Efficient and Precise Sparse Learning: A Unified Framework for Controlling Algorithmic Complexity and Statistical Convergence

Chris Junchi Li[◇]

Department of Electrical Engineering and Computer Sciences[◇]
University of California, Berkeley

October 7, 2024

Abstract

In modern high-dimensional statistical learning, effective estimation often hinges on controlling both computational complexity and statistical accuracy, especially when the number of variables far exceeds the sample size. This paper introduces a novel iterative local adaptive majorize-minimization (TAC) algorithm designed to tackle sparse learning problems where the loss functions are nonconvex, such as those encountered in penalized M-estimators. TAC bridges the gap between theory and practice by ensuring both polynomial-time computational efficiency and optimal statistical convergence, even when working with nonconvex penalties like SCAD and MCP. The algorithm is divided into two key stages: a contraction phase that produces a crude but computationally efficient initial estimate, and a tightening phase that refines this estimate to achieve the optimal statistical rate of convergence. Numerical simulations demonstrate the method's ability to efficiently handle high-dimensional data, while rigorous theoretical analysis guarantees its statistical properties under weak assumptions. The proposed framework is applicable to a wide family of loss functions, offering new insights into folded-concave penalization and adaptive Lasso.

Keywords: Sparse learning, nonconvex optimization, majorize-minimization, statistical convergence, high-dimensional data, folded-concave penalties, adaptive Lasso.

1 Introduction

High-dimensional statistical learning has become a cornerstone of modern data science, where the number of features can vastly exceed the number of observations. Modern data acquisitions routinely measure massive amounts of variables, which can be much larger than the sample size, making statistical inference an ill-posed problem. This poses fundamental challenges for both computational efficiency and statistical accuracy. When dealing with such large datasets, the primary objective is to develop models that are not only interpretable but also capable of generalizing well to unseen data. Sparse learning, where only a small subset of features contribute significantly to the model, has emerged as a promising solution to these challenges.

A widely adopted approach for obtaining sparse models is through the use of penalized M-estimators. These estimators aim to minimize a loss function while incorporating a sparsity-inducing penalty, such as Lasso. However, convex penalties like Lasso are known to introduce estimation bias, prompting the development of nonconvex penalties, such as the Smoothly Clipped Absolute Deviation (SCAD) and Minimax Concave Penalty (MCP). These penalties eliminate the bias and achieve superior statistical rates of convergence. Despite their theoretical advantages, solving the corresponding optimization problems remains computationally challenging due to the nonconvex nature of the penalties.

This paper addresses these challenges by introducing a new algorithmic framework: the *Tightening After Contraction* (TAC) algorithm. TAC is designed to handle high-dimensional, nonconvex optimization problems by iteratively refining an initial estimator in a computationally efficient manner. The proposed method leverages a two-stage process: a *contraction* stage that delivers a coarse but computationally efficient initial estimate, and a *tightening* stage that iteratively refines the estimate to achieve optimal statistical properties. Unlike traditional methods that decouple statistical properties from algorithmic performance, TAC ensures both polynomial-time computational efficiency and strong statistical guarantees for a wide range of loss functions, including square loss and logistic loss.

Background and Formulation For inferential tractability and interpretability, one common approach is to exploit the penalized M-estimator

$$\widehat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}\in\mathbb{R}^d}{\operatorname{argmin}} \left\{ \mathcal{L}(\boldsymbol{\beta}) + \mathcal{R}_\lambda(\boldsymbol{\beta}) \right\}, \tag{1.1}$$

where $\mathcal{L}(\cdot)$ is a smooth loss function, $\mathcal{R}_\lambda(\cdot)$ is a sparsity-inducing penalty with a regularization parameter λ. Our framework encompasses the square loss, logistic loss, Gaussian graphical model negative log-likelihood loss, Huber loss and the family of folded concave penalties [Fan and Li (2001)]. Finding optimal statistical procedures with controlled computational complexity characterizes the efforts of high-dimensional statistical learning in the last two decades. This paper makes an important leap toward this grand challenge by proposing a general algorithmic strategy for solving (1.1) even when $\mathcal{R}_\lambda(\boldsymbol{\beta})$ is nonconvex.

A popular choice of $\mathcal{R}_\lambda(\boldsymbol{\beta})$ is the Lasso penalty [Tibshirani (1996)], a convex penalty. Though a large literature exists on understanding the theory of penalized M-estimators with convex penalties [Bickel, Ritov and Tsybakov (2009), Bunea, Tsybakov and Wegkamp (2007), van de Geer and Bühlmann (2009), Negahban et al. (2012)], it has been well known [Fan and Li (2001), Zou (2006)] that the convex penalties introduce nonnegligible estimation biases. In addition, the algorithmic issues for finding a global minimizer are rarely addressed. To eliminate the estimation bias, a family of folded-concave penalties was introduced by Fan and Li (2001), which includes the smooth clipped absolute deviation (SCAD) [Fan and Li (2001)], minimax concave penalty (MCP) [Zhang (2010a)], and capped ℓ_1-penalty [Zhang (2010b)]. Compared to their convex counterparts, these nonconvex penalties eliminate the estimation bias and attain more refined statistical rates of convergence. However, it is more challenging to analyze the theoretical properties of the resulting estimators due to nonconvexity of the penalty functions. Existing work on nonconvex penalized M-estimators treats the statistical properties and practical algorithms separately. On one hand, statistical properties are established for the hypothetical global optimum (or some local minimum), which is usually unobtainable by any practical algorithm in polynomial time. For example, Fan and Li (2001) showed that there exists a local solution that possesses an oracle property; Kim, Choi and Oh (2008) and Fan and Lv (2011) showed that the oracle estimator is a local minimizer with high probability. Later on, Kim and Kwon (2012) and Zhang and Zhang (2012) proved that the global optimum achieves the oracle property under certain conditions. Nevertheless, none of these papers specify an algorithm to find the desired solution. More recently, Agarwal, Negahban and Wainwright (2012), Loh and Wainwright (2015), Negahban et al. (2012) develop a projected gradient algorithm with desired statistical guarantees. However, they need to modify the estimating procedures to include an additional ℓ_1-ball constraint, $\|\boldsymbol{\beta}\|_1 \le R$, which depends on the unknown true parameter. On the other hand, practitioners have developed numerous heuristic algorithms for nonconvex

optimization problems, but without theoretical guarantees. One such example is the coordinate optimization strategy studied in Breheny and Huang (2011) and Friedman et al. (2007).

So there is a gap between theory and practice: What is actually computed is not the same as what has been proved. To bridge this gap, we propose an iterative local adaptive majorize-minimization (TAC) algorithm for fitting high-dimensional statistical models. Unlike most existing methods, which are mainly motivated from a statistical perspective and ignore the computational consideration, TAC is both algorithmic and statistical: it computes an estimator within polynomial time and achieves optimal statistical accuracy for this estimator. In particular, TAC obtains estimators with the strongest statistical guarantees for a wide family of loss functions under the weakest possible assumptions. Moreover, the statistical properties are established for the estimators computed exactly by our algorithm, which is designed to control the cost of computing resources. Compared to existing works [Agarwal, Negahban and Wainwright (2012), Loh and Wainwright (2015), Negahban et al. (2012)], our method does not impose any constraint that depends on the unknown true parameter.

Inspired by the local linear approximation to the folded concave penalty [Zou and Li (2008)], we use TAC to solve a sequence of convex programs up to a prefixed optimization precision

$$\min_{\boldsymbol{\beta}\in\mathbb{R}^d}\left\{\mathcal{L}(\boldsymbol{\beta})+\mathcal{R}(\boldsymbol{\lambda}^{(\ell-1)}\odot\boldsymbol{\beta})\right\}\qquad\text{for }\ell=1,\ldots,T \tag{1.2}$$

where $\lambda^{(\ell-1)}=\left(\lambda \mathrm{w}\left(|\widetilde{\beta}_1^{(\ell-1)}|\right),\ldots,\lambda \mathrm{w}\left(|\widetilde{\beta}_d^{(\ell-1)}|\right)\right)^{\mathrm{T}},\widetilde{\boldsymbol{\beta}}^{(\ell)}$ is an approximate solution to the ℓ th optimization problem in (1.2), $\mathrm{w}(\cdot)$ is a weighting function, $\mathcal{R}(\cdot)$ is a decomposable convex penalty function and " $\odot$ " denotes the Hadamard product. In this paper, we mainly consider $\mathcal{R}(\boldsymbol{\beta})=\|\boldsymbol{\beta}\|_1$, though our theory is general. The weighting function corresponds to the derivative of the folded concave penalty in Fan and Li (2001), Zou and Li (2008) and Fan and Lv (2011).

In particular, the TAC algorithm obtains a crude initial estimator $\widetilde{\boldsymbol{\beta}}^{(1)}$ and further solves the optimization problem (1.2) for $\ell\geq 2$ with established algorithmic and statistical properties. This provides theoretical insights on how fast the algorithm converges and how much computation is needed, as well as the desired statistical properties of the obtained estimator. The whole procedure consists of T convex programs, each only needs to be solved approximately to control the computational cost. Under mild conditions, we show that only $\log(\lambda\sqrt{n})$ steps are needed to obtain the optimal statistical rate of convergence. Even though TAC solves approximately a sequence of convex programs, the solution enjoys the same optimal statistical property of the unobtainable global optimum for the folded-concave penalized regression. The adaptive stopping rule for solving each convex program in (1.2) allows us to control both computational costs and statistical errors. Figure 1 provides a geometric illustration of the TAC procedure. It contains a contraction stage and a tightening stage as described below.

- Contraction Stage: In this stage ($\ell=1$), we approximately solve a convex optimization problem (1.2), starting from any initial value $\widetilde{\boldsymbol{\beta}}^{(0)}$, and terminate the algorithm as long as the approximate solution enters a desired contraction region, which will be characterized in Section 2.3. The obtained estimator is called the contraction estimator, which is very crude and only serves as initialization.

- Tightening Stage: This stage involves multiple tightening steps ($\ell\geq 2$). Specifically, we iteratively tighten the contraction estimator by solving a sequence of convex programs. Each step contracts its initial estimator toward the true parameter until it reaches the optimal region of

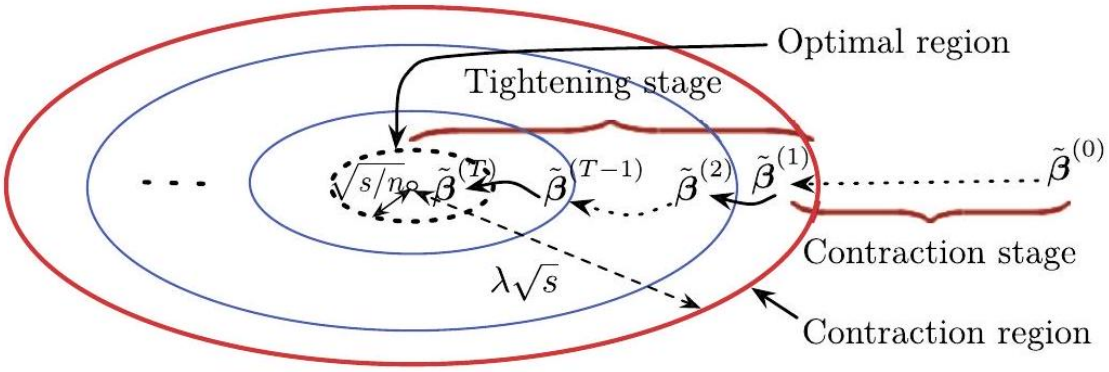

Figure 1. Geometric illustration of the contraction property. The contraction stage produces an initial estimator, starting from any initial value $\widetilde{\boldsymbol{\beta}}^{(0)}$ that falls in the contraction region, which secures the tightening stage to enjoy optimal statistical and computational rates of convergence. The tightening stage adaptively refines the contraction estimator until it enters the optimal region, which is stated in (1.3). Here, λ is a regularization parameter, s the number of nonzero coefficients in $\boldsymbol{\beta}^*$ and n the sample size.

convergence. At that region, further iteration does not improve statistical performance. See Figure 1. More precisely, we will show the following contraction property:

$$\|\widetilde{\boldsymbol{\beta}}^{(\ell)} - \boldsymbol{\beta}^*\|_2 \lesssim \sqrt{\frac{s}{n}} + \delta \cdot \|\widetilde{\boldsymbol{\beta}}^{(\ell-1)} - \boldsymbol{\beta}^*\|_2 \quad \text{for } \ell \geq 2 \tag{1.3}$$

where $\boldsymbol{\beta}^*$ is the true regression coefficient, $\delta \in (0,1)$ a prefixed contraction parameter and $\sqrt{s/n}$ the order of statistical error. Tightening helps improve the accuracy only when $\|\widetilde{\boldsymbol{\beta}}^{(\ell-1)} - \boldsymbol{\beta}^*\|_2$ dominates the statistical error. The iteration effect is clearly demonstrated. Since $\widetilde{\boldsymbol{\beta}}^{(\ell)}$ is only used to create an adaptive weight for $\widetilde{\boldsymbol{\beta}}^{(\ell+1)}$, we can control the iteration complexity by solving each subproblem in (1.2) approximately. What differs from the contraction stage is that the initial estimators in the tightening stage are already in the contraction region, making the optimization algorithm enjoy geometric rate of convergence. This allows us to rapidly solve (1.2) with small optimization error.

- (Phase Transition in Algorithmic Convergence) In the contraction stage ($\ell = 1$), the optimization problem is not strongly convex and, therefore, our algorithm has only a sublinear convergence rate. Once the solution enters the contraction region, we will show that the feasible solutions are sparse and the objective function is essentially "low" dimensional and becomes (restricted) strongly convex and smooth in that region. Therefore, our algorithm has a linear convergence rate for $\ell > 1$. Indeed, this holds even for $\ell = 1$, which admits a sublinear rate until it enters into the contraction region and enjoys a linear rate of convergence after that; see Figure 2. But this estimator (for $\ell = 1$) is the estimator that corresponds to the LASSO penalty, not the folded concave penalty that we are looking for.

Contributions This paper makes several key contributions to the field of high-dimensional sparse learning:

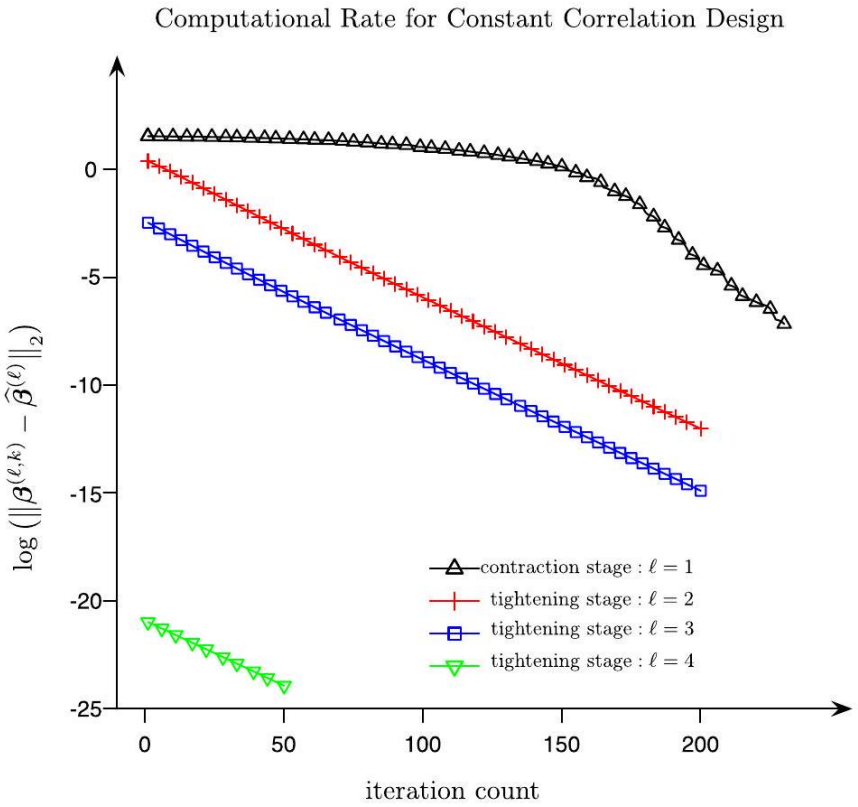

Figure 2. Computational rate of convergence in each stage for the simulation experiment specified in case 2 in Example 6.1. The x-axis is the iteration count k within the ℓ th subproblem. The phase transition from sublinear rate to liner rate of algorithmic convergence is clearly seen once the iterations enter the contraction region. Here, $\widehat{\boldsymbol{\beta}}^{(\ell)}$ is the global minimizer of the ℓ th optimization problem in (1.2) and $\boldsymbol{\beta}^{(\ell,k)}$ is its k th iteration (see Figure 3). For $\ell = 1$, the initial estimation sequence has sublinear rate and once the solution sequence enters the contraction region, it becomes linear convergent. For $\ell \geq 2$, the algorithm achieves linear rate, since all estimators $\boldsymbol{\beta}^{(\ell,k-1)}$ are in the contraction region.

- We propose the TAC algorithm, a general framework that handles both convex and nonconvex penalties in sparse learning problems, ensuring optimal statistical rates while maintaining computational tractability.
- Compared to existing methods, our approach requires weaker assumptions, eliminating the need for constraints like the ℓ_1-ball. This broadens the applicability of TAC to a wider range of sparse learning problems.
- The TAC framework provides theoretical guarantees for a broad family of loss functions, such as square loss and logistic loss, while effectively accounting for approximate optimization errors.
- We bridge the gap between nonconvex penalties (e.g., SCAD, MCP) and adaptive Lasso, offering new theoretical insights and practical solutions by unifying these methodologies under a single framework.

The rest of this paper proceeds as follows. In Section 2, we introduce TAC and its implementa-

tion. Section 3 contributes to new insights into existing methods for high-dimensional regression. In Section 4, we introduce both the localized sparse eigenvalue and localized restricted eigenvalue conditions. Statistical property and computational complexity are then presented. In Section 5, we outline the key proof strategies. Numerical simulations are provided to evaluate the proposed method in Section 6. We conclude by discussions in Section 7. All the proofs are postponed to the Appendix.

Notation. For $\mathbf{u} = (u_1, u_2, \ldots, u_d)^{\mathrm{T}} \in \mathbb{R}^d$, we define the ℓ_q-norm of $\mathbf{u}$ by $\|\mathbf{u}\|_q = \left(\sum_{j=1}^d |u_j|^q\right)^{1/q}$, where $q \in [1, \infty)$. Let $\|\mathbf{u}\|_{\min} = \min\{u_j : 1 \le j \le d\}$. For a set $\mathcal{S}$, let $|\mathcal{S}|$ denote its cardinality. We define the ℓ_0-pseudo norm of $\mathbf{u}$ as $\|\mathbf{u}\|_0 = |\operatorname{supp}(\mathbf{u})|$, where $\operatorname{supp}(\mathbf{u}) = \{j : u_j \neq 0\}$. For an index set $\mathcal{I} \subseteq \{1, \ldots, d\}$, $\mathbf{u}_{\mathcal{I}} \in \mathbb{R}^d$ is defined to be the vector whose i th entry is equal to u_i if $i \in \mathcal{I}$ and zero otherwise. Let $\mathbf{A} = [a_{i,j}] \in \mathbb{R}^{d \times d}$. For $q \ge 1$, we define $\|\mathbf{A}\|_q$ as the matrix operator q-norm of $\mathbf{A}$. For index sets $\mathcal{I}, \mathcal{J} \subseteq \{1, \ldots, d\}$, we define $\mathbf{A}_{\mathcal{I},\mathcal{J}} \in \mathbb{R}^{d \times d}$ to be the matrix whose (i, j) th entry is equal to $a_{i,j}$ if $i \in \mathcal{I}$ and $j \in \mathcal{J}$, and zero otherwise. We use $\operatorname{sign}(x)$ to denote the sign of x : $\operatorname{sign}(x) = x/|x|$ if $x \neq 0$ and $\operatorname{sign}(x) = 0$ otherwise. For two functionals $f(n, d, s)$ and $g(n, d, s)$, we denote $f(n, d, s) \gtrsim g(n, d, s)$ if $f(n, d, s) \ge C g(n, d, s)$ for a constant C; $f(n, d, s) \lesssim g(n, d, s)$ otherwise.

2 Methodology

In this paper, we assume that the loss function $\mathcal{L}(\cdot) \in \mathcal{F}_{\mathcal{L}}$, a family of general convex loss functions specified in Appendix A

2.1 Local adaptive majorize-minimization

Recall that the estimators are obtained by solving a sequence of convex programs in (1.2). We require the function $\mathrm{w}(\cdot)$ used therein to be taken from the tightening function class $\mathcal{T}$, defined as

$$\begin{aligned} \mathcal{T} = \{\mathrm{w}(\cdot) \in \mathcal{M} : \mathrm{w}(t_1) \le \mathrm{w}(t_2) \text{ for all } t_1 \ge t_2 \ge 0 \\ 0 \le \mathrm{w}(t) \le 1 \text{ if } t \ge 0, \mathrm{w}(t) = 0 \text{ if } t \le 0\} \end{aligned} \tag{2.1}$$

To fix ideas, we take $\mathcal{R}_\lambda(\boldsymbol{\beta})$ in (1.1) to be $\sum_{j=1}^d p_\lambda(|\beta_j|)$, where $p_\lambda(\cdot)$ is a folded concave penalty [Fan and Li (2001)] such as the SCAD or MCP. As discussed in Fan and Li (2001), the penalized likelihood function in (1.1) is folded concave with respect to $\boldsymbol{\beta}$, making it difficult to be maximized. We propose to use the adaptive local linear approximation (adaptive LLA) to the penalty function Fan, Xue and Zou (2014), Zou and Li (2008) and approximately solve

$$\underset{\beta}{\operatorname{argmin}} \left\{ \mathcal{L}(\boldsymbol{\beta}) + \sum_{j=1}^d p'_\lambda\left(|\widetilde{\beta}_j^{(\ell-1)}|\right)|\beta_j| \right\} \quad \text{for } 1 \le \ell \le T \tag{2.2}$$

where $\widetilde{\boldsymbol{\beta}}_j^{(\ell-1)}$ is the j th component of $\widetilde{\boldsymbol{\beta}}^{(\ell-1)}$ and $\widetilde{\boldsymbol{\beta}}^{(0)}$ can be an arbitrary bad initial value: $\widetilde{\boldsymbol{\beta}}^{(0)} = \mathbf{0}$, for example. If we assume that $\mathrm{w}(\cdot) \equiv \lambda^{-1} p'_\lambda(\cdot) \in \mathcal{T}$, such as the SCAD or MCP, then the adaptive LLA algorithm can be regarded as a special case of our general formulation (1.2). Note that the LLA algorithm with ℓ_q-penalty ($q < 1$) is not covered by our algorithm since its derivative is

unbounded at the origin, and thus $\lambda^{-1}p'_\lambda(\cdot) \notin \mathcal{T}$. The latter creates a zero-absorbing state: once a component is shrunk to zero, it will remain zero throughout the remaining iterations, as noted in Fan and Lv (2008). Of course, we can truncate the loss derivative of the loss function to resolve this issue.

We now propose a local adaptive majorize-minimization (TAC) principal, which will be repeatedly called to practically solve the optimization problem (2.2). We first review the majorize-minimization (MM) algorithm. To minimize a general function $f(\boldsymbol{\beta})$, at a given point $\boldsymbol{\beta}^{(k)}$, MM majorizes it by $g\left(\boldsymbol{\beta} \mid \boldsymbol{\beta}^{(k)}\right)$, which satisfies

$$g\left(\boldsymbol{\beta} \mid \boldsymbol{\beta}^{(k)}\right) \geq f(\boldsymbol{\beta}) \quad \text{and} \quad g\left(\boldsymbol{\beta}^{(k)} \mid \boldsymbol{\beta}^{(k)}\right) = f(\boldsymbol{\beta}^{(k)})$$

and then compute $\boldsymbol{\beta}^{(k+1)} = \operatorname{argmin}_{\boldsymbol{\beta}} \left\{ g\left(\boldsymbol{\beta} \mid \boldsymbol{\beta}^{(k)}\right) \right\}$ [Hunter and Lange (2004), Lange, Hunter and Yang (2000)]. The objective value of such an algorithm is nonincreasing in each step, since

$$f\left(\boldsymbol{\beta}^{(k+1)}\right) \overset{\text{major.}}{\leq} g\left(\boldsymbol{\beta}^{(k+1)} \mid \boldsymbol{\beta}^{(k)}\right) \overset{\text{min.}}{\leq} g\left(\boldsymbol{\beta}^{(k)} \mid \boldsymbol{\beta}^{(k)}\right) \overset{\text{init.}}{=} f(\boldsymbol{\beta}^{(k)}) \tag{2.3}$$

An inspection of the above arguments shows that the majorization requirement is not necessary. It requires only the local property

$$f\left(\boldsymbol{\beta}^{(k+1)}\right) \leq g\left(\boldsymbol{\beta}^{(k+1)} \mid \boldsymbol{\beta}^{(k)}\right) \quad \text{and} \quad g\left(\boldsymbol{\beta}^{(k)} \mid \boldsymbol{\beta}^{(k)}\right) = f(\boldsymbol{\beta}^{(k)}) \tag{2.4}$$

for the inequalities in (2.3) to hold.

Inspired by the above observation, we locally majorize (2.2) at the ℓ th step. It is similar to the iteration steps used in the (proximal) gradient method [Boyd and Vandenberghe (2004), Nesterov (2013)]. Instead of computing and storing a large Hessian matrix as in Zou and Li (2008), we majorize $\mathcal{L}(\boldsymbol{\beta})$ in (2.2) at $\widetilde{\boldsymbol{\beta}}^{(\ell-1)}$ by an isotropic quadratic function

$$\mathcal{L}\left(\widetilde{\boldsymbol{\beta}}^{(\ell-1)}\right) + \left\langle \nabla \mathcal{L}\left(\widetilde{\boldsymbol{\beta}}^{(\ell-1)}\right), \boldsymbol{\beta} - \widetilde{\boldsymbol{\beta}}^{(\ell-1)} \right\rangle + \frac{\phi}{2} \left\| \boldsymbol{\beta} - \widetilde{\boldsymbol{\beta}}^{(\ell-1)} \right\|_2^2$$

where ∇ is used to denote derivative. By Taylor's expansion, it suffices to take ϕ that is no smaller than the largest eigenvalue of $\nabla^2 \mathcal{L}\left(\widetilde{\boldsymbol{\beta}}^{(\ell-1)}\right)$. More importantly, the isotropic form also allows a simple analytic solution to the subsequent majorized optimization problem:

$$\underset{\boldsymbol{\beta} \in \mathbb{R}^d}{\operatorname{argmin}} \left\{ \mathcal{L}\left(\widetilde{\boldsymbol{\beta}}^{(\ell-1)}\right) + \left\langle \nabla \mathcal{L}\left(\widetilde{\boldsymbol{\beta}}^{(\ell-1)}\right), \boldsymbol{\beta} - \widetilde{\boldsymbol{\beta}}^{(\ell-1)} \right\rangle + \frac{\phi}{2} \left\| \boldsymbol{\beta} - \widetilde{\boldsymbol{\beta}}^{(\ell-1)} \right\|_2^2 + \sum_{j=1}^{d} p'_\lambda\left(|\widetilde{\beta}_j^{(\ell-1)}|\right) |\beta_j| \right\} \tag{2.5}$$

With $\lambda^{(\ell-1)} = \left(p'_\lambda\left(|\widetilde{\beta}_1^{(\ell-1)}|\right), \ldots, p'_\lambda\left(|\widetilde{\beta}_d^{(\ell-1)}|\right)\right)^{\mathrm{T}}$, it is easy to show that (2.5) is minimized at

$$\boldsymbol{\beta}^{(\ell,1)} = T_{\lambda^{(\ell-1)},\phi}\left(\widetilde{\boldsymbol{\beta}}^{(\ell-1)}\right) \equiv S\left(\widetilde{\boldsymbol{\beta}}^{(\ell-1)} - \phi^{-1}\nabla \mathcal{L}\left(\widetilde{\boldsymbol{\beta}}^{(\ell-1)}\right), \phi^{-1}\lambda^{(\ell-1)}\right),$$

where $S(\mathbf{x}, \lambda)$ is the soft-thresholding operator, defined by $S(\mathbf{x}, \lambda) \equiv (\operatorname{sign}(x_j) .\ \max\{|x_j| - \lambda_j, 0\})$. The simplicity of this updating rule is due to the fact that (2.5) is an unconstrained optimization problem. This is not the case in Loh and Wainwright (2015) and Wang, Liu and Zhang (2014).

Algorithm 1 The TAC algorithm in the k th iteration of the ℓ th tightening sub-problem.

procedure TAC($\boldsymbol{\lambda}^{(\ell-1)}, \boldsymbol{\beta}^{(\ell,k-1)}, \phi_0, \phi^{(\ell,k-1)}$)
  $\phi^{(\ell,k)} \leftarrow \max\left\{\phi_0, \gamma_u^{-1}\phi^{(\ell,k-1)}\right\}$
  repeat
    $\boldsymbol{\beta}^{(\ell,k)} \leftarrow T_{\boldsymbol{\lambda}^{(\ell-1)},\phi^{(\ell,k)}}(\boldsymbol{\beta}^{(\ell,k-1)})$
    if $F\left(\boldsymbol{\beta}^{(\ell,k)}, \lambda^{(\ell-1)}\right) > \Psi_{\lambda^{(\ell-1)},\phi^{(\ell,k)}}\left(\boldsymbol{\beta}^{(\ell,k)}; \boldsymbol{\beta}^{(\ell,k-1)}\right)$ **then**
      $\phi^{(\ell,k)} \leftarrow \gamma_u\phi^{(\ell,k)}$
    end if
  until $F(\boldsymbol{\beta}^{(\ell,k)}, \boldsymbol{\lambda}^{(\ell-1)}) \le \Psi_{\boldsymbol{\lambda}^{(\ell-1)},\boldsymbol{\phi}^{(\ell,k)}}\left(\boldsymbol{\beta}^{(\ell,k)}; \boldsymbol{\beta}^{(\ell,k-1)}\right)$
  return $\left\{\boldsymbol{\beta}^{(\ell,k)}, \phi^{(\ell,k)}\right\}$
end procedure

However, finding the value of $\phi \ge \left\|\nabla^2\mathcal{L}\left(\widetilde{\boldsymbol{\beta}}^{(\ell-1)}\right)\right\|_2$ is not an easy task in computation. To avoid storing and computing the largest eigenvalue of a big matrix, we now state the TAC algorithm, thanks to the local requirement (2.4). The basic idea of TAC is to start from a very small isotropic parameter ϕ_0 and then successfully inflate ϕ by a factor $\gamma_u > 1$ (say, 2). If the solution satisfies (2.4), we stop this part of the algorithm, which will make the target value nonincreasing. Since after the k th iteration, $\phi = \gamma_u^{k-1}\phi_0$, there always exists a k such that it is no larger than $\left\|\nabla^2\mathcal{L}\left(\widetilde{\boldsymbol{\beta}}^{(\ell-1)}\right)\right\|_2$. In this manner, the TAC algorithm will find a smallest iteration to make (2.4) hold.

Specifically, our proposed TAC algorithm to solve (2.5) at $\widetilde{\boldsymbol{\beta}}^{(\ell-1)}$ begins with $\phi = \phi_0$, say 10^{-6}, iteratively increases ϕ by a factor of $\gamma_u > 1$ inside the ℓ th step of optimization, and computes

$$\boldsymbol{\beta}^{(\ell,1)} = T_{\lambda^{(\ell-1)},\phi^{(\ell,k)}}(\boldsymbol{\beta}^{(\ell,0)}) \quad \text{with } \phi^{(\ell,k)} = \gamma_u^{k-1}\phi_0, \boldsymbol{\beta}^{(\ell,0)} = \widetilde{\boldsymbol{\beta}}^{(\ell-1)}$$

until the local property (2.4) holds. In our context, TAC stops when

$$\Psi_{\lambda^{(\ell-1)},\phi^{(\ell,k)}}\left(\boldsymbol{\beta}^{(\ell,1)}, \boldsymbol{\beta}^{(\ell,0)}\right) \ge F\left(\boldsymbol{\beta}^{(\ell,1)}, \lambda^{(\ell-1)}\right)$$

where $F\left(\boldsymbol{\beta}, \lambda^{(\ell-1)}\right) \equiv \mathcal{L}(\boldsymbol{\beta}) + \sum_{j=1}^d \lambda_j^{(\ell-1)}|\beta_j|$ and

$$\begin{aligned}\Psi_{\lambda^{(\ell-1)},\phi^{(\ell,k)}}\left(\boldsymbol{\beta}, \boldsymbol{\beta}^{(\ell,0)}\right) \equiv& \mathcal{L}(\boldsymbol{\beta}^{(\ell,0)}) + \left\langle\nabla\mathcal{L}(\boldsymbol{\beta}^{(\ell,0)}), \boldsymbol{\beta} - \boldsymbol{\beta}^{(\ell,0)}\right\rangle \\ &+ \frac{\phi^{(\ell,k)}}{2}\left\|\boldsymbol{\beta} - \boldsymbol{\beta}^{(\ell,0)}\right\|_2^2 + \sum_{j=1}^d \lambda_j^{(\ell-1)}|\beta_j|\end{aligned}$$

Inspired by Nesterov (2013), to accelerate TAC within the next majorizing step, we keep track of the sequence $\left\{\phi^{(\ell,k)}\right\}_{\ell,k}$ and set $\phi^{(\ell,k)} = \max\left\{\phi_0,\ \gamma_u^{-1}\phi^{(\ell,k-1)}\right\}$, with the convention that $\phi_{\ell,0} = \widetilde{\phi}_{\ell-1}$ and $\widetilde{\phi}_0 = \phi_0$, in which $\widetilde{\phi}_{\ell-1}$ is the isotropic parameter corresponding to the solution $\widetilde{\boldsymbol{\beta}}^{(\ell-1)}$. This is summarized in Algorithm 1 with a generic initial value.

The TAC algorithm solves only one local majorization step. It corresponds to moving one horizontal step in Figure 3. To solve (2.2), we need to use TAC iteratively, which we shall call the iterative TAC (TAC) algorithm, and compute a sequence of solutions $\boldsymbol{\beta}^{(\ell,k)}$ using the initial value $\boldsymbol{\beta}^{(\ell,k-1)}$. Figure 3 depicts the schematics of our algorithm: the ℓ th row corresponds to solving the ℓ th subproblem in (2.2) approximately, beginning by computing the adaptive weight $\lambda^{(\ell-1)}$. The number of iterations needed within each row will be discussed in the sequel.

Algorithm 2 TAC algorithm for each subproblem in (2.2)

procedure TAC($\boldsymbol{\lambda}^{(\ell-1)}, \boldsymbol{\beta}^{(\ell,0)}$)
  Input: $\phi_0 > 0$
  for $k = 0, 1, \ldots$ until $\omega_{\lambda^{(\ell-1)}}(\boldsymbol{\beta}^{(\ell,k)}) \leq \varepsilon$ **do**
    $\left\{\boldsymbol{\beta}^{(\ell,k)}, \phi^{(\ell,k)}\right\} \leftarrow \text{TAC}\left(\boldsymbol{\lambda}^{(\ell-1)}, \boldsymbol{\beta}^{(\ell,k-1)}, \phi_0\right)$
  end for
  Output: $\widetilde{\boldsymbol{\beta}}^{(\ell)} \leftarrow \boldsymbol{\beta}^{(\ell,k)}$
end procedure

2.2 Stopping criterion

TAC recognizes that the exact solutions to (2.2) can never be achieved in practice with algorithmic complexity control. Instead, in the ℓ th optimization subproblem, we compute the approximate solution, $\widetilde{\boldsymbol{\beta}}^{(\ell)}$, up to an optimization error ε, the choice of which will be discussed in next subsection. To calculate this approximate solution, starting from the initial value $\boldsymbol{\beta}^{(\ell,0)} = \widetilde{\boldsymbol{\beta}}^{(\ell-1)}$, the algorithm constructs a solution sequence $\left\{\boldsymbol{\beta}^{(\ell,k)}\right\}_{k=1,2,\ldots}$ using the introduced TAC algorithm; see Figure 3.

We then introduce a stopping criterion for the TAC algorithm. From optimization theory [Section 5.5 in Boyd and Vandenberghe (2004)], we know that any exact solution $\widehat{\boldsymbol{\beta}}^{(\ell)}$ to the ℓ th subproblem in (2.2) satisfies the first-order optimality condition

$$\nabla \mathcal{L}(\widehat{\boldsymbol{\beta}}^{(\ell)}) + \lambda^{(\ell-1)} \odot \boldsymbol{\xi} = \mathbf{0} \quad \text{for some } \boldsymbol{\xi} \in \partial \|\widehat{\boldsymbol{\beta}}^{(\ell)}\|_1 \in [-1, 1]^d \tag{2.6}$$

where ∂ is used to indicate the subgradient operator. The set of subgradients of a function $f : \mathbb{R}^d \to \mathbb{R}$ at a point x_0, denoted as $\partial f(x_0)$, is defined as the collection of vectors, $\boldsymbol{\xi}$, such that $f(x) - f(x_0) \geq \boldsymbol{\xi}^{\mathrm{T}}(x - x_0)$, for any x. Thus, a natural measure for suboptimality of $\boldsymbol{\beta}$ can be defined as

$$\omega_{\lambda^{(\ell-1)}}(\boldsymbol{\beta}) = \min_{\boldsymbol{\xi} \in \partial \|\boldsymbol{\beta}\|_1} \{\|\nabla \mathcal{L}(\boldsymbol{\beta}) + \lambda \odot \boldsymbol{\xi}\|_\infty\}$$

For a prefixed optimization error ε, we stop the algorithm within the ℓ th subproblem when $\omega_{\lambda^{(\ell-1)}}(\boldsymbol{\beta}^{(\ell,k)}) \leq \varepsilon$. We call $\widetilde{\boldsymbol{\beta}}^{(\ell)} \equiv \boldsymbol{\beta}^{(\ell,k)}$ an ε-optimal solution. More details can be found in Algorithm 2.

REMARK 2.1. The TAC algorithm is an early-stop variant of the ISTA algorithm to handle general loss functions and nonconvex penalties [Beck and Teboulle (2009)]. The TAC principal serves as a novel perspective for the proximal gradient method.

$$\begin{aligned}
\boldsymbol{\lambda}^{(0)}: &\quad \boldsymbol{\beta}^{(1,0)} = \mathbf{0} \xrightarrow{\text{TAC}} \boldsymbol{\beta}^{(1,1)} \xrightarrow{\text{TAC}} \cdots \xrightarrow{\text{TAC}} \boldsymbol{\beta}^{(1,k_1)} = \widetilde{\boldsymbol{\beta}}^{(1)}, \quad k_1 \lesssim \varepsilon_c^{-2} \\
\boldsymbol{\lambda}^{(1)}: &\quad \boldsymbol{\beta}^{(2,0)} = \widetilde{\boldsymbol{\beta}}^{(1)} \xrightarrow{\text{TAC}} \boldsymbol{\beta}^{(2,1)} \xrightarrow{\text{TAC}} \cdots \xrightarrow{\text{TAC}} \boldsymbol{\beta}^{(2,k_2)} = \widetilde{\boldsymbol{\beta}}^{(2)}, \quad k_2 \lesssim \log(\varepsilon_t^{-1}) \\
&\quad \vdots \qquad\qquad\qquad\qquad\qquad\qquad \vdots \\
\boldsymbol{\lambda}^{(T-1)}: &\quad \boldsymbol{\beta}^{(T,0)} = \widetilde{\boldsymbol{\beta}}^{(T-1)} \xrightarrow{\text{TAC}} \boldsymbol{\beta}^{(T,1)} \xrightarrow{\text{TAC}} \cdots \xrightarrow{\text{TAC}} \boldsymbol{\beta}^{(T,k_T)} = \widetilde{\boldsymbol{\beta}}^{(T)}, \quad k_T \lesssim \log(\varepsilon_t^{-1})
\end{aligned}$$

Figure 3. Paradigm illustration of TAC. The index k_ℓ (for $1 \leq \ell \leq T$) denotes the iteration index for the ℓ-th optimization in (2.2). The precision parameters ε_c and ε_t correspond to the contraction and tightening stages, respectively, and are discussed in detail in Section 2.3.

2.3 Tightening after contraction

From the computational perspective, optimization in (2.2) can be categorized into two stages: contraction ($\ell = 1$) and tightening ($2 \le \ell \le T$). In the contraction stage, we start from an arbitrary initial value, which can be quite remote from the underlying true parameter. We take ε as $\varepsilon_c \asymp \lambda$, reflecting the precision needed to bring the initial solution to a contracting neighborhood of the global minimum. For instance, in linear model with sub-Gaussian errors, ε_c can be taken in the order of $\sqrt{\log d/n}$. This stage aims to find a good initial estimator $\widetilde{\boldsymbol{\beta}}^{(1)}$ for the subsequent optimization subproblems in the tightening stage. Recall that $s = \|\boldsymbol{\beta}^*\|_0$ is the sparsity level. We will show in Section 4.3 that with a properly chosen λ, the approximate solution $\widetilde{\boldsymbol{\beta}}^{(1)}$, produced by the early stopped TAC algorithm, falls in the region of such good initials estimators:

$$\big\{\boldsymbol{\beta} : \|\boldsymbol{\beta} - \boldsymbol{\beta}^*\|_2 \le C\lambda\sqrt{s} \text{ and } \boldsymbol{\beta} \text{ is sparse}\big\}$$

We call this region the contraction region.

However, the estimator $\widetilde{\boldsymbol{\beta}}^{(1)}$ suffers from a suboptimal statistical rate of convergence, which is inferior to the refined one obtained by nonconvex regularization. A second stage to tighten this coarse contraction estimator into the optimal region of convergence is needed. This is achieved by the subsequent optimization ($\ell \ge 2$) and referred to as a tightening stage. Because the initial estimators are already good and sparse at each iteration of the tightening stage, the TAC algorithm at this stage enjoys a geometric rate of convergence, due to the sparse strong convexity. Therefore, the optimization error $\varepsilon = \varepsilon_t$ can be much smaller to simultaneously ensure statistical accuracy and control computational complexity. To achieve the oracle rate $\sqrt{s/n}$: ε_t must be no larger than the order of $\sqrt{1/n}$. A graphical illustration of the full algorithm is presented in Figure 3. Theoretical justifications are provided in Section 4. From this perspective, we shall also call the psuedoalgorithm in (1.2) or (2.2), combined with TAC, the tightening after contraction (TAC) algorithm.

3 New insights into existing methods

3.1 Connection to one-step local linear approximation

In the low-dimensional regime, Zou and Li (2008) shows that the one-step LLA algorithm produces an oracle estimator if the maximum likelihood estimator (MLE) is used for initialization. They thus claim that the multi-step LLA is unnecessary. However, this is not the case in high dimensions, under which an unbiased initial estimator, such as the MLE, is not available. In this paper, we show that starting from a possibly arbitrary bad initial value (such as $\mathbf{0}$), the contraction stage can produce a sparse coarse estimator. Each tightening step then refines the estimator from previous step to the optimal region of convergence by

$$\|\widetilde{\boldsymbol{\beta}}^{(\ell)} - \boldsymbol{\beta}^*\|_2 \lesssim \sqrt{\frac{s}{n}} + \delta \cdot \|\widetilde{\boldsymbol{\beta}}^{(\ell-1)} - \boldsymbol{\beta}^*\|_2 \quad \text{for } 2 \le \ell \le T \tag{3.1}$$

where $\delta \in (0, 1)$ is a prefixed contraction parameter. Unlike the one-step method in Fan, Xue and Zou (2014), the role of iteration is clearly evidenced in (3.1).

An important aspect of our algorithm (2.2) is that we use the solvable approximate solutions, $\widetilde{\boldsymbol{\beta}}^{(\ell)}$,s, rather than the exact ones, $\widehat{\boldsymbol{\beta}}^{(\ell)}$,s. In order to practically implement (2.2) for a general convex

loss function, Zou and Li (2008) propose to locally approximate $\mathcal{L}(\boldsymbol{\beta})$ by a quadratic function

$$\mathcal{L}(\widehat{\boldsymbol{\beta}}^{(0)}) + \left\langle \nabla \mathcal{L}(\widehat{\boldsymbol{\beta}}^{(0)}), \boldsymbol{\beta} - \widehat{\boldsymbol{\beta}}^{(0)} \right\rangle + \frac{1}{2}(\boldsymbol{\beta} - \widehat{\boldsymbol{\beta}}^{(0)})^{\mathrm{T}} \nabla^2 \mathcal{L}(\widehat{\boldsymbol{\beta}}^{(0)})(\boldsymbol{\beta} - \widehat{\boldsymbol{\beta}}^{(0)}) \tag{3.2}$$

where $\widehat{\boldsymbol{\beta}}^{(0)}$ is a "good" initial estimator of $\boldsymbol{\beta}^*$ and $\nabla^2 \mathcal{L}(\widehat{\boldsymbol{\beta}}^{(0)})$ is the Hessian evaluated at $\widehat{\boldsymbol{\beta}}^{(0)}$. However, in high dimensions, evaluating the $d \times d$ Hessian is not only computationally intensive but also requires a large storage cost. In addition, the optimization problem (2.2) cannot be solved analytically with approximation (3.2). We resolve these issues by proposing the isotropic quadratic approximation; see Section 2.

3.2 New insight into folded-concave regularization and adaptive Lasso

The adaptive local linear approximation (2.2) provides new insight into folded-concave regularization and adaptive Lasso. To correct the Lasso's estimation bias, foldedconcave regularization [Fan and Li(2001)] and its one-step implementation, adaptive Lasso [Fan, Xue and Zou (2014), Zou (2006), Zou and Li (2008)] have drawn much research interest due to their attractive statistical properties. For a general loss function $\mathcal{L}(\boldsymbol{\beta})$, the adaptive Lasso solves

$$\widehat{\boldsymbol{\beta}}_{\text{adapt}} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \left\{ \mathcal{L}(\boldsymbol{\beta}) + \lambda \sum_{j=1}^{d} \mathrm{w}\left(\beta_{\text{init},j}\right) |\beta_j| \right\}$$

where $\beta_{\text{init},\,j}$ is an initial estimator of β_j. We see that the adaptive Lasso is a special case of (2.2) with $\ell = 2$. Two important open questions for an adaptive Lasso are to obtain a good enough initial estimator in high dimensions and to select a suitable tuning parameter λ, which achieves the optimal statistical performance. Our solution to the first question is to use, the approximate solution to Lasso with controlled computational complexity, which corresponds to $\ell = 1$ in (2.2). For the choice of λ, Bühlmann and van de Geer (2011) suggested sequential tuning: in the first stage, they use cross validation to select the initial tuning parameter, denoted here by $\widehat{\lambda}_{\text{init,cv}}$ and the corresponding estimator $\widehat{\boldsymbol{\beta}}_{\text{init}}$; in the second stage, they again adopt cross validation to select the adaptive tuning parameter λ in the adaptive Lasso. Despite the popularity of such tuning procedure, there are no theoretical guarantees to support it. As will be shown later in Theorem 4.2 and Corollary 4.3, our framework produces optimal solution by only tuning $\lambda^{(0)} = \lambda 1$ in the contraction stage, indicating that sequential tuning may not be necessary for the adaptive Lasso if $\mathrm{w}(\cdot)$ is chosen from the tightening function class $\mathcal{T}$.

It is worth noting that a classical weight $\mathrm{w}\left(\beta_j\right) \equiv 1/\left|\beta_j\right|$ for the adaptive Lasso does not belong to the tightening function class $\mathcal{T}$. As pointed out by Fan and Lv (2008), zero is an absorbing state of the adaptive Lasso with this choice of weight function. Hence, when the Lasso estimator in the first stage misses any true positives, it will be missed forever in later stages as well. In contrast, the proposed tightening function class $\mathcal{T}$ overcomes such shortcomings by restricting the weight function $\mathrm{w}(\cdot)$ to be bounded. This phenomenon is further elaborated via our numerical experiments in Section 6. The mean square error for the adaptive Lasso can be even worse than the Lasso estimator because the adaptive Lasso may miss true positives in the strongly correlated design case.

Our framework also reveals interesting connections between the adaptive Lasso and folded-concave regularization. Specifically, consider the following foldedconcave penalized regression:

$$\min_{\boldsymbol{\beta} \in \mathbb{R}^d} \{\mathcal{L}(\boldsymbol{\beta}) + \mathcal{R}_\lambda(|\boldsymbol{\beta}|)\} \qquad \text{where } \mathcal{R}_\lambda(|\boldsymbol{\beta}|) \text{ is a folded concave penalty.} \tag{3.3}$$

We assume that $\mathcal{R}_\lambda(\cdot)$ is element-wisely decomposable, that is, $\mathcal{R}_\lambda(|\boldsymbol{\beta}|) = \sum_{k=1}^{d} p_\lambda(|\beta_k|)$. Under this assumption, using the concave duality, we can rewrite
$\mathcal{R}_\lambda(|\boldsymbol{\beta}|)$ as

$$\mathcal{R}_\lambda(|\boldsymbol{\beta}|) = \inf_{\mathbf{v}} \left\{ |\boldsymbol{\beta}|^{\mathrm{T}} \mathbf{v} - \mathcal{R}_\lambda^\star(\mathbf{v}) \right\} \tag{3.4}$$

where $\mathcal{R}_\lambda^\star(\cdot)$ is the dual of $\mathcal{R}_\lambda(\cdot)$. By the duality theory, we know that the minimum of (3.4) is achieved at $\widehat{\mathbf{v}} = \nabla \mathcal{R}_\lambda(|\boldsymbol{\mu}|)|_{\boldsymbol{\mu}=\boldsymbol{\beta}}$. We can employ (3.4) to reformulate (3.3) as

$$(\widehat{\boldsymbol{\beta}}, \widehat{\mathbf{v}}) = \operatorname*{argmin}_{\boldsymbol{\beta}, \mathbf{v}} \left\{ \mathcal{L}(\boldsymbol{\beta}) + \mathbf{v}^{\mathrm{T}} |\boldsymbol{\beta}| - \mathcal{R}_\lambda^\star(\mathbf{v}) \right\}.$$

The optimization above can then be solved by exploiting the alternating minimization scheme. In particular, we repeatedly apply the following two steps:

(1) Optimize over $\boldsymbol{\beta}$ with $\mathbf{v}$ fixed: $\widehat{\boldsymbol{\beta}}^{(\ell)} = \operatorname{argmin}_{\boldsymbol{\beta}} \left\{ \mathcal{L}(\boldsymbol{\beta}) + \left(\widehat{\mathbf{v}}^{(\ell-1)}\right)^{\mathrm{T}} |\boldsymbol{\beta}| \right\}$.

(2) Optimize over $\mathbf{v}$ with $\boldsymbol{\beta}$ fixed. We can obtain closed form solution: $\mathbf{v}^{(\ell)} = \nabla \mathcal{R}_\lambda(|\boldsymbol{\mu}|)\big|_{\boldsymbol{\mu}=\widehat{\boldsymbol{\beta}}^{(\ell)}}$

This is a special case of (1.2) if we take $\mathrm{w}(\boldsymbol{\beta}) = \lambda^{-1} \nabla \mathcal{R}_\lambda(|\boldsymbol{\mu}|)\big|_{\boldsymbol{\mu}=\boldsymbol{\beta}}$ and let ℓ grow until convergence. Therefore, with a properly chosen weight function $\mathrm{w}(\cdot)$, our proposed algorithm bridges the adaptive Lasso and folded-concave penalized regression together under different choices of ℓ. In Corollary 4.3 , we will prove that, when ℓ is in the order of $\log(\lambda\sqrt{n})$, then the proposed estimator enjoys the optimal statistical rate $\left\| \widehat{\boldsymbol{\beta}}^{(\ell)} - \boldsymbol{\beta}^* \right\|_2 \propto \sqrt{s/n}$, under mild conditions.

4 Theoretical results

We establish the optimal statistical rate of convergence and the computational complexity of the proposed algorithm. To establish these results in a general framework, we first introduce the localized versions of the sparse eigenvalue and restricted eigenvalue conditions.

4.1 Localized eigenvalues and assumptions

The sparse eigenvalue condition [Zhang and Zhang (2012)] is commonly used in the analysis of sparse learning problems. However, it is only valid for the least square loss. For a general loss function, the Hessian matrix depends on the parameter $\boldsymbol{\beta}$ and can become nearly singular in certain regions. For example, the Hessian matrix of the logistic loss is

$$\nabla^2 \mathcal{L}(\boldsymbol{\beta}) = \frac{1}{n} \sum_{i=1}^{n} \mathbf{x}_i \mathbf{x}_i^{\mathrm{T}} \cdot \frac{1}{1 + \exp\left(-\mathbf{x}_i^{\mathrm{T}} \boldsymbol{\beta}\right)} \cdot \frac{1}{1 + \exp\left(\mathbf{x}_i^{\mathrm{T}} \boldsymbol{\beta}\right)}$$

which tends to zero as $\|\boldsymbol{\beta}\|_2 \to \infty$, no matter what the data are. One of our key theoretical observations is that: what we really need are the localized conditions around the true parameters $\boldsymbol{\beta}^*$, which we now introduce.

4.1.1 Localized sparse eigenvalue

DEFINITION 4.1 (Localized sparse eigenvalue, LSE). *The localized sparse eigenvalues are defined as*

$$\rho_+(m,r) = \sup_{\mathbf{u},\boldsymbol{\beta}} \left\{ \mathbf{u}_J^{\mathrm{T}} \nabla^2 \mathcal{L}(\boldsymbol{\beta}) \mathbf{u}_J : \|\mathbf{u}_J\|_2^2 = 1, |J| \le m, \|\boldsymbol{\beta} - \boldsymbol{\beta}^*\|_2 \le r \right\}$$

$$\rho_-(m,r) = \inf_{\mathbf{u},\boldsymbol{\beta}} \left\{ \mathbf{u}_J^{\mathrm{T}} \nabla^2 \mathcal{L}(\boldsymbol{\beta}) \mathbf{u}_J : \|\mathbf{u}_J\|_2^2 = 1, |J| \le m, \|\boldsymbol{\beta} - \boldsymbol{\beta}^*\|_2 \le r \right\}$$

Both $\rho_+(m,r)$ and $\rho_-(m,r)$ depend on the Hessian matrix $\nabla^2\mathcal{L}(\boldsymbol{\beta})$, the true coefficient $\boldsymbol{\beta}^*$, the sparsity level m, and an extra locality parameter r. They reduce to the commonly-used sparse eigenvalues when $\nabla^2\mathcal{L}(\boldsymbol{\beta})$ does not change with $\boldsymbol{\beta}$ as in the quadratic loss. The following assumption specifies the LSE condition in detail. Recall that $s = \|\boldsymbol{\beta}^*\|_0$.

ASSUMPTION 4.1 (Localized sparse eigenvalue, LSE). *We say the LSE condition holds if there exist an integer $\widetilde{s} \ge cs$ for some constant c, r and a constant C such that*

$$0 < \rho_* \le \rho_-(2s + 2\widetilde{s}, r) < \rho_+(2s + 2\widetilde{s}, r) \le \rho^* < +\infty \quad \text{and}$$
$$\rho_+(\widetilde{s}, r)/\rho_-(2s + 2\widetilde{s}, r) \le 1 + C\widetilde{s}/s$$

Assumption 4.1 is standard for linear regression problems and is commonly referred to as the sparse eigenvalue condition when $r = \infty$. Such conditions have been employed by Bickel, Ritov and Tsybakov (2009), Loh and Wainwright (2015), Negahban et al. (2012), Raskutti, Wainwright and Yu (2010), Wang, Liu and Zhang (2014). The newly proposd LSE condition, to the best of our knowledge, is the weakest one in the literature.

4.1.2 Localized restricted eigenvalue

In this section, we introduce the localized version of the restricted eigenvalue condition [Bickel, Ritov and Tsybakov (2009)]. This is an alternative condition to Assumption 4.1 that allows us to handle general Hessian matrices that depend on $\boldsymbol{\beta}$, under which the theoretical properties can be carried out parallelly.

DEFINITION 4.2 (Localized restricted eigenvalue, LRE). *The localized restricted eigenvalue is defined as*

$$\kappa_+(m,\gamma,r) = \sup_{\mathbf{u},\boldsymbol{\beta}} \left\{ \mathbf{u}^{\mathrm{T}} \nabla^2 \mathcal{L}(\boldsymbol{\beta}) \mathbf{u} : (\boldsymbol{u}, \boldsymbol{\beta}) \in \mathcal{C}(m,\gamma,r) \right\}$$

$$\kappa_-(m,\gamma,r) = \inf_{\mathbf{u},\boldsymbol{\beta}} \left\{ \mathbf{u}^{\mathrm{T}} \nabla^2 \mathcal{L}(\boldsymbol{\beta}) \mathbf{u} : (\boldsymbol{u}, \boldsymbol{\beta}) \in \mathcal{C}(m,\gamma,r) \right\}$$

where $\mathcal{C}(m,\gamma,r) \equiv \{\mathbf{u}, \boldsymbol{\beta} : S \subseteq J, |J| \le m, \|\mathbf{u}_{J^c}\|_1 \le \gamma \|\mathbf{u}_J\|_1, \|\boldsymbol{\beta} - \boldsymbol{\beta}^*\|_2 \le r\}$ is a local ℓ_1 cone.

Similarly, the localized restricted eigenvalue reduces to the restricted eigenvalue when $\nabla^2\mathcal{L}(\boldsymbol{\beta})$ does not depend on $\boldsymbol{\beta}$. We say the localized restricted eigenvalue condition holds if there exists m, γ, r such that $0 < \kappa_-(m,\gamma,r) \le \kappa_+(m,\gamma,r) < \infty$. In Appendix B, we give a geometric explanation of the local ℓ_1 cone, $\mathcal{C}(m,\gamma,r)$, and the corresponding localized analysis.

4.2 Statistical theory

In this section, we provide theoretical analysis of the proposed estimator under the LSE condition. For completeness, in Appendix B, we also establish similar results under localized restricted eigenvalue condition. We begin with the contraction stage. Recall that the initial value $\widetilde{\boldsymbol{\beta}}^{(0)}$ is taken as $\mathbf{0}$ for simplicity. We need the following assumption on the tightening function.

ASSUMPTION 4.2. *Assume that* $\mathrm{w}(\cdot) \in \mathcal{T}$ *and* $\mathrm{w}(u) \geq 1/2$ *for* $u = 18\rho_*^{-1}\delta^{-1}\lambda$. *Here* $\mathcal{T}$ *is the tightening function class defined in (2.1).*

Our first result characterizes the statistical convergence rate of the estimator in the contraction stage. The key ideas of the proofs are outlined in Section 5. Other technical lemmas and details can be found in the Appendix.

Proposition 4.1 (Statistical rate in the contraction stage)**.** *Suppose that Assumption 4.1 holds. If* λ, ε *and* r *satisfy*

$$4\left(\|\nabla\mathcal{L}(\boldsymbol{\beta}^*)\|_\infty + \varepsilon\right) \leq \lambda \leq r\rho_*/(18\sqrt{s}) \tag{4.1}$$

then any ε_c*-optimal solution* $\widetilde{\boldsymbol{\beta}}^{(1)}$ *satisfies*

$$\left\|\widetilde{\boldsymbol{\beta}}^{(1)} - \boldsymbol{\beta}^*\right\|_2 \leq 18\rho_*^{-1}\lambda\sqrt{s} \lesssim \lambda\sqrt{s}$$

The result above is a deterministic statement. Its proof is omitted as it directly follows from Lemma 5.1 with $\ell = 1$ and $\mathcal{E}_1$ there to be S, the support of the true parameter $\boldsymbol{\beta}^*$. The proof of Lemma 5.1 can be found in Appendix B. In Proposition 4.1, the approximation error ε_c, can be taken to be the order of $\lambda \asymp \sqrt{\log d/n}$ in the sub-Gaussian noise case. The contraction stage ensures that the ℓ_2 estimation error is proportional to $\lambda\sqrt{s}$, which is identical to the optimal rate of convergence for the Lasso estimator [Bickel, Ritov and Tsybakov (2009), Zhang (2009)]. Our result can be regarded as a generalization of the usual Lasso analysis to more general losses, which satisfy the localized sparse eigenvalue condition. We are ready to present the main theorem, which demonstrates the effects of optimization error, shrinkage bias and tightening steps on the statistical rate.

THEOREM 4.2 (Optimal statistical rate)**.** *Suppose Assumptions 4.1 and 4.2 hold. If* $4\left(\|\nabla\mathcal{L}(\boldsymbol{\beta}^*)\|_\infty + (\varepsilon_t \vee \varepsilon_c)\right) \leq \lambda \lesssim r/\sqrt{s}$*, then any* ε_t*-optimal solution* $\widetilde{\boldsymbol{\beta}}^{(\ell)}, \ell \geq 2$*, satisfies the following* δ*-contraction property:*

$$\|\widetilde{\boldsymbol{\beta}}^{(\ell)} - \boldsymbol{\beta}^*\|_2 \leq C\left(\|\nabla\mathcal{L}(\boldsymbol{\beta}^*)_S\|_2 + \varepsilon_t\sqrt{s} + \lambda\left\|\mathrm{w}(|\boldsymbol{\beta}^*_S| - u)\right\|_2\right) + \delta\|\widetilde{\boldsymbol{\beta}}^{(\ell-1)} - \boldsymbol{\beta}^*\|_2$$

where C *is a constant and* $u = 18\rho_*^{-1}\delta^{-1}\lambda$. *Consequently, there exists a constant* C' *such that*

$$\|\widetilde{\boldsymbol{\beta}}^{(\ell)} - \boldsymbol{\beta}^*\|_2 \leq C'\Big(\underbrace{\|\nabla\mathcal{L}(\boldsymbol{\beta}^*)_S\|_2}_{\text{oracle rate}} + \overbrace{\varepsilon_t\sqrt{s}}^{\text{opt err}} + \underbrace{\lambda\left\|\mathrm{w}(|\boldsymbol{\beta}^*_S| - u)\right\|_2}_{\text{coefficient effect}}\Big) + \overbrace{2C'\delta^{\ell-1}\lambda\sqrt{s}}^{\text{tightening effect}}.$$

The effect of the tightening stage can be clearly seen from the theorem above: each tightening step induces a δ-contraction property, which reduces the influence of the estimation error from the previous step by a δ-fraction. Therefore, in order to achieve the oracle rate $\sqrt{s/n}$, we shall carefully choose the optimization error such that $\varepsilon_t \lesssim \|\nabla\mathcal{L}(\boldsymbol{\beta}^*)\|_2/\sqrt{s}$ and make the tightening iterations ℓ large enough. As a corollary, we give the explicit statistical rate under the quadratic loss $\mathcal{L}(\boldsymbol{\beta}) = (2n)^{-1}\|\mathbf{y} - \mathbf{X}\boldsymbol{\beta}\|_2^2$. In this case, we take $\lambda \asymp \sqrt{n^{-1}\log d}$ so that the scaling condition (4.1) holds with high probability. We use sub-Gaussian $\left(0, \sigma^2\right)$ to denote a sub-Gaussian distribution random variable with mean 0 and variance proxy σ^2.

COROLLARY 4.3 (Optimal statistical rate). *Let $y_i = \mathbf{x}_i^{\mathrm{T}}\boldsymbol{\beta}^* + \epsilon_i, 1 \le i \le n$, be independently and identically distributed sub-Gaussian random variables with $\epsilon_i \sim$ sub-Gaussian $\left(0, \sigma^2\right)$. The columns of $\mathbf{X}$ are normalized such that $\max_j \|\mathbf{X}_{*j}\|_2 \le \sqrt{n}$. Assume there exists an $\gamma > 0$ such that $\|\boldsymbol{\beta}_S^*\|_{\min} \ge u + \gamma\lambda$ and $\mathrm{w}(\gamma\lambda) = 0$. Under Assumptions 4.1 and 4.2, if $\lambda \asymp \sqrt{n^{-1}\log d}, \varepsilon_t \le \sqrt{1/n}$ and $T \gtrsim \log\log d$, then with probability at least $1 - 2d^{-\eta_1} - 2\exp\{-\eta_2 s\}$, $\widetilde{\boldsymbol{\beta}}^{(T)}$ must satisfy*

$$\left\|\widetilde{\boldsymbol{\beta}}^{(T)} - \boldsymbol{\beta}^*\right\|_2 \lesssim \sqrt{s/n}$$

where η_1 and η_2 are positive constants.

Corollary 4.3 indicates that TAC can achieve the oracle statistical rate $\sqrt{s/n}$ as if the support for the true coefficients were known in advance. To achieve such rate, we require $\varepsilon_c \lesssim \sqrt{\log d/n}$ and $\varepsilon_t \lesssim \sqrt{1/n}$. In other words, we need only a more accurate estimator in the tightening stage rather than in both stages. This will help us to relax the computational burden, which will be discussed in detail in Theorem 4.7. Our last result concerns the oracle property of the obtained estimator $\widetilde{\boldsymbol{\beta}}^{(\ell)}$ for ℓ large enough, with the proof postponed to Appendix B. We first define the oracle estimator $\widehat{\boldsymbol{\beta}}^{\circ}$ as

$$\widehat{\boldsymbol{\beta}}^{\circ} = \underset{\mathrm{supp}(\boldsymbol{\beta})=S}{\mathrm{argmin}}\ \mathcal{L}(\boldsymbol{\beta})$$

THEOREM 4.4 (Strong oracle property). *Suppose Assumptions 4.1 and 4.2 hold. Assume $\|\boldsymbol{\beta}_S^*\|_{\min} \ge u + \gamma\lambda$ and $\mathrm{w}(\gamma\lambda) = 0$ for some constant γ. Let $4\left(\|\nabla\mathcal{L}(\widehat{\boldsymbol{\beta}}^{\circ})\|_\infty + \varepsilon_c \vee \varepsilon_t\right) \le \lambda \lesssim r/\sqrt{s}$ and $\varepsilon_t \le \lambda/\sqrt{s}$. If $\|\widehat{\boldsymbol{\beta}}^{\circ} - \boldsymbol{\beta}^*\|_{\max} \le \eta_n \lesssim \lambda$, then for ℓ large enough such that $\ell \gtrsim \log\{(1 + \varepsilon_c/\lambda)\sqrt{s}\}$, we have*

$$\widetilde{\boldsymbol{\beta}}^{(\ell)} = \widehat{\boldsymbol{\beta}}^{\circ}$$

The theorem above is again a deterministic result. Large probability bound can be obtained by bounding the probability of the event $\left\{4(\|\nabla\mathcal{L}(\widehat{\boldsymbol{\beta}}^{\circ})\|_\infty + (\varepsilon_c \vee \varepsilon_t)) \le \lambda\right\}$. The assumption that $\|\widehat{\boldsymbol{\beta}}^{\circ} - \boldsymbol{\beta}^*\|_{\max} \lesssim \lambda$ is very mild, because the oracle estimator only depends on the intrinsic dimension s rather than d. For instance, under linear model with sub-Gaussian errors, it can be shown that $\|\widehat{\boldsymbol{\beta}}^{\circ} - \boldsymbol{\beta}\|_{\max} \le \sqrt{\log s/n}$ with high probability.

Theorem 4.4 implies that the oracle estimator $\widehat{\boldsymbol{\beta}}^{\circ}$ is a fixed point of the TAC algorithm, namely, once the initial estimator is $\widehat{\boldsymbol{\beta}}^{\circ}$, the next iteration produces the same estimator. This is in the same spirit as that proved in Fan, Xue and Zou (2014).

4.3 Computational theory

In this section, we analyze the computational rate for all of our approximate solutions. We start with the following assumption.

ASSUMPTION 4.3 (Strong oracle property). *$\nabla\mathcal{L}(\boldsymbol{\beta})$ is locally ρ_c-Lipschitz continuous, that is,*

$$\|\nabla\mathcal{L}(\boldsymbol{\beta}_1) - \nabla\mathcal{L}(\boldsymbol{\beta}_2)\|_2 \le \rho_c\|\boldsymbol{\beta}_1 - \boldsymbol{\beta}_2\|_2 \quad \text{for } \boldsymbol{\beta}_1, \boldsymbol{\beta}_2 \in B_2(R/2, \boldsymbol{\beta}^*) \tag{4.2}$$

where ρ_c is the Lipschitz constant and $R \lesssim \|\boldsymbol{\beta}^*\|_2 + \lambda\sqrt{s}$.

We then give the explicit iteration complexity of the contraction stage in the following proposition. Recall the definition of ϕ_0 and γ_u in Algorithm 2.1, and ρ_* in Assumption 4.1.

Proposition 4.5 (Sublinear rate in the contraction stage)**.** *Assume that Assumptions 4.1 and 4.3 hold. Let* $4\left(\|\nabla\mathcal{L}(\boldsymbol{\beta}^*)\|_\infty + \varepsilon_c\right) \le \lambda \lesssim r/\sqrt{s}$*. To achieve an approximate local solution* $\widetilde{\boldsymbol{\beta}}^{(1)}$ *such that* $\omega_{\lambda^{(0)}}(\widetilde{\boldsymbol{\beta}}^{(1)}) \le \varepsilon_c$ *in the contraction stage, we need no more than* $\left((1+\gamma_u)\, R\rho_c/\varepsilon_c\right)^2$ *TAC iterations, where* ρ_c *is a constant defined in (4.2).*

The sublinear rate is due to the lack of strong convexity of the loss function in the contraction stage, because we allow starting with arbitrary bad initial value, say 0 . Once it enters the contracting region (aka, the tightening stage), the problem becomes sparse strongly convex (see Proposition B. 3 in Appendix B), which endows the algorithm a linear rate of convergence. This is empirically demonstrated in Figure 2. Our next proposition gives a formal statement on the geometric convergence rate for each subproblem in the tightening stage.

Proposition 4.6 (Geometric rate in the tightening stage)**.** *Suppose that the same conditions for Theorem 4.2 hold. To obtain an approximate solution* $\widetilde{\boldsymbol{\beta}}^{(\ell)}$ *satisfying* $\omega_{\lambda^{(\ell-1)}}(\widetilde{\boldsymbol{\beta}}^{(\ell)}) \le \varepsilon$ *in each step of the* ℓ *th tightening stage* ($\ell \ge 2$)*, we need at most* $C' \log\left(C''\lambda\sqrt{s}/\varepsilon\right)$ *TAC iterations, where* C' *and* C'' *are two positive constants.*

Proposition 4.6 suggests that we only need to conduct a logarithmic number of TAC iterations in each tightening step. Simply combining the computational rate in both the contraction and the tightening stages, we manage to obtain the global computational complexity.

THEOREM 4.7 (Geometric rate in the tightening stage)**.** *Assume that* $\lambda\sqrt{s} = o(1)$*. Suppose that the same conditions for Theorem 4.2 hold. To achieve an approximate solution* $\widetilde{\boldsymbol{\beta}}^{(\ell)}$ *such that* $\omega_{\lambda^{(0)}}(\widetilde{\boldsymbol{\beta}}^{(1)}) \le \varepsilon_c \lesssim \lambda$ *and* $\omega_{\lambda^{(k-1)}}(\widetilde{\boldsymbol{\beta}}^{(k)}) \le \varepsilon_t \lesssim \sqrt{1/n}$ *for* $2 \le k \le T$*, the total number of TAC iterations we need is at most*

$$C'\frac{1}{\varepsilon_c^2} + C''(T-1)\log\left(\frac{1}{\varepsilon_t}\right)$$

where C' *and* C'' *are two positive constants, and* $T \asymp \log(\lambda\sqrt{n})$*.*

REMARK 4.8. *We complete this section with a remark on the sublinear rate in the contraction stage. Without further structures, the sublinear rate in the first stage is the best possible one for the proposed optimization procedure when* λ *is held fixed. Linear rate can be achieved when we start from a sufficiently good initial value. Another strategy is to use the path-following algorithm which is developed in Wang, Liu and Zhang (2014), where they gradually reduce the size of* λ *to ensure the solution sequence to be sparse.*

5 Proof strategy for main results

In this section, we present the proof strategies for the main statistical and computational theorems, with technical lemmas and other details left in the Appendix.

5.1 Proof strategy for statistical recovery result in Section 4.2

Proposition 4.1 indicates that the contraction estimator suffers from a suboptimal rate of convergence $\lambda\sqrt{s}$. The tightening stage helps refine the statistical rate adaptively. To suppress the noise

in the ℓ th subproblem, it is necessary to control $\min_j \left\{|\widetilde{\beta}_j^{(\ell-1)}| : j \in S^c\right\}$ in high dimensions. For this, we construct an entropy set $\mathcal{E}_\ell$ of S in each tightening subproblem to bound the magnitude of $\left\|\lambda^{(\ell-1)}_{\mathcal{E}_\ell^c}\right\|_{\min}$. The entropy set at the ℓ th step is defined as

$$\mathcal{E}_\ell = S \cup \left\{j : \lambda_j^{(\ell-1)} < \lambda \mathrm{w}(u), u = 18\delta^{-1}\rho_*^{-1}\lambda \propto \lambda\right\} \tag{5.1}$$

Under mild conditions, we will show that $|\mathcal{E}_\ell| \le 2s$ and $\left\|\lambda^{(\ell)}_{\mathcal{E}_\ell^c}\right\|_{\min} \ge \lambda \mathrm{w}(u) \ge \lambda/2$, which is more precisely stated in the following lemma.

LEMMA 5.1. *Suppose that Assumptions 4.1 and 4.2 hold. If* $4\left(\|\nabla\mathcal{L}(\boldsymbol{\beta}^*)\|_\infty + \ \varepsilon_t \vee \varepsilon_c\right) \le \lambda \lesssim r/\sqrt{s}$*, we must have* $|\mathcal{E}_\ell| \le 2s$*, and the* ε*-optimal solution* $\widetilde{\boldsymbol{\beta}}^{(\ell)}$ *satisfies*

$$\begin{aligned}\|\widetilde{\boldsymbol{\beta}}^{(\ell)} - \boldsymbol{\beta}^*\|_2 &\le 12\rho_*^{-1}\left(\|\lambda_S^{(\ell-1)}\|_2 + \|\nabla\mathcal{L}(\boldsymbol{\beta}^*)_{\mathcal{E}_\ell}\|_2 + \varepsilon\sqrt{|\mathcal{E}_\ell|}\right)\\ &\le 18\rho_*^{-1}\lambda\sqrt{s} \lesssim \lambda\sqrt{s}.\end{aligned}$$

Lemma 5.1 bounds $\|\widetilde{\boldsymbol{\beta}}^{(\ell)} - \boldsymbol{\beta}^*\|_2$ in terms of $\|\lambda_S^{(\ell-1)}\|_2$, which is further upper bounded by the order of $\lambda\sqrt{s}$. The rate $\lambda\sqrt{s}$ coincides with the convergence rate of the contraction estimator. Later, we will exploit this result in our localized analysis to secure that all the approximate solutions $\{\widetilde{\boldsymbol{\beta}}^{(\ell)}\}_{\ell=1,\ldots,T}$ fall in a local ℓ_2-ball centered at β^* with radius $r \gtrsim \lambda\sqrt{s}$.

The next lemma further bounds $\|\lambda_S^{(\ell-1)}\|_2$ using functionals of $\widetilde{\boldsymbol{\beta}}^{(\ell-1)}$, which connects the adaptive regularization parameter to the estimator from previous steps.

LEMMA 5.2. *Assume* $\mathrm{w} \in \mathcal{T}$*. Let* $\lambda_j^{(\ell-1)} = \lambda\mathrm{w}\left(|\widetilde{\beta}_j^{(\ell-1)}|\right)$ *for* $\widetilde{\boldsymbol{\beta}}^{(\ell-1)}$*, then for any norm* $\|\cdot\|_*$*, we have*

$$\|\lambda_S^{(\ell-1)}\|_* \le \lambda\left\|\mathrm{w}(|\boldsymbol{\beta}_S^*| - u)\right\|_* + \lambda u^{-1}\left\|\boldsymbol{\beta}_S^* - \widetilde{\boldsymbol{\beta}}_S^{(\ell-1)}\right\|_*$$

where $\mathrm{w}(|\boldsymbol{\beta}_S^*| - u) \equiv \left(\mathrm{w}\left(|\beta_j^*| - u\right)\right)_{j\in S}$.

Lemma 5.2 bounds the tightening weight $\lambda^{(\ell-1)}$ in the ℓ th subproblem by two terms. The first term describes the coefficient effects: when the coefficients are large enough (in absolute value) such that $\|\boldsymbol{\beta}^*\|_{\min} \ge u + \gamma\lambda$ and $\mathrm{w}(\gamma\lambda) = 0$, it becomes 0 . The second term concerns the estimation error of the estimator from previous step. Combing the above two lemmas, we prove that $\widetilde{\boldsymbol{\beta}}^{(\ell)}$ benefits from the tightening stage and possesses a refined statistical rate of convergence. The proof of Corollary 4.3 is left in Appendix.

Proof of Theorem 4.2. Applying Lemma 5.1, we obtain the size of the entropy set $\mathcal{E}_\ell$ [see the definition in (5.1)] is bounded by $2s$ and

$$\|\widetilde{\boldsymbol{\beta}}^{(\ell)} - \boldsymbol{\beta}^*\|_2 \le C_1\left(\|\lambda_S^{(\ell-1)}\|_2 + \|\nabla\mathcal{L}(\boldsymbol{\beta}^*)_{\mathcal{E}_\ell}\|_2 + \varepsilon_t\sqrt{|\mathcal{E}_\ell|}\right) \lesssim \lambda\sqrt{s} \tag{5.2}$$

where $C_1 = 12\rho_*^{-1}$. Using Lemma 5.2 yields that

$$\|\lambda_S^{(\ell-1)}\|_2 \le \lambda\left\|\mathrm{w}(|\boldsymbol{\beta}_S^*| - u)\right\|_2 + \lambda u^{-1}\left\|(\widetilde{\boldsymbol{\beta}}^{(\ell-1)} - \boldsymbol{\beta}^*)_S\right\|_2$$

Plugging the inequality above into (5.2) obtains that

$$\left\|\widetilde{\boldsymbol{\beta}}^{(\ell)}-\boldsymbol{\beta}^*\right\|_2 \leq C_1(\underbrace{\|\nabla\mathcal{L}(\boldsymbol{\beta}^*)_{\mathcal{E}_\ell}\|_2+\varepsilon_t\sqrt{|\mathcal{E}_\ell|}}_{\mathrm{I}}+\lambda\left\|\mathrm{w}(|\boldsymbol{\beta}^*_S|-u)\right\|_2) \tag{5.3}$$
$$+C_1\lambda u^{-1}\left\|(\widetilde{\boldsymbol{\beta}}^{(\ell-1)}-\boldsymbol{\beta}^*)_S\right\|_2$$

We now simplify the inequality above by providing an upper bound for term I. Decomposing the support set $\mathcal{E}_\ell$ into S and $\mathcal{E}_\ell\backslash S$ and applying the triangle inequality along with the Hölder inequality, we have

$$\mathrm{I}\leq\|\nabla\mathcal{L}(\boldsymbol{\beta}^*)_S\|_2+\varepsilon_t\sqrt{s}+\left(\|\nabla\mathcal{L}(\boldsymbol{\beta}^*)\|_\infty+\varepsilon_t\right)\sqrt{\mathcal{E}_\ell/S} \tag{5.4}$$

Following the proof of Lemma 5.1 in Appendix B, $\sqrt{|\mathcal{E}_\ell\backslash S|}$ can be bounded by

$$\left\|\widetilde{\boldsymbol{\beta}}^{(\ell-1)}_{\mathcal{E}_\ell\backslash S}\right\|_2/u\leq\|\widetilde{\boldsymbol{\beta}}^{(\ell-1)}-\boldsymbol{\beta}^*\|_2/u\quad\text{ where } u=18\rho^{*-1}\delta^{-1}\lambda\propto\lambda$$

Therefore, (5.4) can be simplified to

$$\mathrm{I}\leq\|\nabla\mathcal{L}(\boldsymbol{\beta}^*)_S\|_2+\varepsilon_t\sqrt{s}+\frac{\lambda}{4u}\|\widetilde{\boldsymbol{\beta}}^{(\ell-1)}-\boldsymbol{\beta}^*\|_2$$

which, combining with (5.3), yields the contraction property with δ. Consequently, we obtain

$$\begin{aligned}
&\|\widetilde{\boldsymbol{\beta}}^{(\ell)}-\boldsymbol{\beta}^*\|_2\\
&\quad\leq C\left(\|\nabla\mathcal{L}(\boldsymbol{\beta}^*)_{\mathcal{E}_\ell}\|_2+\varepsilon_t\sqrt{s}+\lambda\left\|\mathbf{w}_S(|\boldsymbol{\beta}^*_S|-u)\right\|_2\right)+\delta^{\ell-1}\left\|\widetilde{\boldsymbol{\beta}}^{(1)}-\boldsymbol{\beta}^*\right\|_2\\
&\quad\leq C\left(\|\nabla\mathcal{L}(\boldsymbol{\beta}^*)_{\mathcal{E}_\ell}\|_2+\varepsilon_t\sqrt{s}+\lambda\left\|\mathbf{w}_S(|\boldsymbol{\beta}^*_S|-u)\right\|_2\right)+C\delta^{\ell-1}\lambda\sqrt{s}
\end{aligned}$$

where $C=C_1/(1-\delta)$ and the last inequality follows from Proposition 4.1. The proof is complete. □

5.2 Proof strategy for computational result in Section 4.3

In this section, we present the sketch for the proofs of the results in Section 4.3. We start with the contraction stage. The next lemma shows that the contraction stage enjoys a sublinear rate of convergence. The proof can be found in Appendix C.

Lemma 5.3. *Recall that* $F(\boldsymbol{\beta},\boldsymbol{\lambda})=\mathcal{L}(\boldsymbol{\beta})+\sum_{j=1}^d\lambda_j|\beta_j|$. *We have*

$$F\left(\boldsymbol{\beta}^{(1,k)},\lambda^{(0)}\right)-F\left(\widehat{\boldsymbol{\beta}}^{(1)},\lambda^{(0)}\right)\leq\frac{\phi_c}{2k}\left\|\boldsymbol{\beta}^{(1,0)}-\widehat{\boldsymbol{\beta}}^{(1)}\right\|_2^2$$

The result above suggests that the optimization error decreases to zero at the rate of $1/k$, while Proposition 4.1 indicates that the best statistical rate for the contraction stage is only in the order of $\lambda\sqrt{s}$. Therefore, one can early stop the TAC iterations in the contraction stage as soon as it enters the contraction region $\{\boldsymbol{\beta}:\|\boldsymbol{\beta}-\boldsymbol{\beta}^*\|_2\lesssim C\lambda\sqrt{s},\boldsymbol{\beta}\text{ is sparse}\}$. It is this lemma that helps characterize the iteration complexity in terms of the total number of TAC updates needed in the contraction stage; see Proposition 4.5.

To utilize the localized sparse eigenvalue condition in the tightening stage, we need the following proposition, which characterizes the sparsity of all the approximate solutions produced by the contraction stage.

Lemma 5.4. *Assume that Assumption 4.1 holds. If* $4\left(\|\nabla\mathcal{L}(\boldsymbol{\beta}^*)\|_\infty + \varepsilon_c\right) \le \lambda \lesssim r/\sqrt{s}$, *then* $\widetilde{\boldsymbol{\beta}}^{(1)}$ *in the contraction stage is* $s + \widetilde{s}$ *sparse. In particular, we have* $\left\|(\widetilde{\boldsymbol{\beta}}^{(1)})_{S^c}\right\|_0 \le \widetilde{\boldsymbol{s}}$.

Together with Proposition 4.1, it ensures that the contraction estimator $\widetilde{\boldsymbol{\beta}}^{(1)}$ falls in the contraction region $\{\boldsymbol{\beta} : \|\boldsymbol{\beta} - \boldsymbol{\beta}^*\|_2 \le C\lambda\sqrt{s}$ and $\boldsymbol{\beta}$ is sparse $\}$. This makes the localized sparse eigenvalue condition useful, and thus makes the geometric rate of convergence possible.

Lemma 5.5 (Geometric rate in the tightening stage)**.** *Under the same conditions for Theorem 4.2 , for any* $\ell \ge 2, \left\{\boldsymbol{\beta}^{(\ell,k)}\right\}$ *converges geometrically,*

$$F(\boldsymbol{\beta}^{(\ell,k)}, \boldsymbol{\lambda}^{(\ell-1)}) - F(\widehat{\boldsymbol{\beta}}^{(\ell)}, \boldsymbol{\lambda}^{(\ell-1)}) \le \left(1 - \frac{1}{4\gamma_u\kappa}\right)^k \left\{F(\boldsymbol{\beta}^{(\ell,0)}, \boldsymbol{\lambda}^{(\ell-1)}) - F(\widehat{\boldsymbol{\beta}}^{(\ell)}, \boldsymbol{\lambda}^{(\ell-1)})\right\}$$

The above result suggests that each subproblem in the tightening stage enjoys a geometric rate of convergence, which is the fastest possible rate among all firstorder optimization methods under the blackbox model. Lemma 5.5 can be used to obtain the computational complexity analysis of each single step of the tightening stage, that is, Proposition 4.6.

6 Numerical examples

In this section, we evaluate the statistical performance of the proposed framework through several numerical experiments. We consider the following three examples.

EXAMPLE 6.1 (Linear regression)**.** *In the first example, continuous responses were generated according to the model*

$$y_i = \mathbf{x}_i^{\mathrm{T}}\boldsymbol{\beta}^* + \epsilon_i, \quad \text{where } \boldsymbol{\beta}^* = (5, 3, 0, 0, -2, \underbrace{0, \ldots, 0}_{d-5})^{\mathrm{T}}, \tag{6.1}$$

and $n = 100$. *Moreover, in model (6.1),* $\{\mathbf{x}_i\}_{i\in[n]}$ *are generated from* $N(0, \boldsymbol{\Sigma})$ *distribution with covariance matrix* $\boldsymbol{\Sigma}$, *which is independent of* $\epsilon_i \sim N(0, 1)$. *We take* $\boldsymbol{\Sigma}$ *as a correlation matrix* $\boldsymbol{\Sigma} = (\rho_{ij})$ *as follows:*

- *Case 1: independent correlation design with* $(\rho_{ij}) = \mathrm{diag}(1, \ldots, 1)$.
- *Case 2: constant correlation design with* $\rho_{ij} = 0.75$ *if* $i \ne j; \rho_{ij} = 1$, *otherwise.*
- *Case 3: autoregressive correlation design with* $\rho_{ij} = 0.95^{|i-j|}$.

EXAMPLE 6.2 (Logistic regression)**.** *In the second example, independent observations with binary responses are generated according to the model*

$$\mathbb{P}\left(y_i = 1 \mid \mathbf{x}_i\right) = \frac{\exp\left\{\mathbf{x}_i^{\mathrm{T}}\boldsymbol{\beta}^*\right\}}{1 + \exp\left\{\mathbf{x}_i^{\mathrm{T}}\boldsymbol{\beta}^*\right\}}, \quad i = 1, \ldots, n$$

where $\boldsymbol{\beta}^*$ *and* $\{\mathbf{x}_i\}_{i\in[n]}$ *are generated in the same manner as in the case 1 of Example 6.1 .*

EXAMPLE 6.3 (Varying dimensions and sample sizes)**.** *In this example, we continue Example 6.1 with varying dimensions and sample sizes. Specifically, we consider linear regression under autoregressive correlation design with* $\rho_{ij} = 0.90^{|i-j|}$ *with* d *varying from 1000 to 3500 and* n *varying from 100 to 500 .*

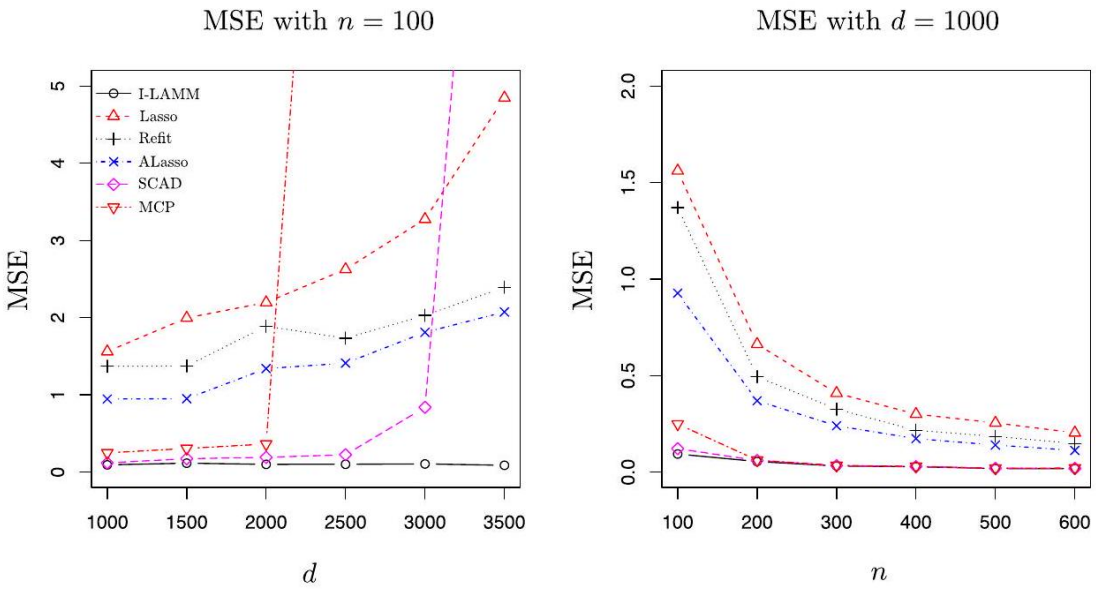

Figure 4: The median of MSE with varying dimensions and sample sizes in Example 6.3.

In the first two cases, we fix the sample size n at 100 and consider $d = 1000$. We investigate the sparsity recovery and estimation properties of the TAC estimator via numerical simulations. We compared the TAC estimator with the following methods: the oracle estimator which assumes the availability of the active set S; the refitted Lasso (Refit), which uses a post least square refit on the selected set from Lasso; the adaptive Lasso (ALasso) estimator with weight function $\mathrm{w}(\beta_j) = 1/|\beta_j|$ proposed by Zou (2006); the smoothly clipped absolute deviation (SCAD) estimator [Fan and Li (2001)] with $a = 3.7$; and the minimax concave penalty (MCP) estimator with $a = 3$ [Zhang (2010a)]. For TAC, we used the 3-fold cross-validation to select the constant $c \in 0.5 \times \{1, 2, \ldots, 20\}$ in the tuning parameter $\lambda = c\sqrt{\log d/n}$ in the contraction stage, with regularization parameters updated automatically at later steps. We further took $\gamma_u = 2, \varepsilon_c = \sqrt{\log d/n}$ and $\varepsilon_t = \sqrt{1/n}$. For the Lasso, we used the TAC algorithm; for the ALasso, sequential tuning in Bühlmann and van de Geer (2011) was used: we employed the 3-fold cross validation in each step with the TAC algorithm used; and the SCAD and MCP estimators were computed using the R package ncvreg and the 3-fold cross-validation was used for tuning parameter selection.

For each simulation setting, we generated 100 simulated datasets and applied different estimators to each dataset. We report different statistics for each estimator in Table 1 and Figure 4. To measure the sparsity recovery performance, we calculated the median of the number of zero coefficients incorrectly estimated to be nonzero (i.e., false positive, denoted as FP), the median of the number of nonzero coefficients correctly estimated to be nonzero (i.e., true positive, denoted by TP). To measure the estimation accuracy, we calculated the median of mean squared error (MSE). To evaluate the computational efficiency, we gave the median of time (in seconds) used to produce the final estimator for different methods. Note that the computational time provided here is merely for a reference. They depend on optimization errors and implementation.

We have several important observations. First, it is not surprising that Lasso tends to overfit. Other procedures improve the performance of Lasso by reducing the estimation bias and the false positive rate. The best overall performance is achieved by the TAC estimator with small MSE and FP in all cases. The MCP and SCAD estimators also have overall good performance in the

logistic regression model, and case 1 and case 2 of the linear regression model. However, all of MCP, SCAD and ALasso breaks down by missing true positives in case 3 , where the design matrix exhibits a strong correlation between features, while TAC remains the best followed by the Lasso estimator. This suggests the superiority of TAC over other implementation-based nonconvex penalized regression methods under strongly correlated designs. The MSE of the TAC estimator keeps flat when the dimension d varies, which justifies the oracle rate $\sqrt{s/n}$. SCAD and MCP have competitive performance when the dimension is relatively small, but they quickly break down when the dimension gets larger. This is possibly due to the numerical instability for directly solving nonconvex systems. This phenomenon is also observed in Wang, Liu and Zhang (2014). When the sample size is increasing, the performances of TAC, SCAD and MCP are almost identical to each other while other convex methods suffer from slightly worse performance.

In addition, to demonstrate the phase transition phenomenon, in Figure 2, we plot the log estimation error verses the number of iterations for each tightening step for case 2 in Example 6.1. Indeed, the contraction stage suffers a sublinear rate of convergence before getting into the contracting region and enjoys a geometric rate afterwards, while the tightening stage has a geometric rate of convergence. These are in line with our asymptotic theory.

7 Conclusions and Discussions

We introduce TAC (Tightening After Contraction), a computational framework that simultaneously controls both algorithmic complexity and statistical error in high-dimensional models. TAC operates in two stages: the first stage solves a convex optimization problem with coarse tolerance to produce an initial estimator, while the second stage iteratively refines this estimator by solving a sequence of convex programs with more stringent precision tolerances. This method introduces a phase transition, where the first stage exhibits sublinear iteration complexity, and the second stage improves to a linear convergence rate.

Though primarily algorithmic, TAC offers optimal statistical performance across a wide range of nonconvex optimization problems. The contraction property demonstrates how iteration reduces statistical errors. Additionally, TAC 's theoretical foundation is built on a localized sparse/restricted eigenvalue condition, allowing it to work under weaker assumptions. Notably, it requires significantly lower minimal signal strength than comparable methods. Numerical experiments confirm the theoretical results, highlighting TAC's efficiency and accuracy in handling high-dimensional data.

7.1 Discussions

Even though TAC only solves a sequence of convex programs approximately, the solution enjoys the same optimal statistical property of the unobtainable global optimum for the foldedconcave penalized regression. Our theoretical treatment relies on a novel localized analysis which avoids the parameter bound contraint, such as $\|\boldsymbol{\beta}\|_1 \leq R$, used in all other recent works. Statistically, a δ-contraction property is established: each convex program contracts the previous estimator by a δ-fraction until the optimal statistical error is reached. Computationally, a phase transition in algorithmic convergence is established. The contraction stage enjoys only a sublinear rate of convergence while the tightening stage converges geometrically fast.

Recently, Negahban et al. (2012) proposed the restricted eigenvalue condition for unified M-estimators. Loh (2017) leveraged this condition, which is more related to our localized conditions. However, there are two major differences. First, their local parameter r is fixed at a constant

independent of n, d, s, while we allow it to go to 0 as long as $r \gtrsim \sqrt{s \log d/n}$. Second, their high-dimensional regression problem relies on the ℓ_1 ball constraint $\|\boldsymbol{\beta}\|_1 \leq R$, while our newly developed localized analysis, together with the localized conditions, removes such type of constraint. In Lozano and Meinshausen (2013), the authors only consider the solutions in a local cone, which makes their analysis much simpler than ours. In this paper, we provide a stronger result: with high probability, all local solutions must fall in a local sparse (or ℓ_1) cone, and thus makes the localized eigenvalue conditions applicable.

More recently, Wang, Kim and Li (2013) proposed a two-step approach named calibrated CCCP which achieve strong oracle properties when using the Lasso estimator as initialization. Our work differs from theirs in two aspects. First, their work aims at analyzing the least square loss while our analysis handles much broader families of loss functions. Second, their procedure attains an oracle rate but requires the minimum signal strength to be in the order of $s\sqrt{\log d/n}$. Such a requirement is suboptimal. In contrast, our results requires only $\sqrt{\log d/n}$. This weakened assumption on minimum signal strength also distinguishes TAC from other convex procedures, such as least squares refit after model selection [Belloni and Chernozhukov (2013)]. In Wang, Kim and Li (2013), the authors also proposed a high-dimensional BIC criterion for variable selection and finding the oracle estimator along the solution path. We believe such a criterion can also be applied to our framework under general conditions. In further studies, Loh and Wainwright (2014, 2015) and Wang, Liu and Zhang (2014) study the theoretical properties of nonconvex penalized M-estimators. Specifically, Loh and Wainwright (2015) and Loh and Wainwright (2014) provide conditions under which all the local optima obtained by an ℓ_1-ball constrained optimization enjoys desired statistical rates. Wang, Liu and Zhang (2014) propose a path-following strategy to obtain optimal computational and statistical rates of convergence, which also relies an extra ball constraint.

Our work differs from the aforementioned literature at least in three aspects:

(1) Our theory exploits a new notion of localized analysis, which is not available in Loh and Wainwright (2014,2015) and Wang, Liu and Zhang (2014). Such analysis allows us to eliminate the extra ball constraints in previous work, which introduce more tuning effort and are intuitively redundant given the penalty function.

(2) Our statistical results tolerate explicit computational precisions and are valid for all obtained approximate solutions, while the analysis in Loh and Wainwright (2015) only targets on the exact local solutions. Moreover, our computational result does not rely on the path-following type strategy as in Wang, Liu and Zhang (2014) and is valid for any algorithm with desired statistical properties as basic building blocks within each of the tightening steps.

(3) We provide a refined oracle statistical rate $\sqrt{s/n}$ for the obtained approximation solution, while Loh and Wainwright (2015) and Wang, Liu and Zhang (2014) do not provide such a result. Loh and Wainwright (2015) provide a statistical rate which is also achievable using the convex Lasso penalty. Wang, Liu and Zhang (2014) only prove the oracle rate for exact local solutions.

Our work can be applied to many different topics: low-rank matrix completion problems, high-dimensional graphical models, quantile regression and many others. We conjecture that in all of the aforementioned topics, TAC can give a faster rate by approximately solving a sequence of convex programs, with controlled computing resources. It is also interesting to see how our algorithm works in large-scale distributed systems. ***Is there any fundamental tradeoffs between statistical efficiency, communication and time complexity?*** We leave these as future research projects.

References

[Agarwal et al.(2012)] A. Agarwal, S. Negahban, and M. J. Wainwright. Fast global convergence of gradient methods for high-dimensional statistical recovery. *Annals of Statistics*, 40:2452–2482, 2012.

[Beck and Teboulle(2009)] A. Beck and M. Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2:183–202, 2009.

[Belloni and Chernozhukov(2013)] A. Belloni and V. Chernozhukov. Least squares after model selection in high-dimensional sparse models. *Bernoulli*, 19:521–547, 2013.

[Bickel et al.(2009)] P. J. Bickel, Y. Ritov, and A. B. Tsybakov. Simultaneous analysis of lasso and Dantzig selector. *Annals of Statistics*, 37:1705–1732, 2009.

[Boyd and Vandenberghe(2004)] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[Breheny and Huang(2011)] P. Breheny and J. Huang. Coordinate descent algorithms for nonconvex penalized regression, with applications to biological feature selection. *Annals of Applied Statistics*, 5:232–253, 2011.

[Bühlmann and van de Geer(2011)] P. Bühlmann and S. van de Geer. *Statistics for High-Dimensional Data*. Springer, 2011.

[Bunea et al.(2007)] F. Bunea, A. Tsybakov, and M. Wegkamp. Sparsity oracle inequalities for the Lasso. *Electronic Journal of Statistics*, 1:169–194, 2007.

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

[Fan and Lv(2008)] J. Fan and J. Lv. Sure independence screening for ultrahigh dimensional feature space. *Journal of the Royal Statistical Society: Series B*, 70:849–911, 2008.

[Fan and Lv(2011)] J. Fan and J. Lv. Nonconcave penalized likelihood with NP-dimensionality. *IEEE Transactions on Information Theory*, 57:5467–5484, 2011.

[Fan et al.(2014)] J. Fan, L. Xue, and H. Zou. Strong oracle optimality of folded concave penalized estimation. *Annals of Statistics*, 42:819–849, 2014.

[Fan et al.(2018)] J. Fan, H. Liu, Q. Sun, and T. Zhang. Supplement to "I-LAMM for sparse learning: Simultaneous control of algorithmic complexity and statistical error". *DOI:10.1214/17AOS1568SUPP*.

[Friedman et al.(2007)] J. Friedman, T. Hastie, H. Höfling, and R. Tibshirani. Pathwise coordinate optimization. *Annals of Applied Statistics*, 1:302–332, 2007.

[Hunter and Lange(2004)] D. R. Hunter and K. Lange. A tutorial on MM algorithms. *American Statistician*, 58:30–37, 2004.

[Kim et al.(2008)] Y. Kim, H. Choi, and H.-S. Oh. Smoothly clipped absolute deviation on high dimensions. *Journal of the American Statistical Association*, 103:1665–1673, 2008.

[Kim and Kwon(2012)] Y. Kim and S. Kwon. Global optimality of nonconvex penalized estimators. *Biometrika*, 99:315–325, 2012.

[Lange et al.(2000)] K. Lange, D. R. Hunter, and I. Yang. Optimization transfer using surrogate objective functions. *Journal of Computational and Graphical Statistics*, 9:1–59, 2000.

[Loh(2017)] P.-L. Loh. Statistical consistency and asymptotic normality for high-dimensional robust M-estimators. *Annals of Statistics*, 45:866–896, 2017.

[Loh and Wainwright(2014)] P.-L. Loh and M. J. Wainwright. Support recovery without incoherence: A case for nonconvex regularization. To appear, 2014. Available at arXiv:1412.5632.

[Loh and Wainwright(2015)] P.-L. Loh and M. J. Wainwright. Regularized M-estimators with nonconvexity: Statistical and algorithmic theory for local optima. *Journal of Machine Learning Research*, 16:559–616, 2015.

[Lozano and Meinshausen(2013)] A. C. Lozano and N. Meinshausen. Minimum distance estimation for robust high-dimensional regression. Available at arXiv:1307.3227.

[Negahban et al.(2012)] S. N. Negahban, P. Ravikumar, M. J. Wainwright, and B. Yu. A unified framework for high-dimensional analysis of M-estimators with decomposable regularizers. *Statistical Science*, 27:538–557, 2012.

[Nesterov(2013)] Y. Nesterov. Gradient methods for minimizing composite functions. *Mathematical Programming*, 140:125–161, 2013.

[Raskutti et al.(2010)] G. Raskutti, M. J. Wainwright, and B. Yu. Restricted eigenvalue properties for correlated Gaussian designs. *Journal of Machine Learning Research*, 11:2241–2259, 2010.

[Rudelson and Vershynin(2013)] M. Rudelson and R. Vershynin. Hanson-Wright inequality and subgaussian concentration. *arXiv preprint arXiv:1306.2872*, 2013.

[Tibshirani(1996)] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B*, 58:267–288, 1996.

[Van de Geer and Bühlmann(2009)] S. A. Van de Geer and P. Bühlmann. On the conditions used to prove oracle results for the Lasso. *Electronic Journal of Statistics*, 3:1360–1392, 2009.

[Wang et al.(2013)] L. Wang, Y. Kim, and R. Li. Calibrating nonconvex penalized regression in ultra-high dimension. *Annals of Statistics*, 41:2505–2536, 2013.

[Wang et al.(2014)] Wang, Z., Liu, H., and Zhang, T. (2014). Optimal computational and statistical rates of convergence for sparse nonconvex learning problems. *Annals of Statistics*, 42, 2164–2201.

[Zhang(2009)] T. Zhang. Some sharp performance bounds for least squares regression with L_1 regularization. *Annals of Statistics*, 37:2109–2144, 2009.

[Zhang(2010a)] C.-H. Zhang. Nearly unbiased variable selection under minimax concave penalty. *Annals of Statistics*, 38:894–942, 2010.

[Zhang(2010b)] T. Zhang. Analysis of multi-stage convex relaxation for sparse regularization. *Journal of Machine Learning Research*, 11:1081–1107, 2010.

[Zhang and Zhang(2012)] C.-H. Zhang and T. Zhang. A general theory of concave regularization for high-dimensional sparse estimation problems. *Statistical Science*, 27:576–593, 2012.

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

[Zou and Li(2008)] H. Zou and R. Li. One-step sparse estimates in nonconcave penalized likelihood models. *Annals of Statistics*, 36:1509–1533, 2008.