이학석사학위논문

Optimization Methods for SCAD-penalized Support Vector Machine

SCAD-벌점화 지지벡터기계 모형에 대한 최적화 방법들

2018 년 2 월

서울대학교 대학원

통계학과

이 한 별

Abstract

Optimization Methods for SCAD-penalized Support Vector Machine

Lee Hanbyul
Department of Statistics
The Graduate School
Seoul National University

The support vector machine (SVM) is a powerful tool for binary classification problem, but it is adversely affected when redundant variables are involved. Several variants of the SVM have been proposed to rectify this problem. Among them, the smoothly clipped absolute deviation penalized SVM (SCAD SVM) has been proven to perform effective variable selection. However, issues regarding non-convexity and multiple local minimums are evident in the process of optimization. This paper summarizes the local quadratic approximation (LQA) and the local linear approximation (LLA) methods, which are primary optimization methods for the SCAD SVM, and further brings two new approaches. First, the envelope method is applied in the derivation of each algorithm instead of the usual Taylor series expansion, which is a more generalized method for the derivation than the conventional one. Next, in addition to the previously known limitations of the LQA method and the comparative advantages of the LLA method, we suggest the insensitivity to initial value of the LLA method and present theories

about the convergence of the LLA algorithm to the oracle estimator for arbitrary initial value. Lastly, we verify through a simulation study that the LLA method gives better results for any initial values than the LQA method.

Keywords: Local approximation algorithm, Smoothly clipped absolute deviation penalty, Support vector machine, Variable selection, Initialization
Student Number: 2016-20274

Contents

List of Tables

Chapter 1

Introduction

The support vector machine (SVM) introduced by Cortes and Vapnik (1995) is a powerful binary classification tool with high accuracy and great flexibility and has been successful in many applications. However, one serious drawback of the standard SVM is that its performance can be adversely affected if many redundant variables are included because its decision rule utilizes all the variables without discrimination (Friedman et al., 2001). To deal with this problem, many variable selection methods have been proposed. Guyon et al. (2002) suggested the recursive feature elimination algorithm, which successively eliminates variables by training a sequence of SVM classifiers. Another approach was to achieve variable selection and prediction simultaneously, by considering the standard SVM in the regularization framework of hinge loss plus the L_2 penalty and replacing the L_2 penalty with another penalty function. Bradley and Mangasarian (1998) suggested the L_1 SVM imposing the absolute value penalty, Wang et al. (2006) proposed to use the elastic net penalty, and Zou (2007) brought up the adaptive

lasso penalty.

Zhang et al. (2006) suggested the smoothly clipped absolute deviation (Fan and Li, 2001) penalized SVM (SCAD SVM) at the first time. The method was applied to the problem of gene selection and produced good results. Since then, Park et al. (2012) studied the oracle property of the SCAD SVM when the number of variables is fixed. It was shown that there exists a local minimizer of the SCAD SVM objective function, which becomes asymptotically the same as the oracle estimator. More recently, Zhang et al. (2016) extended the theory to the situation where the number of variables grows exponentially with the sample size.

Although the SCAD SVM method has the good oracle property, it is hard to find the appropriate solution near the oracle estimator because the SCAD SVM objective function is not convex and might have multiple local minimums. To remedy this problem, the local approximation algorithms with good initial values have been proposed. Zhang et al. (2006) introduced the local quadratic approximation (LQA) algorithm using L_2 approximation, and Zhang et al. (2016) applied the local linear approximation (LLA) algorithm (Zou and Li, 2008) to the optimization problem and studied its convergence in the moderately high dimensional setting.

In this paper, we review the LQA and LLA algorithms finding an estimate of SCAD SVM and bring two new approaches. First, we derive these algorithms using the envelope method (Polson and Scott, 2016), which is a more generalized theory for the derivation than the conventional Taylor series expansion. Second, we study the initialization of the LQA and LLA methods which are to handle the problem of multiple local minimums. We suggest that, in addition to the previously known limitations of the LQA method and the comparative advantages

of the LLA method, the LLA algorithm relatively exhibits insensitivity to initial value unlike the LQA method whose results differ greatly depending on what the initial value is. We explain the theoretical basis for this argument based on the theorems of Zhang et al. (2016) which describes that the LLA algorithm asymptotically identifies the oracle estimator within a small number of iterations when the initial value is given as the estimate of the L_1 SVM. In this paper, we modify this theory a little bit and demonstrate that the LLA algorithm finds the oracle estimator well even if the initial value is given randomly.

This paper is organized as follows. In Chapter 2, we describe the LQA and LLA algorithms with their derivation and properties. Chapter 3 contains the theorems to show the insensitivity to initialization of LLA algorithm. Numerical illustrations via a simulation study are provided in Chapter 4, followed by a conclusion in Chapter 5. Technical proofs are presented in Appendix A.

Chapter 2

Local Approximation Algorithms

In this chapter, we derive the LQA and LLA algorithms which were introduced by Zhang et al. (2006) and Zhang et al. (2016) to solve the SCAD SVM problem, and examine the good properties of the LLA algorithm over the LQA algorithm. Given a random sample $\{(y_i, X_i)\}_{i=1}^n$ for the categorical dependent variable $y \in \{-1, 1\}$ and the independent variables $X = (1, x_1, \cdots, x_p)^T$, the objective function of the SCAD SVM to minimize is

$$l_n(\beta) = \frac{1}{n}\sum_{i=1}^{n}[1 - y_i X_i^T \boldsymbol{\beta}]_+ + \sum_{j=1}^{p} p_\lambda(|\beta_j|) \tag{1}$$

and the SCAD penalty function is

$$\begin{aligned} p_\lambda(|\beta|) =& \lambda|\beta| I(0 \le |\beta| < \lambda) + \frac{a\lambda|\beta| - (\beta^2 + \lambda^2)/2}{a-1} I(\lambda \le |\beta| < a\lambda) \\ &+ \frac{(a+1)\lambda^2}{2} I(|\beta| \ge a\lambda) \end{aligned} \tag{2}$$

for some $a > 2$.

We note that only linear SVMs are considered in this paper because linear classifiers often give better performances than non-linear ones in many applications on high-dimensional data (Friedman et al., 2001). Since the objective function is sum of the non-differentiable loss function and the non-convex penalty, it is not easy to find the minimum value. That is why we need an appropriate approximation method like the LQA and LLA algorithms.

Although the LQA and LLA algorithms can be derived with the Taylor series expansion, here we try to explain these in an envelope framework (Polson and Scott, 2016) that can be more generally applied to non-differentiable functions. The envelope representation theorem for concave functions discussed in Polson and Scott (2016) is as follows.

Theorem 1 *Suppose that $\phi(x)$ is a symmetric function and that is concave and nondecreasing on $\mathcal{R}^+$. Then ϕ can be represented in terms of its concave dual ϕ^*:*

$$\phi(x) = \inf_{\gamma \geq 0} \{\gamma |x| - \phi^*(\gamma)\}$$

where $\phi^*(\gamma) = \inf_{x \geq 0} \{\gamma x - \phi(x)\}$. Also, the minimization value of γ for fixed x satisfies $\hat{\gamma}(x) \in \partial \phi(|x|)$.

Theorem 1 leads to the following Corollary 1.

Corollary 1 *Suppose that $\phi(x)$ is symmetric and $\phi(\sqrt{x})$ is concave and nondecreasing on $\mathcal{R}^+$. Then*

$$\phi(|x|) = \inf_{\gamma \geq 0} \{\gamma x^2 - \theta^*(\gamma)\}$$

where $\theta(x) = \phi(\sqrt{x})$ *on* $\mathcal{R}^+$. *The minimization value of* γ *for fixed* x *satisfies* $\hat{\gamma}(x) \in \partial\theta(x^2)$.

Now we apply the above theorems to derive the LQA and LLA algorithms. Both are MM (majorize-minimize) algorithms, thus ensure convergence.

Local Quadratic Approximation Algorithm

First, we check the quadratic approximation of the hinge loss function, $f(x) \coloneqq (1-x)_+$.

Let $\phi(x) \coloneqq f(x+1) + \frac{1}{2}x = \frac{1}{2}|x|$. Then by Corollary 1, it can be represented as $\phi(x) = \inf_{\gamma \geq 0}\{\gamma x^2 - \theta^*(\gamma)\}$. This implies that

$$f(x) = \inf_{\gamma \geq 0}\left\{\gamma\left(x - 1 - \frac{1}{4\gamma}\right)^2 - \frac{1}{16\gamma} - \theta^*(\gamma)\right\}$$

where the minimization value of γ for fixed x is $\hat{\gamma}(x) = \frac{1}{4|x-1|}$. Then given a fixed $\boldsymbol{\beta}_o$, the hinge loss function can be approximated as

$$\begin{aligned}
f(yX^T\boldsymbol{\beta}) &= (1 - yX^T\boldsymbol{\beta})_+ \\
&\approx \frac{1}{4|yX^T\boldsymbol{\beta}_o - 1|}\left(yX^T\boldsymbol{\beta} - 1 - \frac{1}{4 \cdot \frac{1}{4|yX^T\boldsymbol{\beta}_o - 1|}}\right)^2 + C(\boldsymbol{\beta}_o) \\
&= \frac{(yX^T\boldsymbol{\beta} - 1)^2 - 2|yX^T\boldsymbol{\beta}_o - 1|(yX^T\boldsymbol{\beta} - 1)}{4|yX^T\boldsymbol{\beta}_o - 1|} + \tilde{C}(\boldsymbol{\beta}_o) \\
&= \frac{(yX^T\boldsymbol{\beta} - 1)^2}{4|yX^T\boldsymbol{\beta}_o - 1|} + \frac{1 - yX^T\boldsymbol{\beta}}{2} + \tilde{C}(\boldsymbol{\beta}_o) \\
&= \frac{(X^T\boldsymbol{\beta} - y)^2}{4|X^T\boldsymbol{\beta}_o - y|} + \frac{1 - yX^T\boldsymbol{\beta}}{2} + \tilde{C}(\boldsymbol{\beta}_o).
\end{aligned}$$

Also, the SCAD penalty function $p_\lambda(|\beta|)$ (2) satisfies the conditions of $\phi(x)$ in Corollary 1, so it can be represented as

$$p_\lambda(|\beta|) = \inf_{\gamma \geq 0} \{\gamma\beta^2 - \theta^*(\gamma)\}$$

where $\theta(\beta) = p_\lambda(\sqrt{\beta})$ and $\hat{\gamma}(\beta) = \theta'(\beta^2) = \frac{p'_\lambda(|\beta|)}{2|\beta|}$. Therefore, given a fixed β_o, the SCAD penalty function can be approximated as

$$p_\lambda(|\beta|) \approx \frac{p'_\lambda(|\beta_o|)}{2|\beta_o|} \cdot \beta^2 + C(\beta_o).$$

Accordingly, the objective function of the SCAD SVM (1) can be approximated by the quadratic function as

$$\begin{aligned} l_n(\beta) &= \frac{1}{n}\sum_{i=1}^{n}[1 - y_i X_i^T \boldsymbol{\beta}]_+ + \sum_{j=1}^{p} p_\lambda(|\beta_j|) \\ &\approx \frac{1}{n}\sum_{i=1}^{n}\left\{\frac{(X_i^T\boldsymbol{\beta} - y_i)^2}{4|X_i^T\boldsymbol{\beta}_o - y_i|} + \frac{1 - y_i X_i^T\boldsymbol{\beta}}{2}\right\} + \sum_{j=1}^{p}\left\{\frac{p'_\lambda(|\beta_{oj}|)}{2|\beta_{oj}|}\cdot\beta_j^2\right\} + C(\boldsymbol{\beta}_o). \end{aligned}$$

The resulting local quadratic approximation algorithm is as follows: one starts with an initial value $\boldsymbol{\beta}^{(0)}$ and at each step $t \geq 0$, repeatedly solves

$$\boldsymbol{\beta}^{(t+1)} = \arg\min_{\boldsymbol{\beta}}\left[\frac{1}{n}\sum_{i=1}^{n}\left\{\frac{(X_i^T\boldsymbol{\beta} - y_i)^2}{4|X_i^T\boldsymbol{\beta}^{(t)} - y_i|} + \frac{1 - y_i X_i^T\boldsymbol{\beta}}{2}\right\} + \sum_{j=1}^{p}\left\{\frac{p'_\lambda(|\beta_j^{(t)}|)}{2|\beta_j^{(t)}|}\cdot\beta_j^2\right\}\right].$$

To avoid numerical instability, it is suggested that if $X_i^T\boldsymbol{\beta}^{(t)} - y_i \approx 0$, one replace it with sufficiently small η (Zhang et al., 2006), and if $\beta_j^{(t)} \approx 0$, say $|\beta_j^{(t)}| < \varepsilon_0$ (a prespecified value), then one set $\beta_j^{(t)} = 0$ and delete the jth component of X from the iteration (Fan and Li, 2001). Zou and Li (2008)

discussed the consequent weakness of the LQA algorithm. First, the elimination process leads to a drawback of backward stepwise variable selection: if a variable is deleted at any step, it is necessarily be excluded from the final selected model. Second, one has to choose η and ε_0, which becomes an additional tuning parameter. This sometimes can be difficult, and the size of ε_0 potentially affects the degree of sparsity as well as the speed of convergence. Finally, the initial value $\boldsymbol{\beta}^{(0)}$ must be well-defined. Zhang et al. (2006) empirically suggested the result of the standard SVM as the initial value.

Local Linear Approximation Algorithm

The derivation of the LLA algorithm is simpler than that of the LQA algorithm. Since the SCAD penalty function $p_\lambda(|\beta|)$ (2) satisfies the conditions of $\phi(x)$ in Theorem 1, it can be represented as

$$p_\lambda(|\beta|) = \inf_{\gamma \geq 0} \{\gamma|\beta| - p^*_\lambda(\gamma)\}$$

where the minimization value of γ for fixed β is $\hat{\gamma}(\beta) = p'_\lambda(|\beta|)$. Therefore, given a fixed β_o, the SCAD penalty function can be approximated as

$$p_\lambda(|\beta|) \approx \; p'_\lambda(|\beta_o|)|\beta| + C(\beta_o).$$

Accordingly, the objective function of the SCAD SVM (1) can be approxi-

mated by the linear function as

$$l_n(\beta) = \frac{1}{n}\sum_{i=1}^{n}[1 - y_i X_i^T \boldsymbol{\beta}]_+ + \sum_{j=1}^{p} p_\lambda(|\beta_j|)$$

$$\approx \frac{1}{n}\sum_{i=1}^{n}[1 - y_i X_i^T \boldsymbol{\beta}]_+ + \sum_{j=1}^{p} p'_\lambda(|\beta_{oj}|)|\beta_j| + C(\boldsymbol{\beta}_o).$$

Therefore, the resulting local linear approximation algorithm is as follows: one starts with an initial value $\boldsymbol{\beta}^{(0)}$ and at each step $t \geq 0$, repeatedly solves

$$\boldsymbol{\beta}^{(t+1)} = \arg\min_{\boldsymbol{\beta}} \left[\frac{1}{n}\sum_{i=1}^{n}[1 - y_i X_i^T \boldsymbol{\beta}]_+ + \sum_{j=1}^{p} p'_\lambda(|\beta_j^{(t)}|)|\beta_j|\right].$$

It is suggested that one take $p'_\lambda(|\beta_j^{(t)}|)| = \lambda$ when $\beta_j^{(t)} = 0$.

The above convex optimization problem can be easily recast as a linear programming problem (Zhang et al., 2016)

$$(\boldsymbol{\xi}^{(t+1)}, \boldsymbol{\nu}^{(t+1)}, \boldsymbol{\beta}^{(t+1)}) = \arg\min_{\boldsymbol{\xi},\boldsymbol{\nu},\boldsymbol{\beta}} \left[\frac{1}{n}\sum_{i=1}^{n}\xi_i + \sum_{j=1}^{p} p'_\lambda(|\beta_j^{(t)}|)\nu_j\right]$$

subject to

$$\begin{aligned}
\xi_i &\geq 0, && i = 1, 2, \cdots, n, \\
\xi_i &\geq 1 - y_i X_i^T \boldsymbol{\beta}, && i = 1, 2, \cdots, n, \\
\nu_j &\geq \beta_j, \quad \nu_j \geq -\beta_j, && j = 1, 2, \cdots, p.
\end{aligned}$$

The LLA algorithm has many good properties over the LQA algorithm (Zou and Li, 2008). First, unlike the LQA algorithm, one does not have to delete any small coefficient or introduce an additional tuning parameter in order to

avoid numerical instability. Second, the LLA is the best convex majorization of $p_\lambda(|\beta|)$, which means that for any convex majorization function $\psi(\cdot)$ of $p_\lambda(|\beta|)$ at β_0, the LLA approximation function $\psi^*(\cdot)$ at β_0 satisfies $\psi(\beta) \geq \psi^*(\beta)$ for all β.

Moreover, the LLA algorithm is particularly good at solving the SCAD SVM problem since it allows the minimization problem at each step to be a simple linear programming problem without any approximation of the hinge loss function. In addition, the LLA algorithm is relatively insensitive to initial value as compared with the LQA algorithm. This is discussed in detail in the next chapter.

Chapter 3

Insensitivity to Initialization of the LLA Algorithm

Using the two approximation methods discussed in the previous chapter, we can handle the non-convexity problem of the SCAD SVM objective function. However, since the objective function has multiple local minimums, the algorithms can converge to an undesired value if the initial value is poor. Therefore, previous studies have empirically and theoretically suggested the results of the standard SVM or the L_1 SVM as the initial values of the approximation algorithms (Zhang et al., 2006; Zhang et al., 2016).

In this chapter, we show that the LLA algorithm is relatively insensitive to initial value compared to the LQA algorithm, that is, it can converge to a desired value even if the initial value is given randomly. We explain the theoretical basis for this phenomenon based on the theorems of Zhang et al. (2016) who explained that the LLA algorithm asymptotically identifies the oracle estimator within a small number of iterations when the initial value is given as the estimate of

the L_1 SVM. With slight modifications, we can get the similar result at random initial values that we want to derive in this paper.

We begin with the basic set-up and notation. The population version of $l_n(\beta)$ (1) without the penalty term is

$$L(\boldsymbol{\beta}) = \mathbb{E}\{(1 - yX^T\boldsymbol{\beta})_+\}.$$

Let $\boldsymbol{\beta}^*$ denote the true parameter value which satisfies

$$\boldsymbol{\beta}^* = \arg\min_{\boldsymbol{\beta}} L(\boldsymbol{\beta}).$$

We assume that $\boldsymbol{\beta}^*$ is sparse and $\{1 \leq j \leq q; \beta_j^* \neq 0\}$ is the index set of the non-zero coefficients of $\boldsymbol{\beta}^*$. Without loss of generality, we let the last $p - q$ components of $\boldsymbol{\beta}^*$ are 0, that is $\boldsymbol{\beta}^* = (\boldsymbol{\beta}_1^{*T}, \mathbf{0}^T)^T$. Correspondingly, we write $X^T = (Z^T, R^T)$ where $Z = (1, x_1, \cdots, x_q)^T$ and $R = (x_{q+1}, \cdots, x_p)^T$. Also, we define

$$S(\boldsymbol{\beta}) = -\mathbb{E}\{I(1 - yX^T\boldsymbol{\beta} \geq 0)yX\}$$

and

$$H(\boldsymbol{\beta}) = \mathbb{E}\{\delta(1 - yX^T\boldsymbol{\beta})XX^T\}$$

where $I(\cdot)$ and $\delta(\cdot)$ denotes the indicator function and the Dirac delta function. If well defined, it can be shown that $S(\boldsymbol{\beta})$ and $H(\boldsymbol{\beta})$ are considered to be the gradient vector and Hessian matrix of $L(\boldsymbol{\beta})$, respectively (Koo et al., 2008). Finally, the oracle estimator is defined as $\hat{\boldsymbol{\beta}} = (\hat{\boldsymbol{\beta}_1^T}, \mathbf{0}^T)^T$ where

$$\hat{\boldsymbol{\beta}_1} = \arg\min_{\boldsymbol{\beta}_1} \left[\frac{1}{n} \sum_{i=1}^{n} (1 - y_i Z_i^T \boldsymbol{\beta}_1)_+ \right].$$

We note that non-uniqueness of the above minimizer is not essential here. When the minimizer is not unique, the theoretical results still hold for any particular minimizer (Zhang et al., 2016).

Also, we assume the following conditions.

Condition 1. The conditional densities of X given $y = 1$ and $y = -1$, denoted by f and g in this chapter, are continuous and have finite second moments.

Condition 2. There exists $B(X_0, \delta_0)$, a ball centered at X_0 with radius $\delta_0 > 0$ such that $f(X)$ and $g(X)$ are bounded away from zero on $B(X_0, \delta_0)$.

Condition 3. $\boldsymbol{\beta}_1^*$ is not zero.

Condition 4. $H(\boldsymbol{\beta})$ is positive-definite around $\boldsymbol{\beta}^*$.

Condition 1 ensures that $H(\boldsymbol{\beta})$ is well-defined and continuous in $\boldsymbol{\beta}$. Condition 2 guarantees that the classification problem is non-separable, which implies the oracle estimator is unique. The conditions ensure that $S(\boldsymbol{\beta})$ and $H(\boldsymbol{\beta})$ are a well-defined gradient vector and Hessian matrix. For more detailed discussions, see Koo et al. (2008).

What we want to show is that under appropriate assumption, with a high probability, the LLA algorithm converges to the oracle estimator $\hat{\boldsymbol{\beta}}$ only three times for any initial value. Zhang et al. (2016) considered the case where both n and p increases, but in this paper, we briefly assume the case where n increases with fixed p.

Under the above conditions, the following lemmas can be shown.

Lemma 1 *Consider the objective function of the SVM with the weighted L_1 penalty which is*

$$\tilde{l}_n(\boldsymbol{\beta}) = \frac{1}{n}\sum_{i=1}^{n}(1 - y_i X_i^T \boldsymbol{\beta})_+ + \|D_n \boldsymbol{\beta}\|_1$$

where D_n is any $p \times p$ diagonal matrix whose elements are in $\{p'_{\lambda_n}(\beta); \beta \in \mathcal{R}^+\}$. If $\lambda_n = o(n^{-\frac{1}{2}})$, then $\hat{\boldsymbol{\beta}}^{L_1} = \arg\min_{\boldsymbol{\beta}} \tilde{l}_n(\boldsymbol{\beta})$ satisfies that

$$P(|\hat{\beta}_j^{L_1} - \beta_j^*| > \lambda_n \quad \textit{for some} \quad 0 \le j \le p) \to 0$$

as $n \to \infty$.

Lemma 2 *For the true parameter $\boldsymbol{\beta}^* = (\boldsymbol{\beta}_1^{*T}, \mathbf{0}^T)^T$, the oracle estimator $\hat{\boldsymbol{\beta}} = (\hat{\boldsymbol{\beta}_1^T}, \mathbf{0}^T)^T$ satisfies*

$$\|\hat{\boldsymbol{\beta}}_1 - \boldsymbol{\beta}_1^*\| = O_p(n^{-\frac{1}{2}}).$$

Lemma 3 *Consider the subgradient of the hinge loss term of $l_n(\beta)$ (1) which is the collection of vectors $\boldsymbol{s}(\boldsymbol{\beta}) = (s_0(\boldsymbol{\beta}), \cdots, s_p(\boldsymbol{\beta}))^T$ where*

$$s_j(\boldsymbol{\beta}) = \Bigg\{ -\frac{1}{n}\sum_{i=1}^{n} I(1 - y_i X_i^T \boldsymbol{\beta} \ge 0) y_i x_{ij} - \frac{1}{n}\sum_{i=1}^{n} y_i x_{ij} v_i \quad ; \\ -1 \le v_i \le 0 \quad \textit{if} \quad y_i X_i^T \boldsymbol{\beta} = 1, \quad \textit{and} \quad v_i = 0 \quad \textit{otherwise} \Bigg\}.$$

As $n \to \infty$, $\hat{\boldsymbol{\beta}}$ satisfies

$$P(\mathbf{0} \in \boldsymbol{s}(\hat{\boldsymbol{\beta}})) \to 1.$$

Also, under Conditions 1-4, the following theorem which indicates the good convergence of the LLA algorithm for any initial value can be shown.

Theorem 2 *Consider the following events:*

$$F_{n1} = \{|\hat{\beta}_j^{L_1} - \beta_j^*| > \lambda_n \ \ \textit{for some} \ \ 0 \le j \le p\};$$
$$F_{n2} = \{|\beta_j^*| < (a+1)\lambda_n \ \ \textit{for some} \ \ 0 \le j \le q\};$$
$$F_{n3} = \{\textit{All } s \in s_j(\hat{\boldsymbol{\beta}}) \textit{ satisfy that}$$
$$|s| > \lambda_n \ \ \textit{for some} \ \ q+1 \le j \le p \ \ \textit{or} \ \ s \neq 0 \ \ \textit{for some} \ \ 0 \le j \le q\};$$
$$F_{n4} = \{|\hat{\beta}_j| < a\lambda_n \ \ \textit{for some} \ \ 0 \le j \le q\}.$$

Denote the corresponding probability as $P_{ni} = P(F_{ni})$, $i = 1, 2, 3, 4$. *Then, with probability at least* $1 - P_{n1} - P_{n2} - P_{n3} - P_{n4}$, *the LLA algorithm with any random initial value finds the oracle estimator* $\hat{\boldsymbol{\beta}}$ *after three iterations.*

Theorem 2 shows that when P_{ni}'s are small, the LLA algorithm converges to the oracle estimator regardless of initial value with a high probability. We expect P_{n1} to be small when $\hat{\boldsymbol{\beta}}^{L_1}$ is sufficiently close to the true parameter value $\boldsymbol{\beta}^*$. Also, P_{n2} and P_{n4} can be small when the non-zero elements of $\boldsymbol{\beta}^*$ and the corresponding elements of the oracle estimator $\hat{\boldsymbol{\beta}}$ have sufficiently large values. P_{n3} is small when the subgradient of the hinge loss term at the oracle estimator has a value sufficiently close to 0.

Favorably, the previous Lemma 1, 2 and 3 support the convergence of P_{ni}'s to 0 under appropriate conditions. First, Lemma 1 implies that if $\lambda_n = o(n^{-\frac{1}{2}})$, P_{n1} converges to 0 as $n \to \infty$. Also, since β_j^*'s are fixed and non-zero values by assumption, clearly P_{n2} converges to 0 as $n \to \infty$ if λ_n decreases to 0. In addition, Lemma 3 ensures that P_{n3} converges to 0 as $n \to \infty$. Finally, if $\lambda_n = o(n^{-\frac{1}{2}})$,

P_{n4} converges to 0 as $n \to \infty$ since

$$\begin{aligned} P_{n4} \;\leq\; & P(|\hat{\beta}_j| < a\lambda_n \;\text{ for some }\; 0 \leq j \leq q, \;\; |\hat{\beta}_j - \beta_j^*| \leq \lambda_n \;\text{ for all }\; 0 \leq j \leq q) \\ & + P(|\hat{\beta}_j - \beta_j^*| > \lambda_n \;\text{ for some }\; 0 \leq j \leq q) \\ \leq\; & P(|\beta_j^*| < (a+1)\lambda_n \;\text{ for some }\; 0 \leq j \leq q) \\ & + P(|\hat{\beta}_j - \beta_j^*| > \lambda_n \;\text{ for some }\; 0 \leq j \leq q) \longrightarrow 0 \;\text{ as }\; n \to \infty. \end{aligned}$$

The convergence of the first term is supported by the fact that P_{n2} converges to 0, and the convergence of the second term is guaranteed by Lemma 2.

In sum, Theorem 2 supports that with a certain probability, the LLA algorithm converges to a desired oracle estimator within 3 times regardless of initial value. In addition, Lemma 1, 2 and 3 uphold that the probability is close to 1 as n increases and λ becomes smaller at a speed faster than $n^{-\frac{1}{2}}$. In brief, these theories justify the LLA algorithm's insensitivity to initial value for large n and small λ. In the next chapter, we check this phenomenon through a simulation study.

Chapter 4

Simulation Study

In this chapter, we look through the results of a simulation study to see if the theories in the previous chapter are well applied to data. The arguments to be checked are as follows: First, the LLA algorithm is less sensitive to initial value than the LQA algorithm. Second, as Theorem 2, the LLA algorithm gives good results regardless of initial value as n increases and λ becomes smaller. In this paper, we consider the following data generation process with reference to Zhang et al. (2006):

$$p = 200,\ q = 2,\ P(y = 1) = P(y = -1) = 0.5,$$
$$x_1 = yN(3, 1) \ \text{ and } \ x_2 = yN(0, 1) \ \text{ with probability } \ 0.7,$$
$$x_1 = yN(0, 1) \ \text{ and } \ x_2 = yN(3, 1) \ \text{ with probability } \ 0.3,$$
$$x_j \ \text{ are independently generated from } \ N(0, 20) \ \text{ for } \ j = 3, \cdots, 200.$$

We dealt with three settings for the sample size: $n = 25, 50, 100$. For each case, we run the algorithms on the set of λ of $\{e^{-7}, e^{-4}, e^{-1}, e^{2}\}$. A random value between -100 and 100, the result from the standard SVM, and the result from the L_1 SVM are considered as the initial values. We use $a = 3.7$ for the SCAD penalty.

After estimating the coefficients by the LQA and LLA algorithms, three indicators were calculated from the obtained estimates. 'Signal' is the number of selected relevant variables and 'Noise' is the number of selected irrelevant variables. That is, signal has a maximum value of $q = 2$ and is better as close to $q = 2$, and noise has a maximum value of $p - q = 198$, and is better as close to 0. Finally, a prediction error ($= \frac{FP+FN}{TP+TN+FP+FN}$) was calculated for a test set with $n = 100$. A total of 30 simulations were performed to calculate the mean and variance of each indicator. Additionally, for each signal and noise, we calculated the difference between the maximum value and the minimum value among the mean values obtained from the three types of initial values, which is denoted by 'Max difference'.

Table 4.1, 4.2 and 4.3 show the results of signal, noise, and prediction error, respectively. Numbers outside the parentheses in the tables indicate the mean, and numbers in parentheses indicate the standard deviation. The results from λ giving the best results at each initial value are shown in bold.

First, we examine the argument that the LLA algorithm is less sensitive to initial value than the LQA algorithm. In the case of signal in Table 4.1, the values of max difference of the LLA algorithm is smaller than those of the LQA algorithm in all cases except for one case where $n = 50$ and $\lambda = e^{-4}$. In the case of noise in Table 4.2, the LLA algorithm has a smaller max difference value than the LQA algorithm in all cases. This shows that the initial values in the LLA

	λ	LLA				LQA			
		e^{-7}	e^{-4}	e^{-1}	e^{2}	e^{-7}	e^{-4}	e^{-1}	e^{2}
n=25	random	0.100	0.100	**0.800**	0.000	**2.000**	**2.000**	1.967	0.867
		(0.305)	(0.305)	(0.484)	(0.000)	(0.000)	(0.000)	(0.183)	(0.571)
	l_2	0.167	**1.200**	0.800	0.000	**2.000**	1.567	0.800	0.000
		(0.461)	(0.484)	(0.484)	(0.000)	(0.000)	(0.568)	(0.484)	(0.000)
	l_1	0.500	0.567	**0.800**	0.000	0.133	0.167	0.433	**1.100**
		(0.682)	(0.728)	(0.484)	(0.000)	(0.346)	(0.379)	(0.626)	(0.481)
max difference		0.400	1.100	0.000	0.000	1.867	1.833	1.534	1.100
n=50	random	0.333	0.367	**1.800**	0.033	**2.000**	**2.000**	1.933	1.333
		(0.606)	(0.615)	(0.407)	(0.183)	(0.000)	(0.000)	(0.254)	(0.758)
	l_2	0.500	**1.833**	1.800	0.000	**2.000**	**2.000**	1.800	0.000
		(0.630)	(0.379)	(0.407)	(0.000)	(0.000)	(0.000)	(0.407)	(0.000)
	l_1	1.667	**1.900**	1.800	0.000	0.633	0.700	1.067	**1.200**
		(0.479)	(0.305)	(0.407)	(0.000)	(0.615)	(0.466)	(0.365)	(0.407)
max difference		1.334	1.533	0.000	0.033	1.367	1.300	0.866	1.333
n=100	random	1.500	1.500	**2.000**	0.067	**2.000**	**2.000**	1.900	1.333
		(0.572)	(0.572)	(0.000)	(0.254)	(0.000)	(0.000)	(0.305)	(0.802)
	l_2	1.500	**2.000**	**2.000**	0.000	**2.000**	**2.000**	**2.000**	0.000
		(0.572)	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)	(0.000)
	l_1	**2.000**	**2.000**	**2.000**	0.000	1.033	1.067	1.100	**1.167**
		(0.000)	(0.000)	(0.000)	(0.000)	(0.183)	(0.254)	(0.305)	(0.379)
max difference		0.500	0.500	0.000	0.067	0.967	0.933	0.900	1.333

Table 4.1 Mean and Standard Deviation of Signal

algorithm has less effect on the results. Also, in the case of prediction error in Table 4.3, the LQA algorithm has big prediction error values of 0.3 or more for random initial value and the L_1 SVM initial value, in all cases of n and λ. Only when the initial value is given as a result of the standard SVM, it has small prediction error values less than 0.2 in the appropriate λ. On the other hand, the LLA algorithm has prediction error values less than 0.2 for all initial values in the appropriate λ when n is 50 and 100. These results also show that the LLA algorithm is more insensitive to the initial value than the LQA algorithm.

Next, we examine the argument that as n increases and λ becomes smaller, the LLA algorithm gives good results regardless of initial value. In the case of signal in Table 4.1, it is shown that no λ gives good results for all three initial

		LLA				LQA			
λ		e^{-7}	e^{-4}	e^{-1}	e^{2}	e^{-7}	e^{-4}	e^{-1}	e^{2}
n=25	random	15.367	8.733	19.633	**0.700**	198.000	197.567	184.967	**41.400**
		(3.774)	(2.434)	(1.732)	(1.622)	(0.000)	(0.679)	(3.479)	(13.048)
	l_2	15.133	1.333	19.633	**0.200**	167.167	6.500	19.800	**0.000**
		(4.125)	(1.093)	(1.732)	(1.095)	(5.059)	(2.418)	(2.497)	(0.000)
	l_1	9.900	3.100	19.633	**0.000**	**6.867**	7.133	9.800	34.000
		(4.656)	(2.090)	(1.732)	(0.000)	(2.556)	(2.113)	(3.478)	(3.384)
max difference		5.467	7.400	0.000	0.700	191.133	191.067	175.167	41.400
n=50	random	26.633	13.933	32.133	**0.300**	197.933	190.900	188.267	**58.233**
		(7.411)	(5.085)	(2.623)	(1.208)	(0.254)	(35.875)	(3.118)	(18.860)
	l_2	26.167	1.167	31.933	**0.000**	175.700	8.167	30.867	**0.000**
		(7.226)	(1.341)	(2.778)	(0.000)	(4.380)	(2.574)	(3.693)	(0.000)
	l_1	13.100	0.600	31.933	**0.000**	**10.033**	10.800	15.400	52.800
		(6.294)	(1.163)	(2.778)	(0.000)	(4.853)	(4.923)	(4.523)	(4.180)
max difference		13.533	13.333	0.200	0.300	187.900	182.733	172.867	58.233
n=100	random	42.667	9.733	28.967	**4.467**	198.000	197.600	190.700	**82.467**
		(15.932)	(10.395)	(4.367)	(17.071)	(0.000)	(0.563)	(3.354)	(10.592)
	l_2	36.400	1.800	28.967	**0.000**	178.833	9.100	27.700	**0.000**
		(13.637)	(1.126)	(4.367)	(0.000)	(3.228)	(4.498)	(4.921)	(0.000)
	l_1	16.767	0.667	28.967	**0.000**	**12.567**	13.400	15.800	59.633
		(6.420)	(0.922)	(4.367)	(0.000)	(6.345)	(6.775)	(5.182)	(6.100)
max difference		25.900	9.066	0.000	4.467	185.433	188.500	174.900	82.467

Table 4.2 Mean and Standard Deviation of Noise

values when $n = 25$, and that $\lambda = e^{-1}$ gives good results for all initial values when $n = 50$. Also, λ's smaller than e^{-1} give good signal values close to 2 for all initial values when $n = 100$. In the case of noise in Table 4.2, the overall result is good only if λ is e^{-4} and e^{2}. Combining this with the results of signal, we can ensure that the λ value of e^{-4} is best for all n cases, and the difference among the results of three initial values becomes smaller as n increases at this value of λ. This implies that as n increases, the insensitivity of the LLA algorithm to initial value increases as well. Finally, Table 4.3 shows that the LLA algorithm does not give uniformly good results for initial values in all cases of λ when $n = 25$, and that it gives uniformly good results for initial values only when λ is e^{-1} for $n = 50$. Also, the algorithm gives uniformly good results for initial

	λ	LLA				LQA			
		e^{-7}	e^{-4}	e^{-1}	e^{2}	e^{-7}	e^{-4}	e^{-1}	e^{2}
n=25	random	0.493	0.499	**0.374**	0.500	0.463	0.464	**0.463**	0.465
		(0.082)	(0.072)	(0.120)	(0.000)	(0.055)	(0.056)	(0.051)	(0.053)
	l_2	0.482	**0.187**	0.374	0.500	0.465	**0.200**	0.374	0.500
		(0.082)	(0.126)	(0.120)	(0.000)	(0.050)	(0.111)	(0.115)	(0.000)
	l_1	0.402	**0.350**	0.374	0.500	0.480	0.469	0.452	**0.451**
		(0.147)	(0.197)	(0.120)	(0.000)	(0.078)	(0.078)	(0.083)	(0.071)
n=50	random	0.451	0.434	**0.168**	0.500	0.452	**0.438**	0.456	0.457
		(0.126)	(0.148)	(0.073)	(0.000)	(0.049)	(0.092)	(0.058)	(0.057)
	l_2	0.429	**0.072**	0.164	0.500	0.442	**0.116**	0.163	0.500
		(0.132)	(0.082)	(0.077)	(0.000)	(0.051)	(0.060)	(0.076)	(0.000)
	l_1	0.155	**0.055**	0.164	0.500	0.480	0.440	**0.375**	0.427
		(0.121)	(0.076)	(0.077)	(0.000)	(0.058)	(0.077)	(0.055)	(0.052)
n=100	random	0.253	0.162	**0.042**	0.495	**0.428**	**0.428**	0.429	0.441
		(0.133)	(0.157)	(0.018)	(0.022)	(0.034)	(0.034)	(0.039)	(0.061)
	l_2	0.268	**0.036**	0.042	0.500	0.404	0.085	**0.044**	0.500
		(0.149)	(0.026)	(0.018)	(0.000)	(0.045)	(0.047)	(0.020)	(0.000)
	l_1	0.061	**0.028**	0.042	0.500	0.417	0.392	**0.342**	0.366
		(0.027)	(0.013)	(0.018)	(0.000)	(0.074)	(0.070)	(0.039)	(0.046)

Table 4.3 Mean and Standard Deviation of Prediction Error

values even when λ is smaller than e^{-1} for $n = 100$. In sum, it can be seen that as n increases, the LLA algorithm becomes insensitive to initial values, and the results get better for small λ.

Chapter 5

Conclusion

In this paper, we have summarized two local approximation methods for solving the optimization problem of SCAD SVM, which are the LQA and LLA algorithms. First, although the LQA algorithm gave good results in some studies (Zhang et al., 2006), it has the disadvantages of additional tuning parameters, a characteristic of backward selection, the need for approximation of both loss and penalty functions, and sensitivity to initial value. The LLA algorithm, on the other hand, is better than the LQA algorithm because it does not need to specify additional tuning parameters, is not a kind of backward selection, does not need approximation of hinge loss function, and is relatively insensitive to initial value. We have further supported the fact that the LLA algorithm is relatively insensitive to initial value through theorems and a simulation study. In conclusion, the LLA algorithm is more recommended than the LQA algorithm.

The convergence rate of the LLA algorithm was, however, slower than the LQA algorithm in the simulation, and it becomes much slower as the sample

size increases. In this regard, we might be able to make the LLA algorithm faster by applying a parallel processing method to solve its linear programming problem. Another thing we can study further is to check if the insensitivity theory for initial values can be applied as well in the setting where p is not fixed and increases as n increases. Perhaps more assumptions and conditions would be needed.

Bibliography

Bradley, P. S. and Mangasarian, O. L. (1998). Feature selection via concave minimization and support vector machines. In *ICML*, volume 98, pages 82–90.

Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine learning*, 20(3):273–297.

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

Friedman, J., Hastie, T., and Tibshirani, R. (2001). *The elements of statistical learning*, volume 1. Springer series in statistics New York.

Guyon, I., Weston, J., Barnhill, S., and Vapnik, V. (2002). Gene selection for cancer classification using support vector machines. *Machine learning*, 46(1):389–422.

Koo, J.-Y., Lee, Y., Kim, Y., and Park, C. (2008). A bahadur representation of the linear support vector machine. *Journal of Machine Learning Research*, 9(Jul):1343–1368.

Park, C., Kim, K.-R., Myung, R., and Koo, J.-Y. (2012). Oracle properties of scad-penalized support vector machine. *Journal of Statistical Planning and Inference*, 142(8):2257–2270.

Polson, N. G. and Scott, J. G. (2016). Mixtures, envelopes and hierarchical duality. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 78(4):701–727.

Wang, L., Zhu, J., and Zou, H. (2006). The doubly regularized support vector machine. *Statistica Sinica*, pages 589–615.

Zhang, H. H., Ahn, J., Lin, X., and Park, C. (2006). Gene selection using support vector machines with non-convex penalty. *Bioinformatics*, 22(1):88–95.

Zhang, X., Wu, Y., Wang, L., and Li, R. (2016). Variable selection for support vector machines in moderately high dimensions. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 78(1):53–76.

Zou, H. (2007). An improved 1-norm svm for simultaneous classification and variable selection. In *Artificial Intelligence and Statistics*, pages 675–681.

Zou, H. and Li, R. (2008). One-step sparse estimates in nonconcave penalized likelihood models. *Annals of statistics*, 36(4):1509.

Appendix A

Proofs

Proof of Theorem 1

Consider $\phi^*(\gamma) = \inf_{x\geq 0}\{\gamma x - \phi(x)\}$ when $\gamma < 0$. If $x = \infty$, then $\gamma x - \phi(x) = -\infty$. Therefore, when $x \in \mathcal{R}^+$,

$$\phi(x) = \phi^{**}(x) \coloneqq \inf_{\gamma\in\mathcal{R}}\{\gamma x - \phi^*(\gamma)\} = \inf_{\gamma\geq 0}\{\gamma x - \phi^*(\gamma)\}.$$

Since $\phi(x)$ is symmetric, $\phi(x) = \inf_{\gamma\geq 0}\{\gamma|x| - \phi^*(\gamma)\}$.

Also, $\hat{\lambda} = \arg\min_{\lambda\geq 0}\{\lambda|x| - \phi^*(\lambda)\} \Leftrightarrow \hat{\lambda}|x| - \phi^*(\hat{\lambda}) = \phi(|x|)\ (= \phi(x))$
$\Leftrightarrow \hat{\lambda}|x| - \phi(|x|) = \inf_{x\geq 0}\{\hat{\lambda}x - \phi(x)\} \Leftrightarrow \hat{\lambda}|x| - \phi(|x|) \leq \hat{\lambda}y - \phi(y)$ for $\forall y \geq 0$
$\Leftrightarrow \phi(y) \leq \phi(|x|) + \hat{\lambda}(y - |x|)$ for $\forall y \geq 0 \Leftrightarrow \hat{\lambda} \in \partial\phi(|x|)$.

The proof of Lemma 1 relies on the following Lemma 4.

Lemma 4 *Let* $f_n(\boldsymbol{\beta}) = \frac{1}{n}\sum_{i=1}^n (1 - y_i X_i^T\boldsymbol{\beta})_+$ *and* $\Lambda_n(\boldsymbol{\theta}) = n\{f_n(\boldsymbol{\beta}^* + \frac{1}{\sqrt{n}}\boldsymbol{\theta}) -$

$f_n(\boldsymbol{\beta}^*)\}$. *Then, for* $^\forall\boldsymbol{\theta} \in \mathcal{R}^p$ *and some* $A_n = O_p(1)$,

$$\Lambda_n(\boldsymbol{\theta}) = \frac{1}{2}\boldsymbol{\theta}^T H(\boldsymbol{\beta}^*)\boldsymbol{\theta} + A_n.$$

Proof of Lemma 4

First, by the Taylor series expansion of $L(\boldsymbol{\beta}) = \mathbb{E}\{(1 - yX^T\boldsymbol{\beta})_+\}$ around $\boldsymbol{\beta}^*$,

$$\mathbb{E}[\Lambda_n(\boldsymbol{\theta})] = \frac{1}{2}\boldsymbol{\theta}^T H(\tilde{\boldsymbol{\beta}})\boldsymbol{\theta} \tag{4a}$$

where $\tilde{\boldsymbol{\beta}} = \boldsymbol{\beta}^* + \frac{t}{\sqrt{n}}\boldsymbol{\theta}$ for some $0 \le t \le 1$.

Now, we want to show that

$$\frac{1}{2}\boldsymbol{\theta}^T H(\tilde{\boldsymbol{\beta}})\boldsymbol{\theta} = \frac{1}{2}\boldsymbol{\theta}^T H(\boldsymbol{\beta}^*)\boldsymbol{\theta} + o(1). \tag{4b}$$

Define $D_{jk}(\boldsymbol{\alpha}) = H(\boldsymbol{\beta}^* + \boldsymbol{\alpha})_{jk} - H(\boldsymbol{\beta}^*)_{jk}$ for $0 \le j, k \le p$. Since $H(\boldsymbol{\beta})$ is continuous by *Condition 1*, for $^\forall\varepsilon > 0$ and $^\forall j, k \in \{0, \cdots, p\}$,

$$^\exists\delta > 0 \ \text{ s.t. } \ \|\boldsymbol{\alpha}\| < \delta \Rightarrow |D_{jk}(\boldsymbol{\alpha})| < \varepsilon.$$

Then, for sufficiently large n satisfying $\|\frac{t}{\sqrt{n}}\boldsymbol{\theta}\| < \delta$,

$$|\boldsymbol{\theta}^T\{H(\tilde{\boldsymbol{\beta}}) - H(\boldsymbol{\beta}^*)\}\boldsymbol{\theta}| \ \le \ \sum_{j,k}|\theta_j||\theta_k|\Big|D_{jk}\Big(\frac{t}{\sqrt{n}}\boldsymbol{\theta}\Big)\Big| \ < \ \varepsilon\sum_{j,k}|\theta_j||\theta_k| \ \le \ 2\varepsilon\|\boldsymbol{\theta}\|^2.$$

Therefore, (4b) holds.

Next, define W_n and $R_{in}(\boldsymbol{\theta})$ as

$$W_n = -\sum_{i=1}^n I(y_i X_i^T\boldsymbol{\beta}^* \le 1)y_i X_i$$

and

$$R_{in}(\boldsymbol{\theta}) = \left[1 - y_i X_i^T \left(\boldsymbol{\beta}^* + \frac{1}{\sqrt{n}}\boldsymbol{\theta}\right)\right]_+ - [1 - y_i X_i^T \boldsymbol{\beta}^*]_+ + \frac{1}{\sqrt{n}} I(y_i X_i^T \boldsymbol{\beta}^* \leq 1) y_i X_i^T \boldsymbol{\theta}.$$

Then,

$$\begin{aligned}\Lambda_n(\boldsymbol{\theta}) &= \sum_{i=1}^n R_{in}(\boldsymbol{\theta}) + \frac{1}{\sqrt{n}} W_n^T \boldsymbol{\theta} \\ &= \mathbb{E}[\Lambda_n(\boldsymbol{\theta})] + \frac{1}{\sqrt{n}} W_n^T \boldsymbol{\theta} + \sum_{i=1}^n \{R_{in}(\boldsymbol{\theta}) - \mathbb{E}[R_{in}(\boldsymbol{\theta})]\} \qquad (4c)\end{aligned}$$

since $\mathbb{E}[I(y_i X_i^T \boldsymbol{\beta}^* \leq 1) y_i X_i] = -S(\boldsymbol{\beta}^*) = 0$.

Now, we want to show that

$$\sum_{i=1}^n \{R_{in}(\boldsymbol{\theta}) - \mathbb{E}[R_{in}(\boldsymbol{\theta})]\} = o_p(1). \qquad (4d)$$

Define

$$R = [1 - z]_+ - [1 - a]_+ + I(a \leq 1)(z - a).$$

If $a > 1$, $R = (1 - z) I(z \leq 1)$, and otherwise, $R = (z - 1) I(z > 1)$. Hence,

$$\begin{aligned}R &= (1 - z) I(z \leq 1, a > 1) + (z - 1) I(z > 1, a \leq 1) \\ &\leq |z - a| I(z \leq 1, a > 1) + |z - a| I(z > 1, a \leq 1) \\ &\leq |z - a| I(|1 - a| \leq |z - a|).\end{aligned}$$

Therefore,

$$\begin{aligned}|R_{in}(\boldsymbol{\theta})| &\leq \frac{1}{\sqrt{n}} |y_i X_i^T \boldsymbol{\theta}| I\left(|1 - y_i X_i^T \boldsymbol{\beta}^*| \leq \frac{1}{\sqrt{n}} |y_i X_i^T \boldsymbol{\theta}|\right) \\ &= \frac{1}{\sqrt{n}} |X_i^T \boldsymbol{\theta}| I\left(|1 - y_i X_i^T \boldsymbol{\beta}^*| \leq \frac{1}{\sqrt{n}} |X_i^T \boldsymbol{\theta}|\right).\end{aligned}$$

Now, for $^{\forall}\varepsilon > 0$,

$$\begin{aligned}
\sum_{i=1}^{n} \mathbb{E}\Big\{ \Big(R_{in}(\boldsymbol{\theta}) - \mathbb{E}[R_{in}(\boldsymbol{\theta})] \Big)^2 \Big\} &\leq \sum_{i=1}^{n} \mathbb{E}\{[R_{in}(\boldsymbol{\theta})]^2\} \\
&\leq \sum_{i=1}^{n} \mathbb{E}\Big\{ \frac{1}{n} \|X_i\|^2 \|\boldsymbol{\theta}\|^2 I\Big(|1 - y_i X_i^T \boldsymbol{\beta}^*| \leq \frac{1}{\sqrt{n}} \|X_i\| \|\boldsymbol{\theta}\| \Big) \Big\} \\
&\leq \sum_{i=1}^{n} \frac{\|\boldsymbol{\theta}\|^2}{n} \Big\{ \mathbb{E}\|X_i\|^2 I(\|X_i\| > C) \\
&\qquad + \mathbb{E}\|X_i\|^2 I\Big(|1 - y_i X_i^T \boldsymbol{\beta}^*| \leq \frac{1}{\sqrt{n}} \|X_i\| \|\boldsymbol{\theta}\|, \quad \|X_i\| \leq C \Big) \Big\} \\
&\leq \sum_{i=1}^{n} \frac{\|\boldsymbol{\theta}\|^2}{n} \Big\{ \mathbb{E}\|X_i\|^2 I(\|X_i\| > C) + C^2 P\Big(|1 - y_i X_i^T \boldsymbol{\beta}^*| \leq \frac{C}{\sqrt{n}} \|\boldsymbol{\theta}\| \Big) \Big\} \\
&\leq \frac{\|\boldsymbol{\theta}\|^2}{2} \varepsilon + \frac{1 - \|\boldsymbol{\theta}\|^2}{2} \varepsilon
\end{aligned}$$

for sufficiently large n. The second inequality comes from the Cauchy–Schwarz inequality, and the fifth inequality is supported by the fact that $\mathbb{E}\|X\|^2 < \infty$, which implies that there exists $C > 0$ such that $\mathbb{E}\|X\|^2 I(\|X\| > C) < \frac{\varepsilon}{2}$ for $^{\forall}\varepsilon > 0$, and the fact that the distribution of X is not degenerate, which implies that $\lim_{t \to 0} P(|1 - y_i X_i^T \boldsymbol{\beta}^*| \leq t) = 0$. This proves that

$$\sum_{i=1}^{n} \mathbb{E}\Big\{ \Big(R_{in}(\boldsymbol{\theta}) - \mathbb{E}[R_{in}(\boldsymbol{\theta})] \Big)^2 \Big\} \longrightarrow 0$$

as $n \to 0$. Therefore, (4d) holds.

Finally, note that $\mathbb{E}(W_1) = -S(\boldsymbol{\beta}^*) = 0$ and $G(\boldsymbol{\beta}^*) \coloneqq \mathbb{E}(W_1 W_1^T) = E\{I(y_1 X_1^T \boldsymbol{\beta}^* \leq 1) X_1 X_1^T\}$. By the central limit theorem, $\frac{1}{\sqrt{n}} W_n \xrightarrow{d} N(0, G(\boldsymbol{\beta}^*))$. Thus,

$$\frac{1}{\sqrt{n}} W_n^T \boldsymbol{\theta} = O_p(1). \tag{4e}$$

Combining (4a), (4b), (4c), (4d) and (4e), we have the desired result.

Proof of Lemma 1

Consider $\|\boldsymbol{\theta}\| = \Delta$ for any sufficiently large Δ. Note that

$$\left| \left\| D_n\left(\boldsymbol{\beta}^* + \frac{1}{\sqrt{n}}\boldsymbol{\theta}\right)\right\|_1 - \|D_n\boldsymbol{\beta}^*\|_1 \right| \leq \left\| \frac{1}{\sqrt{n}} D_n\boldsymbol{\theta} \right\|_1$$
$$= \frac{1}{\sqrt{n}} \sum_{j=1}^{p} |d_{jn}\theta_j| \leq \frac{1}{\sqrt{n}}\lambda_n \sum_{j=1}^{p} |\theta_j| = \frac{1}{\sqrt{n}}\lambda_n \|\boldsymbol{\theta}\|_1$$

where d_{jn}'s are diagonal elements of D_n. Since $\|\boldsymbol{\theta}\|_1$ is bounded, there exists $C > 0$ such that

$$n\left| \left\| D_n\left(\boldsymbol{\beta}^* + \frac{1}{\sqrt{n}}\boldsymbol{\theta}\right)\right\|_1 - \|D_n\boldsymbol{\beta}^*\|_1 \right| \leq \sqrt{n}\lambda_n C$$

and it converges to 0 as $n \to 0$ since $\lambda_n = o(n^{-\frac{1}{2}})$. Therefore,

$$n\left\{ \tilde{l}_n\left(\boldsymbol{\beta}^* + \frac{1}{\sqrt{n}}\boldsymbol{\theta}\right) - \tilde{l}_n(\boldsymbol{\beta}^*)\right\} = \Lambda_n(\boldsymbol{\theta}) + n\left\{ \left\| D_n\left(\boldsymbol{\beta}^* + \frac{1}{\sqrt{n}}\boldsymbol{\theta}\right)\right\|_1 - \|D_n\boldsymbol{\beta}^*\|_1 \right\}$$
$$= \Lambda_n(\boldsymbol{\theta}) + o(1).$$

Then by Lemma 4, $n\left\{ \tilde{l}_n\left(\boldsymbol{\beta}^* + \frac{1}{\sqrt{n}}\boldsymbol{\theta}\right) - \tilde{l}_n(\boldsymbol{\beta}^*)\right\} = \frac{1}{2}\boldsymbol{\theta}^T H(\boldsymbol{\beta}^*)\boldsymbol{\theta} + A_n$ for some $A_n = O_p(1)$.

Note that for $^\forall\varepsilon > 0$, $^\exists M > 0$ such that $P(|A_n| < M) \geq 1 - \varepsilon$ for sufficiently large n. Also, since $H(\boldsymbol{\beta}^*)$ is positive definite by *Condition 3*, there exists a sufficiently large Δ' such that $\frac{1}{2}\boldsymbol{\theta}^T H(\boldsymbol{\beta}^*)\boldsymbol{\theta} > M$ for $^\forall\boldsymbol{\theta}$ satisfying $\|\boldsymbol{\theta}\| = \Delta'$. Then, for $^\forall\varepsilon > 0$ and $^\forall\boldsymbol{\theta}$ satisfying $\|\boldsymbol{\theta}\| = \Delta'$,

$$P\left\{ A_n + \frac{1}{2}\boldsymbol{\theta}^T H(\boldsymbol{\beta}^*)\boldsymbol{\theta} > 0\right\} \geq 1 - \varepsilon$$
$$\Rightarrow \; P\left\{ \tilde{l}_n\left(\boldsymbol{\beta}^* + \frac{1}{\sqrt{n}}\boldsymbol{\theta}\right) - \tilde{l}_n(\boldsymbol{\beta}^*) > 0\right\} \geq 1 - \varepsilon$$

for sufficiently large n. Therefore, for $^{\forall}\varepsilon > 0$,

$$P\left\{ \inf_{\|\boldsymbol{\theta}\|=\Delta'} \tilde{l}_n\left(\boldsymbol{\beta}^* + \frac{1}{\sqrt{n}}\boldsymbol{\theta}\right) > \tilde{l}_n(\boldsymbol{\beta}^*) \right\} \geq 1 - \varepsilon$$

for sufficiently large n. Since $\tilde{l}_n(\boldsymbol{\beta})$ is convex, the minimizer $\hat{\boldsymbol{\beta}}^{L_1} = \arg\min_{\boldsymbol{\beta}} \tilde{l}_n(\boldsymbol{\beta})$ satisfies that for $^{\forall}\varepsilon > 0$,

$$P\left(\|\hat{\boldsymbol{\beta}}^{L_1} - \boldsymbol{\beta}^*\| \leq \frac{\Delta'}{\sqrt{n}} \right) \geq 1 - \varepsilon$$

for sufficiently large n. Then, since $\lambda_n = o(n^{-\frac{1}{2}})$, for $^{\forall}\varepsilon > 0$,

$$P\left(\|\hat{\boldsymbol{\beta}}^{L_1} - \boldsymbol{\beta}^*\| > \lambda_n \right) < \varepsilon$$

which implies that

$$P(|\hat{\beta}_j^{L_1} - \beta_j^*| > \lambda_n \ \text{ for some } \ 0 \leq j \leq p) < \varepsilon$$

for sufficiently large n.

Proof of Lemma 2

Since $\boldsymbol{\beta}^* = (\boldsymbol{\beta}_1^{*T}, \mathbf{0}^T)^T$ and $\boldsymbol{\beta}^* = \arg\min_{\boldsymbol{\beta}} L(\boldsymbol{\beta})$ for $L(\boldsymbol{\beta}) = \mathbb{E}\{(1 - yX^T\boldsymbol{\beta})_+\}$,

$$\boldsymbol{\beta}_1^* = \arg\min_{\boldsymbol{\beta}_1} \mathbb{E}\{(1 - yZ^T\boldsymbol{\beta}_1)_+\}.$$

Let $f_n'(\boldsymbol{\beta}_1) = \frac{1}{n}\sum_{i=1}^n (1 - y_i Z_i^T \boldsymbol{\beta}_1)_+$. By replacing X and $\boldsymbol{\beta}$ to Z and $\boldsymbol{\beta}_1$, we can derive the similar result to Lemma 1 and 4, that is, for $^{\forall}\varepsilon > 0$,

$$P\left\{ \inf_{\|\boldsymbol{\theta}\|=\Delta'} f_n'\left(\boldsymbol{\beta}_1^* + \frac{1}{\sqrt{n}}\boldsymbol{\theta}\right) > f_n'(\boldsymbol{\beta}_1^*) \right\} \geq 1 - \varepsilon$$

for sufficiently large n. Since $f_n'(\boldsymbol{\beta}_1)$ is convex and the oracle estimator is unique, $\hat{\boldsymbol{\beta}}_1 = \arg\min_{\boldsymbol{\beta}_1} f_n'(\boldsymbol{\beta}_1)$ satisfies that for $^\forall\varepsilon > 0$,

$$P\left(\|\hat{\boldsymbol{\beta}}_1 - \boldsymbol{\beta}_1^*\| \leq \frac{\Delta'}{\sqrt{n}} \right) \geq 1 - \varepsilon$$

for sufficiently large n. Thus, we have the desired result.

Proof of Lemma 3

Let $\tilde{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \left[\frac{1}{n} \sum_{i=1}^n (1 - y_i X_i^T \boldsymbol{\beta})_+ \right]$. Similarly to Lemma 2, it can be shown that $\|\tilde{\boldsymbol{\beta}} - \boldsymbol{\beta}^*\| = O_p(n^{-\frac{1}{2}})$. Combining with the result of Lemma 2,

$$\|\hat{\boldsymbol{\beta}} - \tilde{\boldsymbol{\beta}}\| = O_p(n^{-\frac{1}{2}}).$$

Also, note that $P(\mathbf{0} \in \boldsymbol{s}(\tilde{\boldsymbol{\beta}})) = 1$.

Now, consider $P(0 \notin s_j(\hat{\boldsymbol{\beta}}))$ for any $j = 0, \cdots, p$. We want to show that $P(0 \notin s_j(\hat{\boldsymbol{\beta}})) \to 0$ as $n \to \infty$.

$$\begin{aligned}
P(0 \notin s_j(\hat{\boldsymbol{\beta}})) &= P\{0 \notin s_j(\hat{\boldsymbol{\beta}}), (1 - y_i X_i^T \hat{\boldsymbol{\beta}})(1 - y_i X_i^T \tilde{\boldsymbol{\beta}}) > 0 \text{ for } ^\forall i = 1, \cdots, n\} \\
&\quad + P\{0 \notin s_j(\hat{\boldsymbol{\beta}}), (1 - y_i X_i^T \hat{\boldsymbol{\beta}})(1 - y_i X_i^T \tilde{\boldsymbol{\beta}}) \leq 0 \text{ for some } i = 1, \cdots, n\} \\
&\leq P(0 \notin s_j(\tilde{\boldsymbol{\beta}})) + P(|y_i X_i^T (\hat{\boldsymbol{\beta}} - \tilde{\boldsymbol{\beta}})| > \eta) \text{ for some } \eta \text{ and } i \\
&\leq 0 + P(\|X_i\| \|\hat{\boldsymbol{\beta}} - \tilde{\boldsymbol{\beta}}\| > \eta)
\end{aligned}$$

The second inequality holds because of two facts that $s_j(\hat{\boldsymbol{\beta}}) = s_j(\tilde{\boldsymbol{\beta}})$ when $(1 - y_i X_i^T \hat{\boldsymbol{\beta}})(1 - y_i X_i^T \tilde{\boldsymbol{\beta}}) > 0$ for $^\forall i = 1, \cdots, n$, and that if $0 \notin s_j(\hat{\boldsymbol{\beta}})$, $1 - y_i X_i^T \hat{\boldsymbol{\beta}} \neq 0$ for some $i = 1, \cdots, n$.

Since $\|\hat{\boldsymbol{\beta}} - \tilde{\boldsymbol{\beta}}\| = O_p(n^{-\frac{1}{2}})$ and $\mathbb{E}\|X_i\| < \infty$, it can be shown that for

$^{\forall}\varepsilon > 0$, $P(0 \notin s_j(\hat{\boldsymbol{\beta}})) < \varepsilon$ for sufficiently large n. Thus, we have the desired result.

Proof of Theorem 2

Let $\boldsymbol{\beta}^{(0)}$ be any random initial value. After one iteration of the LLA algorithm, the solution of the next iteration is

$$\boldsymbol{\beta}^{(1)} = \arg\min_{\boldsymbol{\beta}} \left[\frac{1}{n} \sum_{i=1}^{n} (1 - y_i X_i^T \boldsymbol{\beta})_+ + \sum_{j=1}^{p} p'_\lambda(|\beta_j^{(0)}|)|\beta_j| \right]$$

which is a kind of $\hat{\boldsymbol{\beta}}^{L_1}$.

Assume that none of the events F_{ni}'s is true. Then, from $(F_{n1})^C$ and $(F_{n2})^C$,

$$|\beta_j^{(1)}| = |\beta_j^{(1)} - \beta_j^*| \leq \lambda_n \quad \text{for} \quad q+1 \leq j \leq p \tag{2a}$$

and

$$|\beta_j^{(1)}| \geq |\beta_j^*| - |\beta_j^{(1)} - \beta_j^*| \geq (a+1)\lambda_n - \lambda_n = a\lambda_n \quad \text{for} \quad 1 \leq j \leq q. \tag{2b}$$

By (2b), $p'_\lambda(|\beta_j^{(1)}|) = 0$ for $1 \leq j \leq q$. Therefore, the solution of the next iteration from $\boldsymbol{\beta}^{(1)}$ is

$$\boldsymbol{\beta}^{(2)} = \arg\min_{\boldsymbol{\beta}} \left[\frac{1}{n} \sum_{i=1}^{n} (1 - y_i X_i^T \boldsymbol{\beta})_+ + \sum_{j=q+1}^{p} p'_\lambda(|\beta_j^{(1)}|)|\beta_j| \right]. \tag{2c}$$

Also, from $(F_{n3})^C$,

$$|s_j(\hat{\boldsymbol{\beta}})| \leq \lambda_n \quad \text{for} \quad q+1 \leq j \leq p \tag{2d}$$

and

$$|s_j(\hat{\boldsymbol{\beta}})| = 0 \quad \text{for} \quad 0 \le j \le q. \tag{2e}$$

Then, by (2e) and the supporting hyperplane inequality,

$$\begin{aligned}\frac{1}{n}\sum_{i=1}^{n}(1 - y_i X_i^T \boldsymbol{\beta})_+ &\ge \frac{1}{n}\sum_{i=1}^{n}(1 - y_i X_i^T \hat{\boldsymbol{\beta}})_+ + \sum_{j=0}^{p} s_j(\hat{\boldsymbol{\beta}})(\beta_j - \hat{\beta}_j) \\ &= \frac{1}{n}\sum_{i=1}^{n}(1 - y_i X_i^T \hat{\boldsymbol{\beta}})_+ + \sum_{j=q+1}^{p} s_j(\hat{\boldsymbol{\beta}})(\beta_j - \hat{\beta}_j). \end{aligned} \tag{2f}$$

Hence,

$$\begin{aligned}&\left\{\frac{1}{n}\sum_{i=1}^{n}(1 - y_i X_i^T \boldsymbol{\beta})_+ + \sum_{j=q+1}^{p} p'_\lambda(|\beta_j^{(1)}|)|\beta_j|\right\} \\ &\qquad - \left\{\frac{1}{n}\sum_{i=1}^{n}(1 - y_i X_i^T \hat{\boldsymbol{\beta}})_+ + \sum_{j=q+1}^{p} p'_\lambda(|\beta_j^{(1)}|)|\hat{\beta}_j|\right\} \\ \ge \sum_{j=q+1}^{p} &\left\{s_j(\hat{\boldsymbol{\beta}})\beta_j + p'_\lambda(|\beta_j^{(1)}|)|\beta_j|\right\} = \sum_{j=q+1}^{p} \left\{s_j(\hat{\boldsymbol{\beta}})\mathrm{sgn}(\beta_j) + \lambda_n\right\}|\beta_j| \;\ge\; 0.\end{aligned}$$

The first inequality is due to (2f) and the fact that $\hat{\beta}_j = 0$ for $j = q+1, \cdots, p$. The equality holds because $p'_\lambda(|\beta_j^{(1)}|) = \lambda_n$ by (2a). The second inequality holds because of (2d). Therefore, $\hat{\boldsymbol{\beta}}$ is the minimizer of the problem (2c), that is $\boldsymbol{\beta}^{(2)} = \hat{\boldsymbol{\beta}}$.

Now, $(F_{n4})^C$ implies that $|\hat{\beta}_j| \ge a\lambda_n$ for $1 \le j \le q$, so

$$p'_\lambda(|\hat{\beta}_j|) = 0 \quad \text{for} \quad 1 \le j \le q$$

and

$$p'_\lambda(|\hat{\beta}_j|) = p'_\lambda(0) = \lambda_n \quad \text{for} \quad q+1 \le j \le p.$$

Therefore, the solution of the next iteration from $\boldsymbol{\beta}^{(2)} = \hat{\boldsymbol{\beta}}$ is

$$\boldsymbol{\beta}^{(3)} = \arg\min_{\boldsymbol{\beta}} \left[\frac{1}{n} \sum_{i=1}^{n} (1 - y_i X_i^T \boldsymbol{\beta})_+ + \sum_{j=q+1}^{p} \lambda_n |\beta_j| \right].$$

Then, since $\left\{ \frac{1}{n} \sum_{i=1}^{n} (1 - y_i X_i^T \boldsymbol{\beta})_+ + \sum_{j=q+1}^{p} \lambda_n |\beta_j| \right\} - \left\{ \frac{1}{n} \sum_{i=1}^{n} (1 - y_i X_i^T \hat{\boldsymbol{\beta}})_+ + \sum_{j=q+1}^{p} \lambda_n |\hat{\beta}_j| \right\} \geq \sum_{j=q+1}^{p} \left\{ s_j(\hat{\boldsymbol{\beta}}) \mathrm{sgn}(\beta_j) + \lambda_n \right\} |\beta_j| \ \geq \ 0,$

$$\boldsymbol{\beta}^{(3)} = \boldsymbol{\beta}^{(2)} = \hat{\boldsymbol{\beta}}.$$

That is, the LLA algorithm finds the oracle estimator again and stops.

국문초록

지지벡터기계 모형은 이진분류문제를 푸는 데에 있어 강력한 도구이지만, 불필요한 변수들이 관여되는 경우 예측력에 악영향을 받을 수 있다. 이 문제를 해결하기 위해 몇 가지 변형된 지지벡터기계 모형들이 제안되어 왔고, 그 중 SCAD 벌점화 지지벡터 기계 모형이 효과적인 변수 선택을 해준다는 것이 증명되었다. 그러나 이 모형의 최적화 과정에는 목적 함수의 비볼록성과 여러 국소적 최소값들의 존재성 문제가 제기된다. 이 논문에서는 SCAD 지지벡터 기계 모형을 최적화하는 주된 방법인 국소 2차 근사 방법과 국소 1차 근사 방법에 대해 요약하고, 더 나아가 두 가지의 새로운 접근을 시도하였다. 우선, 각 알고리즘의 유도 과정에서 테일러 급수 전개를 이용한 기존의 방법 대신에 포락선을 이용한 방법을 적용하였는데, 이는 기존의 방법보다 더 일반화된 방법으로서 의미를 갖는다. 다음으로, 기존에 알려졌던 국소 2차 근사 방법의 한계점들과 그에 비한 국소 1차 근사 방법의 장점들에 더하여, 국소 1차 근사 방법의 최소값에 대한 둔감성을 주장하고 그에 대한 근거로서 국소 1차 근사 방법이 임의의 초기값에 대해서 오라클 추정량으로 수렴한다는 이론을 제시하였다. 마지막으로 시뮬레이션 연구를 통해 국소 2차 근사 방법보다 국소 1차 근사 방법이 임의의 초기값에 대해 더 좋은 결과를 준다는 것을 검증하였다.

주요어: 국소 근사 알고리즘, SCAD 벌점함수, 지지벡터 기계, 변수 선택, 초기값 설정

학번: 2016-20274