East Asian Journal on Applied Mathematics
doi: 10.4208/eajam.110121.210621

Vol. **12**, No. 1, pp. 35-52
February 2022

Hard Thresholding Regularised Logistic Regression: Theory and Algorithms

Lican Kang[1], Yanyan Liu[1], Yuan Luo[1] and Chang Zhu[2,*]

[1]*School of Mathematics and Statistics, Wuhan University, Wuhan 430072, China.*
[2]*Department of Anesthesiology, Tongji Hospital, Tongji Medical College, Huazhong University of Science and Technology, Wuhan 430030, China.*

Received 11 January 2021; Accepted (in revised version) 21 June 2021.

Abstract. The hard thresholding regularised logistic regression in high dimensions with larger number of features than samples is considered. The sharp oracle inequality for the global solution is established. If the target signal is detectable, it is proven that with a high probability the estimated and true supports coincide. Starting with the KKT condition, we introduce the primal and dual active sets algorithm for fitting and also consider a sequential version of this algorithm with a warm-start strategy. Simulations and a real data analysis show that SPDAS outperforms LASSO, MCP and SCAD methods in terms of computational efficiency, estimation accuracy, support recovery and classification.

AMS subject classifications: 62J12, 62J02, 62J07

Key words: Sparse logistic regression, hard thresholding regularisation, PDAS, SPDAS.

1. Introduction

Let $y \in \{0,1\}$ be the binary response variable, $\mathbf{x} \in \mathbb{R}^p$ the covariate vector and $\boldsymbol{\beta}^* \in \mathbb{R}^p$ the underlying regression coefficients vector in the logistic regression model

$$P\big(y=1|\mathbf{x}\big) = \frac{\exp\big(\mathbf{x}^T\boldsymbol{\beta}^*\big)}{1+\exp\big(\mathbf{x}^T\boldsymbol{\beta}^*\big)},$$

cf. [10, 11]. Logistic regression is an important generalised linear model (GLM) widely used in statistics, machine learning, social and medical sciences, finance industry and so on. In this work, we focus on the variable estimation and selection in high-dimensional and sparse settings — i.e. if $n \ll p$ and $\|\boldsymbol{\beta}^*\|_0 < n$, where n is the sample size and $\|\boldsymbol{\beta}^*\|_0$ the cardinality of the set of nonzero elements in $\boldsymbol{\beta}^*$.

*Corresponding author. *Email addresses:* kanglican@whu.edu.cn (L. Kang), liuyy@whu.edu.cn (Y. Liu), yuanluo@whu.edu.cn (Y. Luo), changzhu@hust.edu.cn (C. Zhu)

 ©2022 Global-Science Press

To obtain the estimator of $\boldsymbol{\beta}^*$ in high-dimensional and sparse cases, many regularised methods have been proposed. In particular, works [13,17] extend the least absolute shrinkage and selection operator method (LASSO) [16] from the linear regression to GLMs. Friedman *et al.* [4] used the coordinate descent to solve the elastic net penalised GLMs [21]. In Refs. [8, 19], the path following proximal gradient descent method [12] has been applied to variable estimation in GLMs with smoothly clipped absolute deviation (SCAD) and minimax concave (MC) penalties [3]. Besides, Li *et al.* [7] introduced a DC proximal Newton (DCPN) method for GLMs with sparsity promoting non-convex penalties such as MC and SCAD ones.

In this paper, we consider the hard thresholding regulariser

$$\rho_\lambda(t) = \begin{cases} \dfrac{-t^2}{2} + \lambda|t|, & \text{if} \quad |t| < \lambda, \\ \dfrac{\lambda^2}{2}, & \text{if} \quad |t| \geq \lambda, \end{cases} \tag{1.1}$$

where a non-convex and non-smooth function ρ_λ admits the hard thresholding operator (3.2), cf. [1]. The hard thresholding regularised estimator leads to the problem

$$\min_{\boldsymbol{\beta}} \mathscr{L}_n(\boldsymbol{\beta}) + \sum_{i=1}^{p} \rho_\lambda(\beta_i), \tag{1.2}$$

where

$$\mathscr{L}_n(\boldsymbol{\beta}) = \frac{1}{n}\sum_{i=1}^{n} \log\left(1 + \exp\left(\mathbf{x}_i^T \boldsymbol{\beta}\right)\right) - \frac{\mathbf{Y}^T \mathbf{X}\boldsymbol{\beta}}{n}$$

is the negative logarithmic likelihood function and $\lambda > 0$ the tuning parameter.

The ideas of [8, 20] can be used to determine the sharp upper bound for the errors of the estimator (1.2). Nevertheless, since (1.2) is a non-convex non-smooth optimisation problem, it is difficult to develop a stable efficient computational algorithm for its solution, especially in high-dimensional and sparse settings. Inspired by [5, 15], we construct a primal and dual active set (PDAS) algorithm for solving the minimisation problem (1.2). Our approach is motivated by the KKT conditions of the hard thresholding regularised problem.

In PDAS, the active set of a relatively small size is first determined via summation of primal and dual variables generated by the previous iteration. The primal variable is then updated by solving a minimisation problem on the active set, whereas the dual variable is updated by using the gradient information. Further, in order to make PDAS more applicable, we consider a sequential version of PDAS (SPDAS), which combines PDAS with a continuation strategy on the regularisation parameter λ. Thus, by SPDAS algorithm we generate a solution path with a different regularisation parameter λ. Then we can chose one data-driven method such as the modified Bayesian information criteria [6, 18] or the voting method [5] to chose the optimal solution.

The main results of this work are as follows. Using regularity assumptions on the loss function and the covariance matrix, we establish a sharp oracle inequality of the global

solution and prove that for detectable target signals the estimated and true supports coincide with a high probability. After that, we exploit the KKT conditions of the minimiser and construct a primal and dual active sets algorithm (PDAS) for fitting. In PDAS, active sets with small size are identified iteratively via the primal and dual variables in the previous iteration, and the primal variable is updated by the maximum likelihood estimation restricted to the active set. The dual variable is updated explicitly with the gradient information. Furthermore, we consider a sequential PDAS (SPDAS) with a warm-start strategy to provide good initial values for PDAS automatically. Extensive numerical simulations and real data analysis demonstrate the superiority of the method over LASSO, MCP and SCAD in terms of the estimation accuracy, support recovery, computational speed and prediction accuracy in classification.

The rest of this paper is organised as follows. In Section 2, we present the theoretical analysis for the global solution. Under certain conditions we establish non-asymptotic ℓ_1 and ℓ_2-norm error bounds for the global solution and show that its support set coincides with the target support set with a high probability. The PDAS and SPDAS algorithms are introduced in Section 3. Section 4 deals with the simulations aimed to evaluate the performance of SPDAS and illustrate its application. Our conclusion is in Section 5 and the proofs of Theorem 2.1 and Lemma 3.1 are moved to Appendix A.

2. Theoretical Properties of Global Solutions

We first introduce the notations used in this paper. Let $\|\boldsymbol{\beta}\|_q$, $q \in [1,\infty]$ be the usual q-norm on $\mathbb{R}^p$, i.e.

$$\|\boldsymbol{\beta}\|_q := \left(\sum_{i=1}^{p} |\beta_i|^q\right)^{\frac{1}{q}}, \quad \boldsymbol{\beta} = (\beta_1,\ldots,\beta_p)^T \in \mathbb{R}^p.$$

Set

$$\rho_\lambda(\boldsymbol{\beta}) = \sum_{i=1}^{p} \rho_\lambda(\beta_i),$$

and let $\|\boldsymbol{\beta}\|_{\min}$ be the minimal absolute value of $\boldsymbol{\beta}$. Consider the set $S = \{1,\ldots,p\}$ and for any subset $A \subseteq S$ of the size $|A|$ let $\boldsymbol{\beta}_A$ ($\mathbf{X}_A \in \mathbb{R}^{n\times|A|}$) refer to a subvector (a submatrix) whose entries (columns) are listed in A. If $\mathbf{X}$ is covariance matrix, then $\mathbf{X}_{AB}$ denotes a submatrix of $\mathbf{X}$ whose rows and columns are respectively listed in A and B. Besides, the support $\{i \in S : z_i \neq 0\}$ of the vector $\mathbf{z}$ is denoted by $\mathrm{supp}(\mathbf{z})$ and $A^* := \mathrm{supp}(\boldsymbol{\beta}^*)$ and $I^* := (A^*)^c$.

The hard thresholding ρ_λ in (1.1) satisfies [8, Assumption 1] and general assumptions of [20], where a class of regularisation functions is studied. Thus if λ is fixed, then ρ_λ is a one-symmetric function about the ordinate axis, which vanishes at $t = 0$. Moreover, it is a subadditive nondecreasing function on $(0,\infty)$, differentiable for all $t \neq 0$ and such that $\lim_{t\to 0^+} \rho_\lambda(t)' = \lambda$. It indicates that ρ_λ is λ-Lipschitz continuous. Moreover, $\rho_{\lambda,\mu}(t) = \rho_\lambda(t)+\mu t^2/2$ is convex for any $\mu \geq 1$ [8,20]. Therefore, we can use [8] in order to derive an oracle nonasymptotic error bound for the global solution and to study the support recovery property.

Following [8], we also consider the feasible set $\Omega = \{\boldsymbol{\beta} : g(\boldsymbol{\beta}) < R\}$, where

$$g(\boldsymbol{\beta}) = \frac{\rho_\lambda(\boldsymbol{\beta}) + (\mu/2)\|\boldsymbol{\beta}\|^2}{\lambda},$$

R is a positive constant such that $\boldsymbol{\beta}^* \in \Omega$ and $\mu \geq 1$. Let $\boldsymbol{\beta}^\diamond$ denote the global solution of the optimisation problem (1.2) restricted to Ω. According to [8], the restricted strong convexity (RSC) condition has the form

$$\left\langle \nabla \mathscr{L}_n(\boldsymbol{\beta}^* + \boldsymbol{\Delta}) - \nabla \mathscr{L}_n(\boldsymbol{\beta}^*), \boldsymbol{\Delta} \right\rangle \geq \begin{cases} \alpha_1 \|\boldsymbol{\Delta}\|_2^2 - \tau_1 \dfrac{\log p}{n} \|\boldsymbol{\Delta}\|_1^2, & \forall\ \|\boldsymbol{\Delta}\|_2 \leq 1, \\ \alpha_2 \|\boldsymbol{\Delta}\|_2 - \tau_2 \dfrac{\log p}{n} \|\boldsymbol{\Delta}\|_1, & \forall\ \|\boldsymbol{\Delta}\|_2 \geq 1, \end{cases} \tag{2.1}$$

where α_1, α_2 are strictly positive constants and τ_1, τ_2 nonnegative constants. This RSC inequality implies that the set $\mathscr{L}_n(\boldsymbol{\beta})$ is strong convex over the cone of the form

$$\left\{ \frac{\|\Delta\|_1}{\|\Delta\|_2} \leq c \sqrt{\frac{n}{\log p}} \right\}.$$

Theorem 2.1. *Let $\mathscr{L}_n(\boldsymbol{\beta})$ satisfy the RSC condition* (2.1) *with $3\mu/4 < \alpha_1$ and $\mu \geq 1$. If λ satisfies the inequality*

$$4 \max \left\{ \|\nabla \mathscr{L}_n(\boldsymbol{\beta}^*)\|_\infty, \alpha_2 \sqrt{\log(p)/n} \right\} \leq \lambda \leq \frac{\alpha_2}{6R}$$

and

$$n \geq \left(16R^2 \max\left(\tau_1^2, \tau_2^2\right)/\alpha_2^2\right) \log(p),$$

then

$$\|\boldsymbol{\beta}^\diamond - \boldsymbol{\beta}^*\|_1 \leq \frac{24\lambda|A^*|}{4\alpha_1 - 3\mu}, \quad \|\boldsymbol{\beta}^\diamond - \boldsymbol{\beta}^*\|_2 \leq \frac{6\lambda\sqrt{|A^*|}}{4\alpha_1 - 3\mu}.$$

Moreover, if the entries of $\mathbf{X}$ are i.i.d. sub-Gaussian, then there exist universal constants $\{c_1, c_2, c_3\}$ with $0 < c_i < \infty$, $i = 1, 2, 3$, such that

$$\|\boldsymbol{\beta}^\diamond - \boldsymbol{\beta}^*\|_1 \leq \frac{96|A^*|(c_1 + \alpha_2)\sqrt{\log(p)/n}}{4\alpha_1 - 3\mu},$$

$$\|\boldsymbol{\beta}^\diamond - \boldsymbol{\beta}^*\|_2 \leq \frac{24(c_1 + \alpha_2)\sqrt{|A^*|\log(p)/n}}{4\alpha_1 - 3\mu}$$

with probability at least $1 - c_2 \exp(-c_3 \log(p))$.

Our next goal is to study the support recovery property of the minimiser $\boldsymbol{\beta}^\diamond$. To do this, we need a condition that would guaranty the detectability of the signal.

Condition 2.1. The term $\|\boldsymbol{\beta}^*_{A^*}\|_{\min}$ satisfies the inequality

$$\|\boldsymbol{\beta}^*_{A^*}\|_{\min} > \frac{96|A^*|(c_1 + \alpha_2)\sqrt{\log(p)/n}}{4\alpha_1 - 3\mu},$$

where μ, c_1, α_2 are defined in Theorem 2.1.

Theorem 2.2. *If the conditions of Theorem* 2.1 *and Condition* 2.1 *hold, then*

$$A^* \subseteq supp(\boldsymbol{\beta}^\diamond)$$

with the probability at least $1 - c_2 \exp(-c_3 \log(p))$.

Proof. The Condition 2.1 shows that

$$\|\boldsymbol{\beta}^* - \boldsymbol{\beta}^\diamond\|_\infty < \|\boldsymbol{\beta}^*_{A^*}\|_{\min}.$$

It implies that $A^* \subseteq \operatorname{supp}(\boldsymbol{\beta}^\diamond)$. □

3. PDAS and SPDAS

According to the above analysis, the global solution $\boldsymbol{\beta}^\diamond$ is the oracle estimator of the target regression coefficients $\boldsymbol{\beta}^*$. But (1.2) is a non-convex non-smooth optimisation problem. This creates various difficulties in construction of iterative computational algorithms in finding this oracle estimator. Following the ideas of [5, 15], we develop a primal and dual active sets algorithm (PDAS) for computations. After that we introduce the sequential version of PDAS algorithm with a warm-start strategy.

3.1. PDAS Algorithm

Based on the KKT condition of the hard thresholding regularised problem (1.2), we can determine a minimiser on $\mathbb{R}^p$ as the following lemma shows.

Lemma 3.1. *If* $\boldsymbol{\beta}^\diamond$ *is the minimiser of the problem* (1.2)*, then* $\boldsymbol{\beta}^\diamond$ *satisfies the equations*

$$\begin{aligned} \mathbf{d}^\diamond &= -\nabla \mathscr{L}_n(\boldsymbol{\beta}), \\ \boldsymbol{\beta}^\diamond &= \Gamma_\lambda(\boldsymbol{\beta}^\diamond + \mathbf{d}^\diamond), \end{aligned} \tag{3.1}$$

where the i*-th element of* Γ_λ *is defined by*

$$\big(\Gamma_\lambda(\boldsymbol{\beta})\big)_i = \begin{cases} 0, & |\beta_i| \le \lambda, \\ \beta_i, & |\beta_i| > \lambda. \end{cases} \tag{3.2}$$

Conversely, if $\boldsymbol{\beta}^\diamond$ *and* $\mathbf{d}^\diamond$ *satisfy the Eqs.* (3.1)*, then* $\boldsymbol{\beta}^\diamond$ *is a stationary point of* (1.2).

Lemma 3.1 provides an implicit expression of the minimiser of $\boldsymbol{\beta}^\diamond$ and it is a base for the PDAS algorithm. Note that Γ_λ in (3.2) is the hard thresholding operator corresponding to the regularisation ρ_λ. Writing $A^\diamond = \operatorname{supp}(\boldsymbol{\beta}^\diamond)$, $I^\diamond = (A^\diamond)^c$ and using the definition of Γ_λ and the Eqs. (3.1) yields

$$A^\diamond = \left\{ i \in S : \left|\beta_i^\diamond + d_i^\diamond\right| > \lambda \right\}, \quad I^\diamond = \left\{ i \in S : \left|\beta_i^\diamond + d_i^\diamond\right| \le \lambda \right\},$$

and

$$\boldsymbol{\beta}^{\diamond}_{I^{\diamond}} = \mathbf{0}, \quad \mathbf{d}^{\diamond}_{A^{\diamond}} = \mathbf{0}, \quad \boldsymbol{\beta}^{\diamond}_{A^{\diamond}} \in \underset{\boldsymbol{\beta}_{A^{\diamond}}}{\operatorname{argmin}} \widetilde{\mathscr{L}_n}(\boldsymbol{\beta}_{A^{\diamond}}), \quad \mathbf{d}^{\diamond}_{I^{\diamond}} = \big[-\nabla \mathscr{L}_n(\boldsymbol{\beta}^{\diamond})\big]_{I^{\diamond}},$$

where

$$\widetilde{\mathscr{L}_n}(\boldsymbol{\beta}_{A^{\diamond}}) = \mathscr{L}_n(\boldsymbol{\beta}|_{A^{\diamond}}) = -\frac{1}{n}\sum_{i=1}^{n}\Big[y_i \mathbf{x}^T_{i(A^{\diamond})}\boldsymbol{\beta}_{A^{\diamond}} - \log\Big(1+\exp\big(\mathbf{x}^T_{i(A^{\diamond})}\boldsymbol{\beta}_{A^{\diamond}}\big)\Big)\Big].$$

For a fixed λ, let $\{\boldsymbol{\beta}^k, \mathbf{d}^k\}$ be the values in the k-th iteration in PDAS algorithm. We denote by $\{A^k, I^k\}$ the active and inactive sets corresponding to $\{\boldsymbol{\beta}^k, \mathbf{d}^k\}$. More precisely, we have

$$A^k = \Big\{i \in S : \big|\beta^k_i + d^k_i\big| > \lambda\Big\}, \quad I^k = \Big\{i \in S : \big|\beta^k_i + d^k_i\big| \le \lambda\Big\}.$$

This leads to the new approximation pair $\{\boldsymbol{\beta}^{k+1}_{I^k}, \mathbf{d}^{k+1}_{A^k}, \boldsymbol{\beta}^{k+1}_{A^k}, \mathbf{d}^{k+1}_{I^k}\}$ with the terms

$$\boldsymbol{\beta}^{k+1}_{I^k} = \mathbf{0}, \quad \mathbf{d}^{k+1}_{A^k} = \mathbf{0}, \quad \boldsymbol{\beta}^{k+1}_{A^k} = \underset{\boldsymbol{\beta}_{A^k}}{\operatorname{argmin}} \widetilde{\mathscr{L}_n}(\boldsymbol{\beta}_{A^k}), \quad \mathbf{d}^{k+1}_{I^k} = \big[-\nabla \mathscr{L}_n\big(\boldsymbol{\beta}^{k+1}\big)\big]_{I^k},$$

where

$$\widetilde{\mathscr{L}_n}(\boldsymbol{\beta}_{A^k}) = -\frac{1}{n}\sum_{i=1}^{n}\Big[y_i \mathbf{x}^T_{i(A^k)}\boldsymbol{\beta}_{A^k} - \log\Big(1+\exp\big(\mathbf{x}^T_{i(A^k)}\boldsymbol{\beta}_{A^k}\big)\Big)\Big].$$

The proposed PDAS algorithm is formulated as Algorithm 3.1.

Remark 3.1. Algorithm 3.1 terminates computation when the sequential estimated support coincides with each other or the maximum iteration number exceeds a given number K. The step 3 of the algorithm distinguishes the active set by combining primal and dual variables. The minimisation problem restricted to the selected active set A^k is then solved as described in step 6.

Algorithm 3.1 PDAS algorithm

1: Input: $\boldsymbol{\beta}^0$, $\mathbf{d}^0$, λ, $k = 0$, K.
2: **for** $k = 0, 1, \ldots, K$, **do**
3: $A^k = \big\{j \in S : \big|\beta^k_j + d^k_j\big| > \lambda\big\}$, $I^k = (A^k)^c$;
4: $\boldsymbol{\beta}^{k+1}_{I^k} = \mathbf{0}$;
5: $\mathbf{d}^{k+1}_{A^k} = \mathbf{0}$;
6: $\boldsymbol{\beta}^{k+1}_{A^k} = \underset{\boldsymbol{\beta}_{A^k}}{\operatorname{argmin}} \widetilde{\mathscr{L}_n}(\boldsymbol{\beta}_{A^k})$;
7: $\mathbf{d}^{k+1}_{I^k} = \big[-\nabla \mathscr{L}_n(\boldsymbol{\beta}^{k+1})\big]_{I^k}$;
8: **if** $A^k = A^{k+1}$ or $k \ge K$ **then**
9: Stop and denote the last iteration $\boldsymbol{\beta}_{\hat{A}}$, $\boldsymbol{\beta}_{\hat{I}}$, $\mathbf{d}_{\hat{A}}$, $\mathbf{d}_{\hat{I}}$;
10: **else**
11: $k = k + 1$;
12: **end if**
13: **end for**
14: Output: $\hat{\boldsymbol{\beta}}(\lambda) = (\boldsymbol{\beta}^{\mathrm{T}}_{\hat{A}}, \boldsymbol{\beta}^{\mathrm{T}}_{\hat{I}})^{\mathrm{T}}$ and $\hat{\mathbf{d}}(\lambda) = (\mathbf{d}^{\mathrm{T}}_{\hat{A}}, \mathbf{d}^{\mathrm{T}}_{\hat{I}})^{\mathrm{T}}$ as the estimators at λ.

3.2. SPDAS algorithm

PDAS algorithm (Algorithm 3.1) only solves the minimisation problem (1.2) with fixed tuning parameter λ. However, we are more interested in the solution path. Here, we propose a sequential PDAS algorithm (SPDAS), which combines PDAS and a continuation strategy thus providing good initial guesses and determines a solution path. Consider the decreasing sequence of regularisation parameters $\lambda_m = \lambda_0 \alpha^m$, $\alpha \in (0,1)$.

According to Lemma 3.1, the vector $\mathbf{0}$ is the minimiser of the problem (1.2) if $\lambda \geq \|\nabla \mathscr{L}_n(\mathbf{0})\|_\infty$. Therefore, we set $\lambda_0 = \|\nabla \mathscr{L}_n(\mathbf{0})\|_\infty$ so that

$$\hat{\boldsymbol{\beta}}(\lambda_0) = \mathbf{0} \quad \text{and} \quad \hat{\mathbf{d}}(\lambda_0) = -\nabla \mathscr{L}_n(\mathbf{0}).$$

Then we apply Algorithm 3.1 to the sequence $\{\lambda_m\}_m$ with the solution $\{\hat{\boldsymbol{\beta}}(\lambda_m), \hat{\mathbf{d}}(\lambda_m)\}$ as the initial guess for the λ_{m+1}-problem. We can terminate the SPDAS algorithm and obtain the solution path up to

$$\left\|\hat{\boldsymbol{\beta}}(\lambda_m)\right\|_0 > \left\lfloor \frac{n}{\log p} \right\rfloor$$

for an m. After that, we can employ a data-driven method — e.g. the modified Bayesian information criteria [6,18] or the voting method [5] to choose the optimal solution without any extra computational overhead. The pseudocode of SPDAS algorithm is described by Algorithm 3.2.

Algorithm 3.2 SPDAS algorithm

Input: $\hat{\boldsymbol{\beta}}(\lambda_0) = \mathbf{0}$, $\hat{\mathbf{d}}(\lambda_0) = -\nabla \mathscr{L}_n(\mathbf{0})$, $\lambda_0 = \|\nabla \mathscr{L}_n(\mathbf{0})\|_\infty$, M, α.
for $m = 1, \ldots, M$ **do**
  $\lambda = \lambda_m = \lambda_0 \alpha^m$, $\boldsymbol{\beta}^0 = \hat{\boldsymbol{\beta}}(\lambda_{m-1})$, $\mathbf{d}^0 = \hat{\mathbf{d}}(\lambda_{m-1})$;
  Run Algorithm 3.1 to get $\hat{\boldsymbol{\beta}}(\lambda_m)$ and $\hat{\mathbf{d}}(\lambda_m)$;
  if $\|\hat{\boldsymbol{\beta}}(\lambda_m)\|_0 > \lfloor \frac{n}{\log p} \rfloor$ **then**
    Stop;
  end if
end for
Output: $\left\{\hat{\boldsymbol{\beta}}(\lambda_0), \hat{\boldsymbol{\beta}}(\lambda_1), \ldots, \hat{\boldsymbol{\beta}}(\lambda_M)\right\}$.

4. Simulation Studies

Here, we carry out simulations to illustrate the effectiveness of the SPDAS algorithm. We also use four real data sets to compare this algorithm with LASSO, MCP and SCAD. Note that LASSO, MCP and SCAD are implemented in R package `ncvreg` [2]. In all experiments, the $n \times p$ covariates matrix $\mathbf{X}$ is generated according to the following procedure.

(I) We first generate an $n \times p$ random Gaussian matrix $\widetilde{\mathbf{X}}$ whose entries are i.i.d. $\sim N(0,1)$. Then the covariates matrix $\mathbf{X}$ is generated with $\mathbf{x}_1 = \widetilde{\mathbf{x}}_1$, $\mathbf{x}_p = \widetilde{\mathbf{x}}_p$, and $\mathbf{x}_j = \widetilde{\mathbf{x}}_j + \rho(\widetilde{\mathbf{x}}_{j+1} + \widetilde{\mathbf{x}}_{j-1})$, $j = 2, \ldots, p-1$, where ρ is the measure of the correlation between the covariates.

(II) The rows of $\mathbf{X}$ are i.i.d $\sim N(0,\Sigma)$, where

$$\Sigma_{i,j}=\rho^{|i-j|},\quad 1\le i,j\le p,$$

and ρ is the correlation parameter.

The support A^* is chosen uniformly from S with $|A^*|=T<n$. The nonzero entries are generated as $\beta_i^*=\theta_i R^{\kappa_i}$, $i\in A^*$, where θ_i are i.i.d. Bernoulli random variables with the parameter 0.5, κ_i i.i.d. uniform random variables in $[0,1]$, and $R>1$. Then the response y_i is equivalent to Binomial$(1,p_i)$ for

$$p_i=\frac{\exp\left(\mathbf{x}_i^T\boldsymbol{\beta}^*\right)}{1+\exp\left(\mathbf{x}_i^T\boldsymbol{\beta}^*\right)},\quad i=1,\dots,n.$$

4.1. Accuracy and efficiency

We now randomly choose 80% of the samples as the training set and the rest as the test set in calculating the classification accuracy rate for predicting. Then we compare SPDAS with LASSO, MCP and SCAD in terms of the average ℓ_2 relative error (RE), the average CPU time in seconds (Time), and the average classification accuracy rate by prediction (ACRP). Besides, we compare the performance of the support recovery for all four methods, evaluating the mean size of the estimated support (MSES), the average positive discovery rate (APDR), the average false discovery rate (AFDR) and the average combined discovery rate (ADR) [9]. Let J denote the number of independent replications. Then above criteria are defined

$$\text{RE}:=\frac{1}{J}\sum_{j=1}^{J}\frac{\left\|\hat{\boldsymbol{\beta}}^{(j)}-\boldsymbol{\beta}^*\right\|}{\|\boldsymbol{\beta}^*\|},\qquad \text{MSES}:=\frac{1}{J}\sum_{j=1}^{J}|\hat{A}^{(j)}|,$$

$$\text{APDR}:=\frac{1}{J}\sum_{j=1}^{J}\frac{\left|\hat{A}^{(j)}\bigcap A^*\right|}{|A^*|},\qquad \text{AFDR}:=\frac{1}{J}\sum_{j=1}^{J}\frac{\left|\hat{A}^{(j)}\bigcap A^{*c}\right|}{|\hat{A}^{(j)}|},$$

$$\text{ADR}:=\text{APDR}+(1-\text{AFDR}),\qquad \text{Time}:=\frac{1}{J}\sum_{j=1}^{J}t^{(j)},$$

where $\hat{\boldsymbol{\beta}}^{(j)}$ is the estimator at j-th simulation, $\hat{A}^{(j)}$ the estimated support, and $t^{(j)}$ the j-th running time. We observe that for small RE the corresponding method performs well in variable estimation. If APDR and ADR are close to 1 and 2, respectively, AFDR close to 0, and MSES takes values approximating the target sparse levels, then the target support can be estimated. The smaller values the Time takes, the faster computational speed the associated methods have. Meanwhile, high values of ACRP yield excellent prediction results. Let $\mathbf{X}$ be the matrix obtained according to method (I) and let $n=1000$, $p=10000$, $T=20$, $R=10$, $\rho=0.2:0.2:0.8$.

The results presented in Table 1 are based on 100 independent replications. Note that SPDAS has a better relative error and it is about 2-5 times faster than LASSO, MCP and

Table 1: Numerical results. $\mathbf{X}$ obtained by (I), $n = 1000$, $p = 10000$, $T = 20$, $R = 10$, $\rho = 0.2 : 0.2 : 0.8$.

ρ	Method	RE	Time(s)	APDR	AFDR	DR	MSES	ACRP
0.2	LASSO	0.86	35.07	0.90	0.93	0.97	236.26	89.64%
	MCP	0.43	71.67	0.91	0.33	1.58	28.01	94.69%
	SCAD	0.43	50.96	0.94	0.64	1.30	55.82	94.33%
	SPDAS	0.27	14.28	0.87	0.04	1.83	18.21	95.00%
0.4	LASSO	0.87	35.60	0.90	0.92	0.98	228.84	90.21%
	MCP	0.47	75.55	0.92	0.30	1.62	26.87	94.84%
	SCAD	0.45	52.62	0.94	0.63	1.31	53.54	94.56%
	SPDAS	0.28	13.67	0.86	0.04	1.82	17.85	94.91%
0.6	LASSO	0.88	37.78	0.90	0.92	0.98	226.99	90.58%
	MCP	0.47	74.16	0.91	0.28	1.63	25.84	95.40%
	SCAD	0.46	56.77	0.94	0.58	1.36	48.14	95.28%
	SPDAS	0.38	20.57	0.85	0.05	1.80	17.63	94.92%
0.8	LASSO	0.90	37.77	0.89	0.92	0.97	224.73	90.32%
	MCP	0.51	66.64	0.90	0.26	1.64	25.04	95.50%
	SCAD	0.52	53.67	0.93	0.58	1.35	48.43	95.24%
	SPDAS	0.50	15.23	0.84	0.05	1.79	17.32	94.85%

SCAD. In term of the support recovery, SPDAS is similar to APDR, and takes lowest values on AFDR and highest values on DR. Moreover, SPDAS takes values closest to the target sparse level on MSES. It means that SPDAS can avoid selecting the erroneous variable while choosing as many relevant variables as possible into the model. Besides, SPDAS has the higher classification accuracy rate for $\rho \leq 0.4$, but for $\rho > 0.4$ MCP and SCAD are slightly better. In summary, SPDAS performs better or is comparable with LASSO, MCP and SCAD in estimation errors, computational speed, support recovery and classification accuracy.

4.2. Influence of the model parameters

In this subsection, we consider the influence of the model parameters such as sample size n, ambient dimension p and correlation ρ on the performance of SPDAS and other methods on the computational speed and on the support recovery terms APDR, AFDR, ADR and MSES. Let $\mathbf{X}$ be the matrix generated according to method (II). The sample size n, the covariates dimension p, the correlation ρ and others are set as following:

- $n = 200 : 50 : 500$, $p = 600$, $T = 10$, $R = 5$, $\rho = 0.5$.
- $n = 200$, $p = 500 : 200 : 1700$, $T = 10$, $R = 5$, $\rho = 0.5$.
- $n = 200$, $p = 600$, $T = 10$, $R = 5$, $\rho = 0.1 : 0.1 : 0.9$.

The respective results are presented in Tables 2-4. Note that SPDAS is fastest of the methods considered. On support recovery, SPDAS takes values comparable to others on APDR, and

Table 2: Numerical results (APDR, AFDR, ADR), $p = 600$, $T = 10$, $R = 5$, $\rho = 0.5$, $n = 200 : 50 : 500$.

n	Method	Times(s)	APDR	AFDR	ADR	MSES
200	LASSO	2.18	0.91	0.83	1.08	57.61
	MCP	10.47	0.83	0.35	1.48	13.08
	SCAD	18.38	0.88	0.61	1.27	23.5
	SPDAS	0.85	0.79	0.22	1.57	10.29
250	LASSO	2.82	0.96	0.85	1.11	67.99
	MCP	13.46	0.90	0.32	1.58	13.59
	SCAD	21.48	0.93	0.58	1.35	23.57
	SPDAS	1.02	0.86	0.19	1.67	11.12
300	LASSO	3.39	0.98	0.86	1.12	73.8
	MCP	18.07	0.93	0.29	1.64	13.52
	SCAD	24.69	0.96	0.58	1.38	24.26
	SPDAS	1.08	0.93	0.13	1.80	10.96
350	LASSO	3.75	0.99	0.87	1.12	77.25
	MCP	22.05	0.96	0.28	1.68	14.05
	SCAD	21.83	0.98	0.57	1.41	24.41
	SPDAS	1.19	0.96	0.10	1.86	10.82
400	LASSO	4.05	0.99	0.87	1.12	79.5
	MCP	25.94	0.98	0.25	1.73	13.59
	SCAD	19.45	0.99	0.56	1.43	24.20
	SPDAS	1.14	0.96	0.08	1.88	10.61
450	LASSO	4.47	0.99	0.87	1.12	80.61
	MCP	32.29	0.99	0.26	1.74	13.95
	SCAD	18.26	0.99	0.55	1.44	24.03
	SPDAS	1.39	0.98	0.07	1.91	10.68
500	LASSO	4.88	0.99	0.88	1.11	83.83
	MCP	32.23	0.99	0.24	1.75	13.5
	SCAD	15.78	0.99	0.55	1.44	23.85
	SPDAS	1.44	0.99	0.05	1.94	10.55

takes lowest and highest values on APDR and ADR respectively, and SPDAS takes values about 10 on MSES for all settings considered here. Overall, SPDAS can simultaneously selects relevant variables and avoid the irrelevant variables for a wide spectrum of the values of n, p, ρ.

4.3. Real data analysis

We apply SPDAS to four data sets — viz. duke breast-cancer, gisette, leukemia and splice described in Table 5. These data sets are available at `https://www.csie.ntu.edu.tw/`

Table 3: Numerical results (APDR, AFDR, ADR), $n = 200$, $T = 10$, $R = 5$, $\rho = 0.5$, $p = 500 : 200 : 1700$.

p	Method	Times(s)	APDR	AFDR	ADR	MSES
500	LASSO	2.01	0.93	0.82	1.11	55.09
	MCP	10.03	0.86	0.33	1.53	13.23
	SCAD	16.70	0.90	0.59	1.31	23.11
	SPDAS	0.61	0.82	0.23	1.59	10.92
700	LASSO	2.24	0.91	0.84	1.07	58.51
	MCP	10.31	0.83	0.34	1.49	12.96
	SCAD	17.46	0.88	0.61	1.27	23.84
	SPDAS	0.69	0.79	0.24	1.55	10.57
900	LASSO	2.52	0.89	0.85	1.04	62.28
	MCP	10.89	0.82	0.40	1.42	14.17
	SCAD	20.12	0.87	0.66	1.21	27.17
	SPDAS	0.74	0.74	0.24	1.50	10.13
1100	LASSO	2.76	0.87	0.86	1.01	65.57
	MCP	10.84	0.82	0.39	1.43	14.01
	SCAD	21.01	0.87	0.67	1.20	27.80
	SPDAS	0.81	0.74	0.22	1.52	9.86
1300	LASSO	2.85	0.87	0.86	1.01	65.57
	MCP	12.36	0.80	0.42	1.38	14.37
	SCAD	23.09	0.87	0.69	1.18	29.95
	SPDAS	0.94	0.71	0.25	1.46	9.8
1500	LASSO	3.05	0.87	0.87	1.00	68.59
	MCP	13.05	0.80	0.44	1.36	14.65
	SCAD	23.79	0.86	0.70	1.16	29.83
	SPDAS	1.03	0.72	0.24	1.48	9.95
1700	LASSO	3.22	0.84	0.85	0.99	64.88
	MCP	11.96	0.78	0.46	1.32	15.03
	SCAD	23.68	0.85	0.72	1.13	31.93
	SPDAS	1.16	0.71	0.21	1.50	9.48

`cjlin/libsvmtools/datasets/`. The duke breast-cancer and leukemia data sets have been standardised such that the mean of each predictor is 0 and variance is 1. The response variable takes the value $y = 1$ if the subject has the disease and $y = 0$ otherwise. We fit these data sets with the logistic regression model and compare the classification accuracy rate of the SPDAS algorithm with LASSO, MCP and SCAD. Table 6 shows that the classification accuracy rates of SPDAS are comparable with those of LASSO, MCP and SCAD. Moreover, Table 7 demonstrates that for every data set, the number of selected variables $\hat{T}$ for SPDAS is similar to other methods.

Table 4: Numerical results (APDR, AFDR, ADR), $n = 200$, $p = 600$, $T = 10$, $R = 5$, $\rho = 0.1 : 0.1 : 0.9$.

ρ	Method	Times(s)	APDR	AFDR	ADR	MSES
0.1	LASSO	2.13	0.90	0.83	1.07	57.62
	MCP	10.53	0.86	0.35	1.51	13.8
	SCAD	19.93	0.90	0.62	1.28	24.28
	SPDAS	0.63	0.79	0.21	1.58	10.24
0.2	LASSO	2.15	0.92	0.84	1.08	59.61
	MCP	10.66	0.87	0.37	1.50	14.24
	SCAD	19.04	0.91	0.62	1.29	24.87
	SPDAS	0.61	0.81	0.24	1.57	11.05
0.3	LASSO	2.14	0.93	0.84	1.09	62.32
	MCP	10.39	0.85	0.35	1.50	13.54
	SCAD	18.54	0.92	0.60	1.32	24
	SPDAS	0.61	0.81	0.21	1.60	10.63
0.4	LASSO	2.07	0.92	0.84	1.08	58.9
	MCP	10.80	0.85	0.35	1.50	13.39
	SCAD	18.87	0.91	0.62	1.29	24.65
	SPDAS	0.60	0.77	0.22	1.55	10.34
0.5	LASSO	2.17	0.91	0.84	1.07	57.61
	MCP	10.36	0.83	0.34	1.49	13.08
	SCAD	19.56	0.89	0.60	1.29	23.12
	SPDAS	0.61	0.79	0.23	1.56	10.60
0.6	LASSO	2.31	0.90	0.83	1.07	55.92
	MCP	10.43	0.81	0.36	1.45	12.95
	SCAD	19.96	0.88	0.60	1.28	22.89
	SPDAS	0.59	0.76	0.23	1.53	10.25
0.7	LASSO	2.49	0.89	0.84	1.05	57.49
	MCP	10.32	0.79	0.33	1.46	12.06
	SCAD	19.79	0.85	0.61	1.24	22.66
	SPDAS	0.57	0.74	0.26	1.48	10.27
0.8	LASSO	2.94	0.86	0.83	1.03	51.73
	MCP	10.86	0.68	0.40	1.28	11.72
	SCAD	20.57	0.80	0.62	1.18	21.44
	SPDAS	0.50	0.67	0.33	1.34	10.39
0.9	LASSO	3.78	0.77	0.84	0.93	48.19
	MCP	11.45	0.51	0.52	0.99	10.75
	SCAD	19.33	0.63	0.64	0.99	18.03
	SPDAS	0.40	0.51	0.47	1.04	10.12

Table 5: Description of four real data sets.

Data name	n samples	p features	training size n_1	testing set n_2
duke breast-cancer	42	7129	38	4
gisette	7000	5000	6000	1000
leukemia	72	7129	38	34
splice	3175	60	1000	2175

Table 6: Classification accuracy rate.

Data name	SPDAS	LASSO	MCP	SCAD
duke breast-cancer	75%	1	25%	75%
gisette	54.70%	51.30%	59.90%	57.10%
leukemia	94.12%	91.17%	94.11%	91.17%
splice	84.18%	85.70%	84.91%	85.01%

Table 7: The number of selected variables ($\hat{T}$).

Data name	SPDAS	LASSO	MCP	SCAD
duke breast-cancer	14	23	5	17
gisette	47	507	49	121
leukemia	14	13	4	11
splice	22	40	26	33

5. Conclusion

Using the hard thresholding regularisation [1], we introduce the primal and dual active sets algorithm for variable estimation and selection in high-dimensional and sparse logistic regression models. In addition, we propose a sequential version of this algorithm (abbreviated as SPDAS) with a warm-start strategy. We also obtain the sharp nonasymptotic error bounds in ℓ_1- and ℓ_2-norms for the global solution of the hard thresholding regularisation problem and study its support recovery property. Simulations and real data analysis show that SPDAS outperforms LASSO, MCP and SCAD methods in terms of computational efficiency, estimation accuracy, support recovery and classification.

Appendix A

Let us recall auxiliary results needed for the proofs of Theorem 2.1 and Lemma 3.1.

Lemma A.1 (cf. Van de Geer [8, Lemma 5]). *Let $\mathbf{v} \in \mathbb{R}^p$, and let A denote the index set of the T largest elements of $\mathbf{v}$ in magnitude. Suppose $\xi > 0$ such that $\xi\rho_\lambda(\mathbf{v}_A) - \rho_\lambda(\mathbf{v}_{A^c}) \geq 0$. Then*

$$\xi\rho_\lambda(\mathbf{v}_A) - \rho_\lambda(\mathbf{v}_{A^c}) \leq \lambda\left(\xi\|\mathbf{v}_A\|_1 - \|\mathbf{v}_{A^c}\|_1\right).$$

Moreover, if $\boldsymbol{\beta}^* \in \mathbb{R}^p$ is T-sparse, that is $|A^*| = T$, then for any $\boldsymbol{\beta} \in \mathbb{R}^p$ satisfying $\xi\rho_\lambda(\boldsymbol{\beta}^*) - \rho_\lambda(\boldsymbol{\beta}) > 0$ with $\xi \geq 1$, we have

$$\xi\rho_\lambda(\boldsymbol{\beta}^*) - \rho_\lambda(\boldsymbol{\beta}) \leq \lambda\big(\xi\|\boldsymbol{v}_A\|_1 - \|\boldsymbol{v}_{A^c}\|_1\big),$$

where $\boldsymbol{v} = \boldsymbol{\beta} - \boldsymbol{\beta}^*$ and A is the index set of the T largest elements of $\boldsymbol{v}$ in absolute values.

Lemma A.2 (cf. Van de Geer [8, Corollary 2])**.** *Assume the entries of $\mathbf{X}$ are sub-Gaussian and $n \gtrsim \log(p)$, then there exists universal constants (c_1, c_2, c_3) with $0 < c_i < \infty$, $i = 1, 2, 3$ such that*

$$P\left(\|\nabla\mathscr{L}_n(\boldsymbol{\beta}^*)\|_\infty \geq c_1\sqrt{\frac{\log(p)}{n}}\right) \leq c_2\exp\big(-c_3\log(p)\big).$$

Now we can proceed with the proofs of main results.

A.1 Proof of Theorem 2.1

Denote $\hat{\boldsymbol{\Delta}} = \boldsymbol{\beta}^\diamond - \boldsymbol{\beta}^*$. We first show that $\|\hat{\boldsymbol{\Delta}}\|_2 \leq 1$. Otherwise, if $\|\hat{\boldsymbol{\Delta}}\|_2 > 1$, the RSC condition implies

$$\left\langle\nabla\mathscr{L}_n(\boldsymbol{\beta}^\diamond) - \nabla\mathscr{L}_n(\boldsymbol{\beta}^*), \hat{\boldsymbol{\Delta}}\right\rangle \geq \alpha_2\|\hat{\boldsymbol{\Delta}}\|_2 - \tau_2\sqrt{\frac{\log(p)}{n}}\|\hat{\boldsymbol{\Delta}}\|_1.$$

Therefore,

$$\left\langle-\nabla\rho_\lambda(\boldsymbol{\beta}^\diamond) - \nabla\mathscr{L}_n(\boldsymbol{\beta}^*), \hat{\boldsymbol{\Delta}}\right\rangle \geq \alpha_2\|\hat{\boldsymbol{\Delta}}\|_2 - \tau_2\sqrt{\frac{\log(p)}{n}}\|\hat{\boldsymbol{\Delta}}\|_1. \tag{A.1}$$

It follows from the Hölder and triangle inequalities that

$$\left\langle-\nabla\rho_\lambda(\boldsymbol{\beta}^\diamond) - \nabla\mathscr{L}_n(\boldsymbol{\beta}^*), \hat{\boldsymbol{\Delta}}\right\rangle \leq \big(\|\nabla\rho_\lambda(\boldsymbol{\beta}^\diamond)\|_\infty + \|\nabla\mathscr{L}_n(\boldsymbol{\beta}^*)\|_\infty\big)\|\hat{\boldsymbol{\Delta}}\|_1.$$

Since

$$4\max\left\{\|\nabla\mathscr{L}_n(\boldsymbol{\beta}^*)\|_\infty, \alpha_2\sqrt{\frac{\log(p)}{n}}\right\} \leq \lambda,$$

we have $\|\nabla\mathscr{L}_n(\boldsymbol{\beta}^*)\|_\infty \leq \lambda/2$. Taking into account the estimate $\|\nabla\rho_\lambda(\boldsymbol{\beta}^\diamond)\|_\infty \leq \lambda$, we obtain

$$\left\langle-\nabla\rho_\lambda(\boldsymbol{\beta}^\diamond) - \nabla\mathscr{L}_n(\boldsymbol{\beta}^*), \hat{\boldsymbol{\Delta}}\right\rangle \leq \frac{3\lambda}{2}\|\hat{\boldsymbol{\Delta}}\|_1, \tag{A.2}$$

and the inequalities (A.1) and (A.2) show that

$$\|\hat{\boldsymbol{\Delta}}\|_2 \leq \frac{\|\hat{\boldsymbol{\Delta}}\|_1}{\alpha_2}\left(\frac{3\lambda}{2} + \tau_2\sqrt{\frac{\log(p)}{n}}\right) \leq \frac{2R}{\alpha_2}\left(\frac{3\lambda}{2} + \tau_2\sqrt{\frac{\log(p)}{n}}\right). \tag{A.3}$$

Since

$$4\max\left\{\|\nabla\mathscr{L}_n(\boldsymbol{\beta}^*)\|_\infty, \alpha_2\sqrt{\frac{\log(p)}{n}}\right\} \leq \lambda \leq \frac{\alpha_2}{6R}$$

and

$$n \geq \left(\frac{16R^2}{\alpha_2^2} \max\left(\tau_1^2, \tau_2^2\right) \right) \log(p),$$

the right-hand side of (A.3) does not exceed 1, so that $\|\hat{\boldsymbol{\Delta}}\|_2 \leq 1$. The RSC condition then implies

$$\left\langle \nabla \mathscr{L}_n(\boldsymbol{\beta}^\diamond) - \nabla \mathscr{L}_n(\boldsymbol{\beta}^*), \hat{\boldsymbol{\Delta}} \right\rangle \geq \alpha_1 \|\hat{\boldsymbol{\Delta}}\|_2^2 - \tau_1 \frac{\log(p)}{n} \|\hat{\boldsymbol{\Delta}}\|_1^2. \tag{A.4}$$

The convexity of $\rho_{\lambda,\mu}(\boldsymbol{\beta})$ yields

$$\rho_{\lambda,\mu}(\boldsymbol{\beta}^*) - \rho_{\lambda,\mu}(\boldsymbol{\beta}^\diamond) \geq \left\langle \nabla \rho_{\lambda,\mu}(\boldsymbol{\beta}^\diamond), \boldsymbol{\beta}^* - \boldsymbol{\beta}^\diamond \right\rangle = \left\langle \nabla \rho_\lambda(\boldsymbol{\beta}^\diamond) + \mu \boldsymbol{\beta}^\diamond, \boldsymbol{\beta}^* - \boldsymbol{\beta}^\diamond \right\rangle.$$

Therefore,

$$\left\langle \nabla \rho_\lambda(\boldsymbol{\beta}^\diamond), \boldsymbol{\beta}^* - \boldsymbol{\beta}^\diamond \right\rangle \leq \rho_\lambda(\boldsymbol{\beta}^*) - \rho_\lambda(\boldsymbol{\beta}^\diamond) + \frac{\mu}{2} \|\boldsymbol{\beta}^\diamond - \boldsymbol{\beta}^*\|^2. \tag{A.5}$$

Combining (A.4) and (A.5), we write

$$\alpha_1 \|\hat{\boldsymbol{\Delta}}\|_2^2 - \tau_1 \frac{\log(p)}{n} \|\hat{\boldsymbol{\Delta}}\|_1^2 \leq -\left\langle \nabla \mathscr{L}_n(\boldsymbol{\beta}^*), \hat{\boldsymbol{\Delta}} \right\rangle + \rho_\lambda(\boldsymbol{\beta}^*) - \rho_\lambda(\boldsymbol{\beta}^\diamond) + \frac{\mu}{2} \|\hat{\boldsymbol{\Delta}}\|_2^2.$$

Hence,

$$\begin{aligned} \left(\alpha_1 - \frac{\mu}{2}\right) \|\hat{\boldsymbol{\Delta}}\|_2^2 &\leq \rho_\lambda(\boldsymbol{\beta}^*) - \rho_\lambda(\boldsymbol{\beta}^\diamond) + \|\nabla \mathscr{L}_n(\boldsymbol{\beta}^*)\|_\infty \cdot \|\hat{\boldsymbol{\Delta}}\|_1 + \tau_1 \frac{\log(p)}{n} \|\hat{\boldsymbol{\Delta}}\|_1^2 \\ &\leq \rho_\lambda(\boldsymbol{\beta}^*) - \rho_\lambda(\boldsymbol{\beta}^\diamond) + \left(\|\nabla \mathscr{L}_n(\boldsymbol{\beta}^*)\|_\infty + 4R\tau_1 \frac{\log(p)}{n} \right) \|\hat{\boldsymbol{\Delta}}\|_1. \end{aligned} \tag{A.6}$$

The assumption

$$4 \max\left\{ \|\nabla \mathscr{L}_n(\boldsymbol{\beta}^*)\|_\infty, \alpha_2 \sqrt{\frac{\log(p)}{n}} \right\} \leq \lambda \leq \frac{\alpha_2}{6R}$$

gives

$$\|\nabla \mathscr{L}_n(\boldsymbol{\beta}^*)\|_\infty + 4R\tau_1 \frac{\log(p)}{n} \leq \frac{\lambda}{4} + \alpha_2 \sqrt{\frac{\log(p)}{n}} \leq \frac{\lambda}{2}. \tag{A.7}$$

Using (A.6), (A.7) and the subadditivity of $\rho_\lambda(\cdot)$, we arrive at the estimate

$$\begin{aligned} \left(\alpha_1 - \frac{\mu}{2}\right) \|\hat{\boldsymbol{\Delta}}\|_2^2 &\leq \rho_\lambda(\boldsymbol{\beta}^*) - \rho_\lambda(\boldsymbol{\beta}^\diamond) + \frac{\lambda}{2} \cdot \left(\frac{\rho_\lambda(\hat{\boldsymbol{\Delta}})}{\lambda} + \frac{\mu}{2\lambda} \|\hat{\boldsymbol{\Delta}}\|_2^2 \right) \\ &\leq \rho_\lambda(\boldsymbol{\beta}^*) - \rho_\lambda(\boldsymbol{\beta}^\diamond) + \frac{1}{2}\left(\rho_\lambda(\boldsymbol{\beta}^*) + \rho_\lambda(\boldsymbol{\beta}^\diamond)\right) + \frac{\mu}{4} \|\hat{\boldsymbol{\Delta}}\|_2^2. \end{aligned}$$

Therefore,

$$0 \leq \left(\alpha_1 - \frac{3\mu}{4}\right) \|\hat{\boldsymbol{\Delta}}\|_2^2 \leq \frac{3}{2} \rho_\lambda(\boldsymbol{\beta}^*) - \frac{1}{2} \rho_\lambda(\boldsymbol{\beta}^\diamond).$$

In particular, if $\xi = 3$, then $\xi\rho_\lambda(\boldsymbol{\beta}^*) - \rho_\lambda(\boldsymbol{\beta}^\diamond) \geq 0$. Recalling Lemma A.1, we write

$$3\rho_\lambda(\boldsymbol{\beta}^*) - \rho_\lambda(\boldsymbol{\beta}^\diamond) \leq 3\lambda\|\hat{\boldsymbol{\Delta}}_A\|_1 - \lambda\|\hat{\boldsymbol{\Delta}}_{A^c}\|_1, \tag{A.8}$$

where A refers to the index set of the T largest elements of $\boldsymbol{\beta}^\diamond - \boldsymbol{\beta}^*$ in absolute values with $T = |A^*|$. Then we have the cone condition

$$\|\hat{\boldsymbol{\Delta}}_{A^c}\|_1 \leq 3\|\hat{\boldsymbol{\Delta}}_A\|_1. \tag{A.9}$$

Substituting (A.9) into (A.8) yields

$$\left(2\alpha_1 - \frac{3\mu}{2}\right)\|\hat{\boldsymbol{\Delta}}\|_2^2 \leq 3\lambda\|\hat{\boldsymbol{\Delta}}_A\|_1 - \lambda\|\hat{\boldsymbol{\Delta}}_{A^c}\|_1 \leq 3\lambda\|\hat{\boldsymbol{\Delta}}_A\|_1 \leq 3\lambda\sqrt{|A^*|}\|\hat{\boldsymbol{\Delta}}\|_2,$$

so that

$$\|\hat{\boldsymbol{\Delta}}\|_2 \leq \frac{6\lambda\sqrt{|A^*|}}{4\alpha_1 - 3\mu}.$$

It follows from (A.9) that

$$\|\hat{\boldsymbol{\Delta}}\|_1 \leq 4\|\hat{\boldsymbol{\Delta}}_A\|_1 \leq 4\sqrt{|A^*|}\|\hat{\boldsymbol{\Delta}}\|_2.$$

Therefore,

$$\|\hat{\boldsymbol{\Delta}}\|_1 \leq \frac{24\lambda|A^*|}{4\alpha_1 - 3\mu}.$$

Besides, if the entries of $\mathbf{X}$ are sub-Gaussian, then according to Lemma A.2, there are universal finite and positive constants (c_1, c_2, c_3) such that

$$\|\nabla\mathscr{L}_n(\boldsymbol{\beta}^*)\|_\infty \leq c_1\sqrt{\frac{\log(p)}{n}}$$

with the probability at least $1 - c_2\exp(-c_3\log(p))$.

Choosing α_2 and R such that

$$4(c_1 + \alpha_2)\sqrt{\frac{\log p}{n}} \leq \frac{\alpha_2}{6R}$$

and setting

$$\lambda = 4(c_1 + \alpha_2)\sqrt{\frac{\log p}{n}},$$

we obtain that

$$\|\boldsymbol{\beta}^\diamond - \boldsymbol{\beta}^*\|_1 \leq \frac{96|A^*|(c_1 + \alpha_2)\sqrt{\log(p)/n}}{4\alpha_1 - 3\mu},$$
$$\|\boldsymbol{\beta}^\diamond - \boldsymbol{\beta}^*\|_2 \leq \frac{24(c_1 + \alpha_2)\sqrt{|A^*|\log(p)/n}}{4\alpha_1 - 3\mu}$$

with the probability at least $1 - c_2\exp(-c_3\log(p))$. Theorem 2.1 is proven. □

A.2 Proof of Lemma 3.1

Let

$$L_\lambda(\boldsymbol{\beta}) = \mathscr{L}_n(\boldsymbol{\beta}) + \sum_{i=1}^{p} \rho_\lambda(\beta_i) = \frac{1}{n}\sum_{i=1}^{n} \log\big(1+\exp(\mathbf{x}_i^T\boldsymbol{\beta})\big) - \frac{\mathbf{Y}^T\mathbf{X}\boldsymbol{\beta}}{n} + \sum_{i=1}^{p} \rho_\lambda(\beta_i)$$

and $\boldsymbol{\beta}^\diamond = (\beta_1^\diamond, \ldots, \beta_p^\diamond) \in \mathbb{R}^p$ be a minimiser of the function L_λ. According to [14, Theorem 10.1], we have

$$\mathbf{0} \in \nabla \mathscr{L}_n(\boldsymbol{\beta}^\diamond) + \sum_{i=1}^{p} \partial \rho_\lambda(\beta_i^\diamond), \tag{A.10}$$

where $\partial\rho_\lambda(\beta_i^\diamond)$ denotes the limiting subdifferential of ρ_λ at $\beta_i^\diamond$. Let $\mathbf{d}^\diamond = -\nabla\mathscr{L}_n(\boldsymbol{\beta}^\diamond)$, cf. [14, Definition 8.3]. Define

$$G(\boldsymbol{\beta}) = \frac{1}{2}\|\boldsymbol{\beta} - (\boldsymbol{\beta}^\diamond + \mathbf{d}^\diamond)\|^2 + \sum_{i=1}^{p} \partial\rho_\lambda(\beta_i^\diamond)$$

and note that the relation (A.10) is equivalent to

$$\mathbf{0} \in \boldsymbol{\beta}^\diamond - (\boldsymbol{\beta}^\diamond + \mathbf{d}^\diamond) + \sum_{i=1}^{p} \partial\rho_\lambda(\beta_i^\diamond).$$

Moreover, $\widetilde{\boldsymbol{\beta}}$ is the minimiser of $G(\boldsymbol{\beta})$ if and only if $\mathbf{0} \in \partial G(\widetilde{\boldsymbol{\beta}})$. Obviously, $\mathbf{0} \in \partial G(\boldsymbol{\beta}^\diamond)$. Therefore, $\boldsymbol{\beta}^\diamond$ is a KKT point of $G(\boldsymbol{\beta})$. Consequently, $\boldsymbol{\beta}^\diamond = \Gamma_\lambda(\boldsymbol{\beta}^\diamond + \mathbf{d}^\diamond)$, since the KKT points of $G(\boldsymbol{\beta})$ coincide with its coordinate-wise minimisers [5].

On the other hand, assuming that $\boldsymbol{\beta}^\diamond$ and $\mathbf{d}^\diamond$ satisfy (3.1), we show that $\boldsymbol{\beta}^\diamond$ is a stationary point of (1.2). Indeed, let

$$A^\diamond := \big\{i : |\beta_i^\diamond + d_i^\diamond| \geq \lambda\big\}, \quad I^\diamond := \big\{i : |\beta_i^\diamond + d_i^\diamond| < \lambda\big\}.$$

By the definition of $\Gamma_\lambda(\cdot)$ in (3.2) and (3.1), we conclude that $|\beta_i^\diamond| \geq \lambda$ when $i \in A^\diamond$ and $\boldsymbol{\beta}_{I^\diamond}^\diamond = 0$. It follows that $\operatorname{supp}(\boldsymbol{\beta}^\diamond) = A^\diamond$. In addition, we have $\mathbf{d}_{A^\diamond}^\diamond = [-\nabla\mathscr{L}_n(\boldsymbol{\beta}^\diamond)]_{A^\diamond} = 0$, which is equivalent to $\boldsymbol{\beta}_{A^\diamond}^\diamond \in \operatorname{argmin}_{\boldsymbol{\beta}_{A^\diamond}} \widetilde{\mathscr{L}_n}(\boldsymbol{\beta}_{A^\diamond})$. Hence $\boldsymbol{\beta}^\diamond$ and $\mathbf{d}^\diamond$ satisfy (A.10), so that $\boldsymbol{\beta}^\diamond$ is a stationary point of (1.2). □

Acknowledgments

We wish to thank two anonymous reviewers for their constructive comments, which helped to improve the manuscript significantly.

The work of Y. Liu is supported in part by the National Science Foundation of China (Grant No. 11971362) and the work of C. Zhu is supported in part by the National Science Foundation of China (Grant No. 81873793).

References

[1] A. Antoniadis, *Wavelet methods in statistics: Some recent developments and their applications*, Stat. Surv. **1**, 16–55 (2007).

[2] P. Breheny and J. Huang, *Coordinate descent algorithms for nonconvex penalized regression, with applications to biological feature selection*, Ann. Appl. Stat. **5(1)**, 232 (2011).

[3] J. Fan and R. Li, *Variable selection via nonconcave penalized likelihood and its oracle properties*, J. Amer. Statist. Assoc. **96(456)**, 1348–1360 (2001).

[4] J. Friedman, T. Hastie and R. Tibshirani, *Regularization paths for generalized linear models via coordinate descent*, J. Stat. Softw. **33(1)**, 1 (2010).

[5] J. Huang, Y. Jiao, B. Jin, J. Liu, X. Lu and C. Yang, *A unified primal dual active set algorithm for nonconvex sparse recovery*, Statist. Sci. **36(2)**, 215–238 (2021).

[6] Y. Kim, S. Kwon and H. Choi, *Consistent model selection criteria on high dimensions*, J. Mach. Learn. Res. **13**, 1037–1057 (2012).

[7] X. Li, L. Yang, J. Ge, J. Haupt, T. Zhang and T. Zhao, *On quadratic convergence of dc proximal Newton algorithm in nonconvex sparse learning*, NIPS. **30** (2017).

[8] P.L. Loh and M.J. Wainwright, *Regularized m-estimators with nonconvexity: Statistical and algorithmic theory for local optima*, J. Mach. Learn. Res. **16(1)**, 559–616 (2015).

[9] S. Luo and Z. Chen, *Sequential lasso cum ebic for feature selection with ultra-high dimensional feature space*, J. Amer. Statist. Assoc. **109(507)**, 1229–1240 (2014).

[10] P. McCullagh and J.A. Nelder, *Generalized Linear Models*, Routledge (2019).

[11] J.A. Nelder and R.W. Wedderburn, *Generalized linear models*, J. Roy. Statist. Soc. Ser. A. **135(3)**, 370–384 (1972).

[12] Y. Nesterov, *Gradient methods for minimizing composite functions*, Math. Program. **140(1)**, 125–161 (2013).

[13] M.Y. Park and T. Hastie, *L1-regularization path algorithm for generalized linear models*, J. R. Stat. Soc. Ser. B. Stat. Methodol. **69(4)**, 659–677 (2007).

[14] R.T. Rockafellar and R.J.B. Wets, *Variational Analysis (Vol. 317)*, Springer Science & Business Media (2009).

[15] Y. Shi, J. Huang, Y. Jiao and Q. Yang, *A semismooth newton algorithm for high-dimensional nonconvex sparse learning*, IEEE Trans. Neural Netw. Learn. Syst. **31(8)**, 2993–3006 (2019).

[16] R. Tibshirani, *Regression shrinkage and selection via the lasso*, J. R. Stat. Soc. Ser. B. Stat. Methodol. **58(1)**, 267–288 (1996).

[17] S.A. Van de Geer, *High-dimensional generalized linear models and the lasso*, Ann. Statist. **36(2)**, 614–645 (2008).

[18] L. Wang, Y. Kim and R. Li, *Calibrating non-convex penalized regression in ultra-high dimension*, Ann. Statist. **41(5)**, 2505 (2013).

[19] Z. Wang, H. Liu and T. Zhang, *Optimal computational and statistical rates of convergence for sparse nonconvex learning problems*, Ann. Statist. **42(6)**, 2164 (2014).

[20] C.H. Zhang and T. Zhang, *A general theory of concave regularization for high-dimensional sparse estimation problems*, Statist. Sci. **27(4)**, 576–593 (2012).

[21] H. Zou and T. Hastie, *Regularization and variable selection via the elastic net*, J. R. Stat. Soc. Ser. B. Stat. Methodol. **67(2)**, 301–320 (2005).