\min}(\mathbf{x}_*)/2 = 0.05$. Any $\theta < \lambda_{\min}(\mathbf{x}_*)$ should work as well (Zhang, 2012). Here we choose $\theta = \lambda_{\min}(\mathbf{x}_*)/2$ because it is at the same time large enough to suppress the noise. For the proposed algorithm, we similarly set $\theta = \lambda_{\min}(\mathbf{x}_*)/2 = 0.05$. We tune parameter q in the set $\{0.8, 0.9, 0.95, 0.99, 0.995\}$.

Two metrics are used to evaluate the recovery performance of different algorithms. To evaluate the recovery error, we follow compress sensing settings (Candes and Tao, 2005;

Table 2: Dataset Statistics

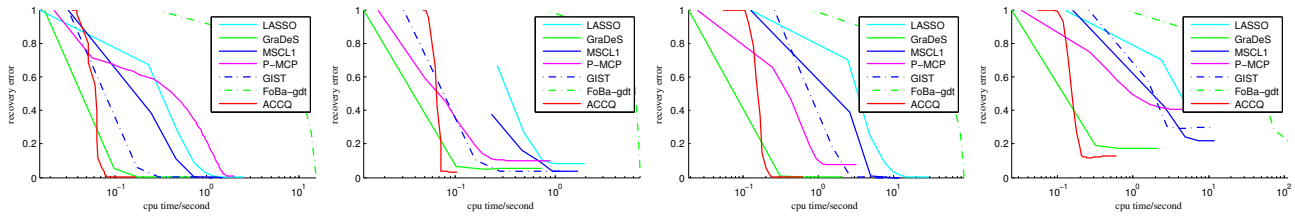
	d	$\|\mathbf{x}_*\|_0$	$\|\mathbf{x}_*\|_\infty$	$\|\mathbf{x}_*\|_2$	$\lambda_{\min}(\mathbf{x}_*)$
airplanes	$6.3 \times 10^4 \pm 6 \times 10^3$	$4.2 \times 10^2 \pm 92$	0.36 ± 0.059	0.96 ± 0.011	0.01
butterfly	$7 \times 10^4 \pm 1 \times 10^4$	$6.3 \times 10^2 \pm 2.6 \times 10^2$	0.28 ± 0.084	0.93 ± 0.045	0.01
camera	$7.4 \times 10^4 \pm 1.5 \times 10^4$	$5 \times 10^2 \pm 1.6 \times 10^2$	0.33 ± 0.11	0.94 ± 0.019	0.01
dolphin	$6.4 \times 10^4 \pm 1.1 \times 10^4$	$6 \times 10^2 \pm 2.1 \times 10^2$	0.27 ± 0.061	0.94 ± 0.027	0.01

Statistics of dataset of each category. d the dimension of $\mathbf{x}_*$. $\|\mathbf{x}_*\|_0$ is the number of non-zero entries in $\mathbf{x}_*$. $\|\mathbf{x}_*\|_\infty$ is the maximal amplitude of entries in $\mathbf{x}_*$. $\|\mathbf{x}_*\|_2$ is the ℓ_2–norm of $\mathbf{x}_*$. $\lambda_{\min}(\mathbf{x}_*)$ is the smallest absolute value of the non-zero entries in $\mathbf{x}_*$. All numbers in the table are average values plus variances.

Table 3: Support Set Recovery Errors ϵ_{supp}

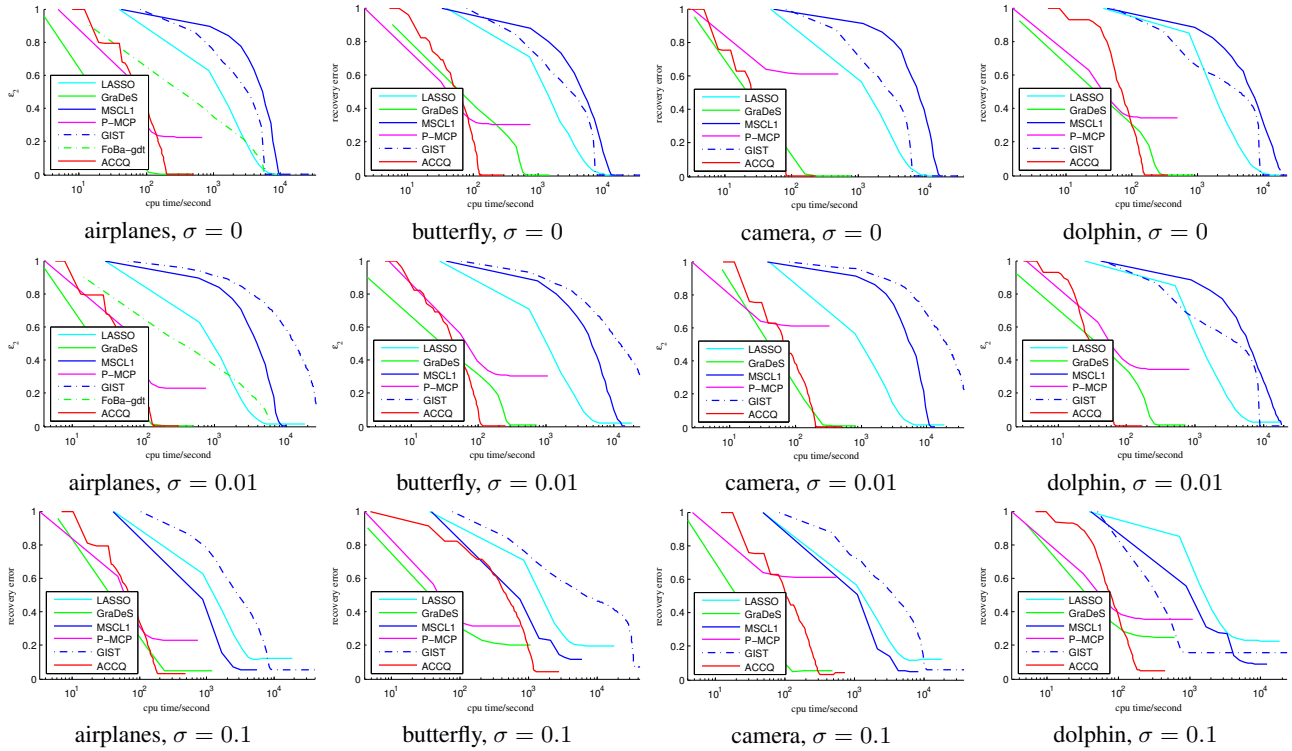
	LASSO	GraDeS	MSCL1	P-MCP	GIST	ACCQ
airplanes						
$\sigma = 0$	2.1×10^{-3}	0.0×10^{0}	0.0×10^{0}	2.0×10^{-1}	0.0×10^{0}	0.0×10^{0}
$\sigma = 0.01$	5.9×10^{-2}	8.2×10^{-3}	2.1×10^{-4}	2.0×10^{-1}	1.7×10^{-1}	0.0×10^{0}
$\sigma = 0.1$	6.0×10^{-2}	1.5×10^{-3}	5.3×10^{-4}	4.0×10^{-1}	4.4×10^{-2}	1.6×10^{-4}
butterfly						
$\sigma = 0$	3.8×10^{-2}	7.9×10^{-3}	4.4×10^{-3}	4.1×10^{-1}	5.9×10^{-3}	3.1×10^{-3}
$\sigma = 0.01$	3.9×10^{-2}	8.5×10^{-3}	4.7×10^{-3}	4.1×10^{-1}	1.0×10^{-1}	2.5×10^{-3}
$\sigma = 0.1$	6.3×10^{-2}	9.3×10^{-3}	1.1×10^{-2}	4.1×10^{-1}	8.4×10^{-3}	6.7×10^{-3}
camera						
$\sigma = 0$	5.2×10^{-3}	0.0×10^{0}	0.0×10^{0}	6.8×10^{-3}	0.0×10^{0}	0.0×10^{0}
$\sigma = 0.01$	5.4×10^{-2}	6.1×10^{-3}	5.0×10^{-3}	6.8×10^{-3}	9.7×10^{-2}	0.0×10^{0}
$\sigma = 0.1$	5.4×10^{-2}	2.6×10^{-3}	7.9×10^{-4}	6.8×10^{-3}	4.0×10^{-2}	1.5×10^{-4}
dolphin						
$\sigma = 0$	1.3×10^{-2}	3.8×10^{-3}	1.9×10^{-2}	2.1×10^{-1}	1.4×10^{-2}	0.0×10^{0}
$\sigma = 0.01$	2.3×10^{-2}	9.7×10^{-3}	1.9×10^{-2}	2.1×10^{-1}	1.4×10^{-2}	0.0×10^{0}
$\sigma = 0.1$	6.8×10^{-2}	5.9×10^{-3}	1.1×10^{-3}	2.1×10^{-1}	2.0×10^{-3}	1.5×10^{-4}

Support set recovery errors under different noise level. The smaller, the better.



toy1 $\sigma = 0$ toy1 $\sigma = 0.1$ toy2 $\sigma = 0$ toy2 $\sigma = 0.5$

Figure 1: ℓ_2 Norm Recovery Errors For Synthetic Data Sets. x–Axis is CPU Time (Seconds) in Logrithmic Scale and y–Axis is ℓ_2 Norm Recovery Error $\|\mathbf{x}_t - \mathbf{x}_*\|$



airplanes, $\sigma = 0$ butterfly, $\sigma = 0$ camera, $\sigma = 0$ dolphin, $\sigma = 0$

airplanes, $\sigma = 0.01$ butterfly, $\sigma = 0.01$ camera, $\sigma = 0.01$ dolphin, $\sigma = 0.01$

airplanes, $\sigma = 0.1$ butterfly, $\sigma = 0.1$ camera, $\sigma = 0.1$ dolphin, $\sigma = 0.1$

Figure 2: The Recovery Error For Real Images From The Caltech 101 Data Set

Zou and Hastie, 2005; Loh and Wainwright, 2012) and measure the ℓ_2 norm of the difference between the target vector and the recovered one. In order to take into the computational efficiency, we plot the recovery error vs. the running time for each algorithm. To evaluate the property of selection consistency, we also measure the accuracy in recovering the support set of the original sparse vector that defined as follows:

$$\epsilon_{\text{supp}} = (|S_t \backslash S_*| + |S_* \backslash S_t|)/d \,.$$

Therefore $\epsilon_{\text{supp}} = 0$ if and only if $S_t = S_*$.

All codes are implemented in Matlab, running on Intel(R) Core(TM)2 Duo CPU, P8700@2.53 GHz, Windows 7 64bit, 4GB memory. We terminate each algorithm if it runs more than five hours.

4.1 EXPERIMENTS WITH SYNTHETIC DATA

First we verify the effectiveness of the proposed algorithm on synthetic data. We generate A and $\mathbf{z}_*$ independently from standard normal distributions. To generate the sparse vector $\mathbf{x}_*$, we first draw a random Gaussian vector $\mathbf{x}'_*$. We then normalize $\mathbf{x}'_*$ to be one and only keep the largest s entries in $\mathbf{x}'_*$. We create two synthetic data sets: toy1 ($d = 1000$, $s = 50$, $\sigma = \{0, 0.1\}$, $n = 500$), and toy2 ($d = 5000$, $s = 50$, $\sigma = \{0, 0.5\}$, $n = 1000$). Clearly toy2 is more difficult than toy1 because of the high dimensionality and large noise. The performance averaged over 10 trials is reported in this study. Since GraDeS needs to set all the entries in the intermediate solution to be zero except for the first k largest entries at each iteration, we tune k in set $\{40, 60, 80, 100\}$, and report the best performance.

Figure 1 shows the recovery results for the synthetic data sets, where the horizontal axis is CPU time (second) in the logarithmic scale. We observe that the proposed method ACCQ, although has the slow start at the beginning, is able to find a solution with small recovery error significantly faster than the other baseline methods. The slow start of the proposed algorithm is mostly due to the fact that the initial threshold τ_1 is set too high, leading to $\mathbf{x}_t = \mathbf{0}$ for the first a few of iterations. The LASSO bias is revealed by the noisy cases, where LASSO has the worst recovery error compared to the other methods. We also observe that the GraDeS method, an iterative hard thresholding algorithm, works well for the two toy datasets, in terms of both computational time and recovery error. We however found that for the real images, the GraDeS method behaves unstably when measurements are contaminated with random noise, as shown in the experimental result in the next subsection.

4.2 EXPERIMENTS WITH REAL IMAGES

Dataset We select a subset of images in Caltech101 as the sparse signals. Five images randomly chosen from four categories categories— "airplanes", "butterfly", "camera" and "dolphin"— are used in this study. For the convenience

of presentation, we denote each of the five images selected from one category by "001.jpg" to "005.jpg". The total number of images we extract is 20. To generate a truely s–sparse signal, all images are first normalized to be zero–mean and unit variance. We then apply Fourier transform to the normalized image and filter out Fourier components with coefficient smaller than 0.01. The final sparse vector $\mathbf{x}_*$ is constructed based on the survived Fourier components. Table 2 summarizes the statistics of our dataset.

For each s–sparse signal $\mathbf{x}_*$, we independently generate random Gaussian design matrix A with $n = 5000$. The entries in noise vector $\mathbf{z}$ are independently drawn from a Gaussian distribution with variance σ varied in set $\{0, 0.01, 0.1\}$. When $\sigma > 0.1$, no algorithm could do a good job due to heavy information corruption. For parameter k in GraDeS, i.e. the number of non-zero entries to be kept at each iteration, we tune it in set $\{500, 1000, 2000\}$ and report the best performance.

Figure 2 presents the convergence rate of ℓ_2-norm recovery error of the six methods. To save space we only show the result for "001.jpg" in each category here, and the results for the remaining images can be found in the supplementary document. Similar to the result of the synthetic data sets, The bias of LASSO is again revealed by its large recovery error for the noisy cases compared to several algorithms in comparison. The proposed algorithm ACCQ, although with a slow start, is able to find the solution with a small error significantly faster than the baseline algorithms on almost all cases except for airplanes with $\sigma = 0$ and camera with $\sigma = 0.1$, where the GraDeS method is the most efficient but with a slightly worse error. We notice that the performance of the GraDeS method appears to be not very consistent across images: it performs well for airplane and camera images, but does poorly for the images of butterfly and dolphin. Similarly, P-MCP and GIST, two sparse recovery algorithms with non-convex regularizers, although works well for the synthetic datasets, performs poorly for a number of cases. More investigation is needed to further understand the behavior of these algorithms.

In Table 2, we report the support set recovery error for each dataset under different noise level. Similar to ℓ_2 norm recovery error, ACCQ achieves perfect support set recovery on almost all datasets under $\sigma = \{0, 0.01\}$ except for *butterfly*. When noise $\sigma = 0.1$ is large, although ACCQ is unable to recover the exact support set, its error in recovering the support set is the smallest among the methods in comparison.

5 CONCLUSION

We propose an adaptive non–convex method to efficiently recover the sparse signal under compress sensing settings. The proposed method achieves a geometrical convergence rate for ℓ_2–norm recovery error up to the statistical tolerance. By using a non-convex regularizer, the proposed method is able to remove the LASSO bias and achieve the selection consistency and oracle property. Experiments with both synthetic data sets and real images verify both the efficiency and effectiveness of the proposed method compared to the state-of-the-art methods for sparse recovery. In the future, we would like to improve our analysis to remove the extra condition in (13) for selection consistency and oracle property. We also plan to extend the proposed algorithm to other sparse recovery problems such as group sparsity and low rank matrix recovery.

Acknowledgments

The authors would like to thank the anonymous reviewers for their insightful suggestions. This work is supported by 973 Program(2013CB329503), NSFC (Grant No. 91120301) and Tsinghua National Laboratory for Information Science and TechnologyTNListCross-discipline Foundation.

References

Alekh Agarwal, Sahand Negahban, and Martin J Wainwright. Fast global convergence of gradient methods for high-dimensional statistical recovery. *Ann. Stat.*, 40(5):2452–2482, 2012.

Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm with application to wavelet-based image deblurring. In *ICASSP*, pages 693–696, 2009.

Peter J Bickel, Ya'acov Ritov, and Alexandre B Tsybakov. Simultaneous analysis of lasso and dantzig selector. *Ann. Stat.*, 37(4):1705–1732, 2009.

Emmanuel Candes and Terence Tao. The dantzig selector: statistical estimation when p is much larger than n. *Ann. Stat.*, pages 2313–2351, 2007.

Emmanuel J Candes and Terence Tao. Decoding by linear programming. *IEEE Transactions on Information Theory*, 51(12): 4203–4215, 2005.

Emmanuel J Candès, Justin Romberg, and Terence Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on Information Theory*, 52(2):489–509, 2006.

Emmanuel J Candes, Michael B Wakin, and Stephen P Boyd. Enhancing sparsity by reweighted ℓ_1 minimization. *Journal of Fourier Analysis and Applications*, 14(5-6):877–905, 2008.

Mark A Davenport, Marco F Duarte, Yonina C Eldar, and Gitta Kutyniok. Introduction to compressed sensing. *Preprint*, 93, 2011.

Bradley Efron, Trevor Hastie, Iain Johnstone, and Robert Tibshirani. Least angle regression. *Ann. Stat.*, 32(2):407–499, 2004.

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

Simon Foucart. Sparse recovery algorithms: sufficient conditions in terms of restricted isometry constants. In *Approximation Theory XIII: San Antonio 2010*, pages 65–77. Springer, 2012.

Simon Foucart and Ming-Jun Lai. Sparsest solutions of underdetermined linear systems via ℓ_q-minimization for $0 < q < 1$. *Applied and Computational Harmonic Analysis*, 26(3):395–407, 2009.

Rahul Garg and Rohit Khandekar. Gradient descent with sparsification: an iterative algorithm for sparse recovery with restricted isometry property. In *ICML*, pages 337–344, 2009.

Donald Geman and Chengda Yang. Nonlinear image recovery with half-quadratic regularization. *ITIP*, 4(7):932–946, 1995.

Pinghua Gong, Jieping Ye, and Changshui Zhang. Multi-stage multi-task feature learning. *JMLR*, 14:2979–3010, 2013a.

Pinghua Gong, Changshui Zhang, Zhaosong Lu, Jianhua Z. Huang, and Jieping Ye. A general iterative shrinkage and thresholding algorithm for non-convex regularized optimization problems. In *ICML*, 2013b.

Jieping Ye Ji Liu, Ryohei Fujimaki. Forward-backward greedy algorithms for general convex smooth functions over a cardinality constraint. *arXiv:1401.0086*, 2013.

V. Koltchinskii. *Oracle Inequalities in Empirical Risk Minimization and Sparse Recovery Problems*, volume 2033. Springer, 2011.

Po-Ling Loh and Martin J Wainwright. High-dimensional regression with noisy and missing data: Provable guarantees with nonconvexity. *Ann. Stat.*, 40(3):1637–1664, 2012.

Po-Ling Loh and Martin J Wainwright. Regularized m-estimators with nonconvexity: Statistical and algorithmic theory for local optima. *arXiv preprint arXiv:1305.2436*, 2013.

Peter Ochs, Alexey Dosovitskiy, Thomas Brox, and Thomas Pock. An iterated ℓ_1 algorithm for non-smooth non-convex optimization in computer vision. In *CVPR*, 2013.

Garvesh Raskutti, Martin J Wainwright, and Bin Yu. Minimax rates of convergence for high-dimensional regression under q-ball sparsity. In *Annual Allerton Conference on Communication, Control, and Computing*, pages 251–257, 2009.

Mathieu Rosenbaum and Alexandre B Tsybakov. Sparse recovery under matrix uncertainty. *Ann. Stat.*, 38(5):2620–2651, 2010.

R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

Joel A Tropp and Stephen J Wright. Computational methods for sparse solution of linear inverse problems. *Proceedings of the IEEE*, 98(6):948–958, 2010.

Joshua Trzasko and Armando Manduca. Relaxed conditions for sparse signal recovery with general concave priors. *IEEE Transactions on Signal Processing*, 57(11):4347–4354, 2009.

Sara A Van De Geer and Peter Bühlmann. On the conditions used to prove oracle results for the lasso. *Electronic Journal of Statistics*, 3:1360–1392, 2009.

Shuo Xiang, Xiaotong Shen, and Jieping Ye. Efficient sparse group feature selection via nonconvex optimization. In *ICML*, 2013.

Lin Xiao and Tong Zhang. A proximal-gradient homotopy method for the sparse least-squares problem. *arXiv preprint arXiv:1203.3002*, 2012.

Huan Xu, Constantine Caramanis, and Shie Mannor. Robust regression and lasso. *IEEE Transactions on Information Theory*, 56(7):3561–3574, 2010.

C.H. Zhang and T. Zhang. A general theory of concave regularization for high dimensional sparse estimation problems. *arXiv preprint arXiv:1108.4988*, 2011.

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

Tong Zhang. Analysis of multi-stage convex relaxation for sparse regularization. *JMLR*, 11:1081–1107, 2010b.

Tong Zhang. Adaptive forward-backward greedy algorithm for learning sparse representation. *IEEE Transactions on Information Theory*, 57:4689–4708, 2011.

Tong Zhang. Multi-stage convex relaxation for feature selection. *Bernoulli*, 2012.

Tong Zhang Zhaoran Wang, Han Liu. Optimal computational and statistical rates of convergence for sparse nonconvex learning problems. *arXiv:1306.4960*, 2013.

Hui Zou. The adaptive lasso and its oracle properties. *Journal of the American Statistical Association*, 101(476):1418–1429, 2006.

Hui Zou and Trevor Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society Series B Statistical Methodology*, 67(2):301–320, 2005.