이학박사 학위논문

Weighted Mnet Penalty for Twice Differentiable Convex Losses on High Dimensions

고차원 자료에서 두 번 미분 가능한 볼록 손실 함수에 대한 WMnet 벌점화 방법 연구

2014년 8월

서울대학교 대학원

통계학과

김 주 유

Weighted Mnet Penalty for Twice Differentiable Convex Losses on High Dimensions

지도교수 김 용 대

이 논문을 이학박사 학위논문으로 제출함.

2014년 4월

서울대학교 대학원

통계학과

김 주 유

김주유의 이학박사 학위논문을 인준함.

2014년 6월

위 원 장 : 오 희 석 (인)

부 위원장 : 김 용 대 (인)

위 원 : 장 원 철 (인)

위 원 : 원 중 호 (인)

위 원 : 전 종 준 (인)

Weighted Mnet Penalty for Twice Differentiable Convex Losses on High Dimensions

By

Jooyoo Kim

A Thesis

Submitted in fulfillment of the requirements

for the degree of

Doctor of Philosophy

in Statistics

Department of Statistics

College of Natural Sciences

Seoul National University

August, 2014

Abstract

In many regression problems, covariates can be naturally correlated. Kim and Jeon [2014] proposed weighted Mnet penalty which is defined combination of weighted minimax concave penalty(MCP) and weighted ridge penalty. They showed that the weighted Mnet penalty is useful to squared loss when the covariates of correlations are highly correlated. They also point out that the weighted l_2 penalty is equivalent to the Laplacian penalty with certain weights and the weighted Mnet estimator has the oracle property to the squared loss under regular conditions. We extend the weighted Mnet estimator to twice differentiable convex losses. We showed that the weighted l_2 penalty to twice differentiable convex losses also can be equivalent to the Laplacian penalty with certain weights and the weighted Mnet has an oracle property on high dimensional model in the sense that it is equal to the oracle ridge estimator with high probability. By simulations and real data analysis, we show that the weighted Mnet penalty is a useful to the other competitors including the

elastic net, the Ment and the sparse Laplacian penalty.

Keywords: weighted Mnet penalty, twice differentiable losses, generalized linear model, oracle ridge

Student Number: 2009 − 30069

차 례

표 차례

그림 차례

제 1 장

Introduction

1.1 Overview

Variable selection is important on high dimensional models, which affect model interpretation and prediction accuracy. In high dimensional models, the sparse penalized method has received a great attention recently as an alternative method of classical subset selections. Traditional variable selection methods including stepwise and the best subset selection using information criterion such as Mallows' Cp by Mallows [1973], Akaike information criterion (AIC) by Akaike [1973] and Bayesian information criterion (BIC) by Schwarz [1978], which have various drawbacks; computationally intensive, unstable and difficult to draw sampling properties. See Breiman [1996]. In contrast, sparse

penalized methods give stable estimators with automatic variable selection and hence resulting estimators performs well in prediction. There are several sparse penalized methods such as least absolute shrinkage and selection operator (LASSO) by Tibshirani [1996], smoothly clipped absolute deviation(SCAD) by (Fan and Li [2001]) and minimax concave penalty(MCP) by Zhang [2010]. In addition, when the true model is sparse, sparse penalized methods have desirable large sample properties such as selection consistency, the oracle property and optimal convergence rate.

However, sparse penalized estimators may not be optimal when the covariates are highly correlated. For example, Zou and Hastie [2005] explained that when the group of covariates are highly correlated, solutions of sparse penalized methods usually select a covariate from the correlated group of covariates. Empirically, when the correlations of covariates are high, the ridge estimator outperforms sparse penalized methods. Moreover, not all the desirable large sample properties of sparse penalized methods may be valid when the correlations between covariates is high.

There are two alternative regularized methods to overcome these disadvantages of the standard sparse regularized methods. The first approach is to use a linear combination of sparse and non-sparse penalties. Zou and Hastie [2005] proposed the elastic net(Enet) penalty, which is a linear combination of l_1 and

l_2 penalties. Bondell and Reich [2008] proposed the octagonal shrinkage and clustering algorithm for regression, which uses a linear combination of the l_1 penalty and a pairwise l_∞ penalty. Wu et al. [2009] considered a linear combination of the l_1 and the l_∞ penalty. Huang et al. [2010] proposed the Mnet, which replaces the l_1 penalty in the Enet with the MCP. And Kim and Jeon [2014] proposed weighted Mnet penalty is a combination of weighted MCP and weighted l_2 penalties.

The second approach is to penalize pairwise differences of regression coefficients. Examples are the clustered LASSO by She [2010], group pursuit by Shen and Huang [2010] and the sparse Laplacian penalty by Huang et al. [2011] and generalized LASSO by Tibshirani [2011].

In this thesis, we extend the weighted Mnet penalty of Kim and Jeon [2014] to general convex loss functions including the generalized linear models. There is an interesting relation between the weighted l_2 penalty and the Laplacian penalty of Huang et al. [2011]. Kim and Jeon [2014] proved that the weighted l_2 penalty with a certain choice of weights becomes the Laplacian penalty of Huang et al. [2011] when the squared loss is used. This implies that the weighted l_2 penalty shares many desirable properties with the Laplacian penalty. An advantage of the weighted l_2 penalty over the Laplacian penalty is that the former is simpler to use because the penalty is always strictly convex.

Another advantage is that the weighted Mnet penaty reduces the bias induced by the l_2 penalty. As explained by Zou and Hastie [2005], the post-scaled l_2 penalized estimator converges to the univariate regression estimator. However, the univariate regression estimator may not be optimal when there are highly correlated covariates. In contrast, the post-scaled weighted l_2 penalized estimator converges to the optimal estimator. The weighted MCP is employed to improve variable selectivity. In this thesis, we show that these nice properties of the weighted Mnet penalty for the squared loss are preserved for general convex losses.

The contributions of this thesis are as follows.

- We show that the weighted l_2 penalty with certain weights is approximately equal to the Laplacian penalty for twice differentiable convex loss functions. Comparing to the Laplacian penalty, computation of weighted Mnet penalty is easy since the weighted Mnet penalty is always strictly convex.

- We prove that the weighted Mnet estimator with a twice differentiable convex loss has an oracle property under regularity conditions. This is the first one for general convex losses including the generalized linear model on high dimensional data. In addition, the regular conditions we give for the oracle property of the weighted Mnet are more intuitive and

simpler than those for the oracle property of the Mnet or sparse Laplacian penalties.

- In numerical studies, we show that the weighted Mnet estimator yields better prediction accuracies than the other competing penalized methods such as Mnet and Laplcian. In particular, in extremely highly correlated data, the weighted Mnet method outperforms the competitors significantly.

The losses considered in this thesis includes the logistic loss, exponential loss and poisson likelihood functions, which are twice differentiable convex losses. Therefore the results of this thesis are useful to various generalized linear models.

1.2 Outline of the thesis

This thesis is organized as follows. In chapter 2, we review the relevant penalized methods. In chapter 3, we study the theoretical properties of the weighted Mnet estimator for twice differentiable convex losses on high dimensions such as approximately equivalence to Laplacian penalty and the oracle property. In chapter 4, we discuss the optimization algorithm to estimate the weighted Mnet estimator. In chapter 5, we present the results of analysis of simulated

and real data sets. Concluding remarks follow in chapter 6, and all technical details are presented in Appendix.

제 2 장

Literature Review : Variable Selection Methods on High Dimensions

Variable selection is an important issue on high dimensional models, which affects model interpretation and prediction accuracy. Without variable selection, there are too many variables in the model which makes it difficult to interpret the model. Besides, it is well known that to keep too many variables in the model makes the prediction accuracy worse. Traditional variable selection methods for the regression model including stepwise and the best subset selection using information criterion such as Mallows' Cp by Mallows

[1973], AIC by Akaike [1973] and BIC by Schwarz [1978], which have various drawbacks such as computationally intensive, unstable and difficult to draw sampling properties. Details see Breiman [1996].

Sparse penalized methods for high dimensional models have received a great attention recently as an alternative method of subset selection methods. Examples are LASSO by Tibshirani [1996], SCAD by Fan and Li [2001] and MCP by Zhang [2010]. A sparse penalized method gives a stable estimator with automatic variable selection and thus the estimator performs well in prediction. In addition, when the true model is sparse, sparse penalized methods have desirable large sample properties such as selection consistency, the oracle property and optimal convergence rate.

However, the standard sparse regularization methods are not optimal when the correlations of covariates are large. Zou and Hastie [2005] pointed out that if there is a group of covariates that are highly correlated, the solutions of sparse regularization methods usually select a covariate among the group, and thus the prediction accuracy becomes poor. In addition, all of the desired large sample properties may not be valid when the correlations of covariates are very high.

We can use two methods to overcome these disadvantages of the standard sparse penalized methods. The first approach is to use a linear combination of

sparse and non-sparse penalties. The second approach is to penalize pairwise differences of regression coefficients.

For the first approach, Zou and Hastie [2005] proposed the elastic net (Enet) penalty, which is a linear combination of the l_1 and l_2 penalties. Bondell and Reich [2008] proposed the octagonal shrinkage and clustering algorithm for regression, which uses a linear combination of the l_1 penalty and a pairwise l_∞ penalty. Wu et al. [2009] considered a linear combination of the LASSO and the l_∞ penalty. Huang et al. [2010] proposed the Mnet, which replaces the l_1 penalty in the Enet with the MCP. Kim and Jeon [2014] proposed the *weighted Mnet* penalty, which is a linear combination of the weighted MCP and and the weighted l_2 penalty

Examples of the second approach are the clustered LASSO by She [2010], group pursuit by Shen and Huang [2010], the sparse Laplacian penalty by Huang et al. [2011] and the generalized LASSO(GLASSO) by Tibshirani [2011].

In this chapter, we review the several penalized methods. First, we review the standard sparse penalized methods in section 2.1. Second, we review the combinations of penalties which are devised for highly correlated covariates in section 2.2. In section 2.3 we review favorable properties of the weighted Mnet penalty with the squared loss.

2.1 Sparse regularization methods

Let $(y_1, \mathbf{x}_1), \ldots, (y_n, \mathbf{x}_n)$ be n pairs of observations where $\mathbf{x}_i \in \mathcal{X} \subset \mathbb{R}^p$ is a covariate vector and $y_i \in \mathcal{Y} \subset \mathbb{R}$ is a response variable. We assume that $(y_1, \mathbf{x}_1), \ldots, (y_n, \mathbf{x}_n)$ are independent copies of a random vector $(y, \mathbf{x})$. The objective of regression models is to find the regression coefficient vector, $\boldsymbol{\beta} \in \mathbb{R}^p$, which minimizes the prediction error evaluated by the expected loss $E[l(y; \mathbf{x}^T\beta)]$, where $l : \mathcal{Y} \times \mathbb{R} \to \mathbb{R}$ is a given loss function. Examples of the loss function are squared loss $l(y, \mathbf{x}^T\beta) = (y - \mathbf{x}^T\beta)^2$ and the logistic loss $l(y, \mathbf{x}^T\beta) = \log(1 + \exp(-y^T\mathbf{x}^T\beta)$ for the logistic regression model.

Unfortunately, the prediction error is not available since the underlying distribution of $(y, \mathbf{x})$ is unknown. An alternative technique to resolve this problem is to estimate $\boldsymbol{\beta}$ by minimizing the empirical expected loss $\mathcal{L}(\boldsymbol{\beta})$ defined by $\frac{1}{n}\sum_{i=1}^{n} l(y_i, \mathbf{x}_i T\beta)$. However, directly minimizing the empirical expected loss suffers from the so-called 'over-fitting', especially when p is large compared to n. To avoid over-fitting, we minimize the penalized empirical expected loss. That is, we estimate the regression coefficient $\boldsymbol{\beta}$ by minimizing the the penalized empirical expected loss given as

$$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} \Big\{ \mathcal{L}(\boldsymbol{\beta}) + \sum_{j=1}^{p} J_\lambda(\beta_j) \Big\},$$

where $J_\lambda(\cdot)$ is a penalty function and λ is a regularization parameter to control

the effect of the penalty to the estimator.

An earlier research of penalized methods is the ridge penalty Hoerl and Kennard [1970]. The ridge estimator is defined as

$$\hat{\boldsymbol{\beta}}^{Ridge} = \arg\min_{\boldsymbol{\beta}} \Big\{ \mathcal{L}(\boldsymbol{\beta}) + \lambda \sum_{j=1}^{p} \beta_j^2 \Big\},$$

where $\lambda \geq 0$ is a tunning parameter that shrinks the estimator toward 0. the larger the value of λ is, the greater the amount of shrinkage is.

Typically, the shrinkage reduces the variance while it increases the bias. Let $\hat{\boldsymbol{\beta}}$ is the ordinary estimator that minimizes $\mathcal{L}(\boldsymbol{\beta})$. It can be shown that $||\hat{\boldsymbol{\beta}}^{Ridge}||_2 \leq ||\hat{\boldsymbol{\beta}}||_2$, where $||\cdot||_2$ is L_2 norm. This phenomenon is called the shrinkage which is helpful when the variance of the ordinary estimator is large compare to the bias. When there are many correlated covariate, the ordinary estimator has a large variance due to the multicollinearity. Hence, the ridge estimator achieves better prediction performance by reducing variance significantly while introducing a small amount of bias. For high dimensional models, typically the ordinary estimator has a large variance and the shrinkage improves the performance of the estimator much.

But, the ridge regularization method has two critical drawbacks. First, the ridge estimator penalizes heavily the large coefficients that leads to serious biases to large coefficients. Second, the ridge estimator does not produce a sparse solution. That is, none of the estimated coefficients is zero and thus the

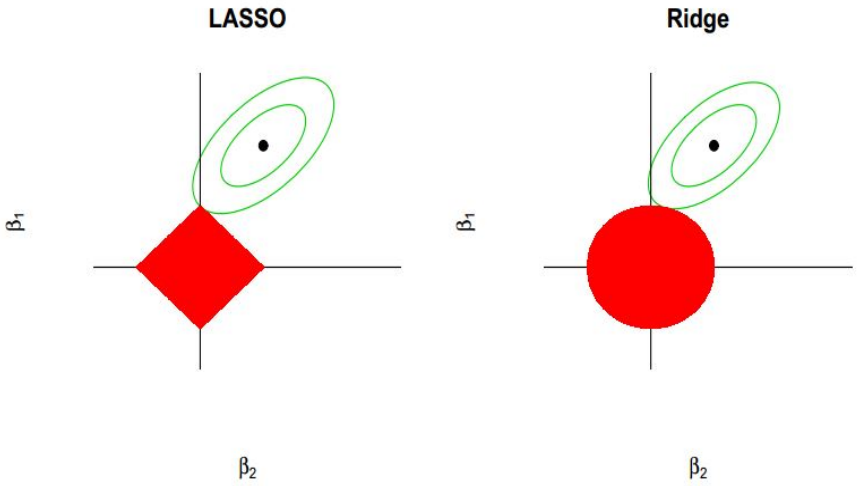

그림 2.1: How the LASSO solution can be exact zero

model interpretation is difficult.

Tibshirani [1996] proposed the least absolute shrinkage and selection operator (LASSO) to overcome these drawbacks of the ridge penalty. The LASSO estimator is defined as

$$\hat{\boldsymbol{\beta}}^{LASSO} = \arg\min_{\boldsymbol{\beta}} \left\{ \mathcal{L}(\boldsymbol{\beta}) + \lambda \sum_{j=1}^{p} |\beta_j| \right\}.$$

Due to the singularity of the objective function at the origin, LASSO penalty can shrink some of the estimated coefficients to be exact zero when the tuning parameter λ is sufficiently large. Figure 2.1(Tibshirani [1996]) illustrates that how the coefficient can be exact zero with the LASSO penalty.

The LASSO estimator varies continuously as λ varies. So, the LASSO estimator can be understood as a continuous subset selection procedure that results in lower variance. Furthermore, the LASSO estimator is less biased than the ridge estimator since the effect of the LASSO penalty to the estimator is smaller.

Many researchers have studied theoretical properties of the LASSO estimator. Raskutti et al. [2011] proved that the minimax lower bound with is $O(K \log p/n)$ when the number of the true nonzero coefficients is K. Bickel et al. [2009] proved the minimax optimality of the LASSO for ultra-high dimensional models where $p = O(\exp(an))$ for some $a > 0$ under regularity conditions.

For finite dimensional linear regression models, Knight and Fu [2000] proved that the LASSO is consistent and the resulting limiting distribution has a positive mass at the origin for the parameter whose true value is exactly zero. However, Zhao and Yu [2006] proved that this positive mass of the limiting distribution could be strictly less than 1, which implies that the LASSO may not be selection consistent. That is, the probability of selecting at least one noisy covariate converges to 1.

In high dimensional data, Zhao and Yu [2006] and Meinshausen and Bühlmann [2006] proved that there exist a sequence of λ_n such that the LASSO estimator

is selection consistent when $p = O(\exp(-an))$ for some $a > 0$ under the irrepresentable condition. Zou [2006] showed that the irrepresentable condition is also necessary. But,Zhao and Yu [2006] explained the irrepresentable condition hardly holds for high dimensional data. That is, the LASSO estimator is not a good choice for variable selection.

The selection consistency and optimal convergence rate cannot be achieved simultaneously by the LASSO estimator. Suppose that design matrix is orthogonal. It can be shown that

$$\hat{\boldsymbol{\beta}}_k^{LASSO}(\lambda) = sign(\hat{\boldsymbol{\beta}}_k^{ols})(|\hat{\boldsymbol{\beta}}_k^{ols}| - \lambda)_+$$

If $\lambda_n/\sqrt{n} \to 0$, then the LASSO estimator is asymptotically equivalent to the ordinary least squared (OLS) estimator, and hence the selection consistency dose not hold. If $\lambda_n/\sqrt{n} \to \infty$, the selection consistency holds but convergence rate of the LASSO estimator is slower than $1/\sqrt{n}$

To make the LASSO be selection consistent, Zou [2006] proposed the adaptive LASSO. The adaptive LASSO preserves the convexity of LASSO but achieves the selection consistency by controlling the weights over the shrinkage parameters. The adaptive LASSO estimator is defined as

$$\hat{\boldsymbol{\beta}}^{aLASSO} = \arg\min_{\boldsymbol{\beta}} \Big\{ \mathcal{L}(\boldsymbol{\beta}) + \lambda \sum_{j=1}^{p} \hat{w}_j |\beta_j| \Big\}.$$

Here, the weight components are givens as $\hat{w}_j = 1/|\tilde{\beta}_j|$ for some initial $\sqrt{n}$-

consistent estimator $\tilde{\beta}$. The weights play an important role for selection consistency. The closer the true coefficient value is to 0, the larger the weight of the penalty is. On the other hand, for the large true coefficient, the weight of penalty is close to 0.

Another method to achieve selection consistency is to use a nonconvex penalty. Huang et al. [2008] studied asymptotic properties of the bridge estimator in sparse high dimensional regression models. They showed that the bridge estimator for $0 < q < 1$ satisfies the selection consistency, where the bridge estimator is defined as

$$\hat{\boldsymbol{\beta}}^{bridge} = \arg\min_{\boldsymbol{\beta}} \Big\{ \mathcal{L}(\boldsymbol{\beta}) + \lambda \sum_{j=1}^{p} |\beta_j|^q \Big\}.$$

Fan and Li [2001] advocated three desirable properties for sparse penalties; sparsity, unbiasedness and continuity. Sparsity means that the resulting estimator automatically sets small coefficients to zero to accomplish variable selection and reduce model complexity. Unbiasedness means that the resulting estimator is nearly unbiased, especially when the true coefficient β_j^* is large, to reduce model bias. Continuity means that the resulting estimator varies continuously in the regularization parameter to reduce instability. Fan and Li [2001] proposed a nonconvex penalty called the smoothly clipped absolute deviation (SCAD) which has the three desired properties. The SCAD estimator

is defined as

$$\hat{\boldsymbol{\beta}}^{SCAD} = \arg\min_{\boldsymbol{\beta}} \Big\{ \mathcal{L}(\boldsymbol{\beta}) + \sum_{j=1}^{p} J_\lambda(|\beta_j|) \Big\},$$

where $J_\lambda(\cdot)$ is the SCAD penalty given by

$$J_\lambda(|\beta|) = \begin{cases} \lambda|\beta| & , |\beta| < \lambda, \\ \frac{a\lambda(|\beta|-\lambda)-(\beta^2-\lambda^2)/2}{a-1} + \lambda^2 & , \lambda \le |\beta| < a\lambda, \\ \frac{(a-1)\lambda^2}{2} + \lambda^2 & , |\beta| \ge a\lambda, \end{cases}$$

for $a > 2$. The singularity at the origin of the SCAD penalty produces a sparse solution as the LASSO does. But for $|\beta| \ge a\lambda$, the penalty becomes flat and so no penalization is applied to large nonzero coefficients. Thus the large nonzero coefficients has no bias asymptotically.

A penalty of similar spirit is the minimax concave penalty (MCP) proposed by Zhang [2010], which is defined as

$$\hat{\boldsymbol{\beta}}^{MCP} = \arg\min_{\boldsymbol{\beta}} \Big\{ \mathcal{L}(\boldsymbol{\beta}) + \sum_{j=1}^{p} J_\lambda(|\beta_j|) \Big\},$$

where $J_\lambda(\cdot)$ is

$$J_\lambda(|\beta|) = \begin{cases} -\beta^2/2a + \lambda|\beta| & , |\beta| < a\lambda, \\ \frac{1}{2}a\lambda^2 & , |\beta| \ge a\lambda, \end{cases}$$

for $a > 0$. See Figure 2.2 for comparison of various penalties.

The key theoretical property of these two nonconvex penalty functions is the oracle property described in Fan and Li [2001]. The oracle property means

the asymptotic equivalence between the estimator and the oracle estimator that is an ideal non-penalized estimator driven as if we knew the irrelevant variables in advance.

Fan and Li [2001] and Fan and Peng [2004] proved the oracle property of the SCAD penalized maximum likelihood estimator when $p = O(n^d)$ for some constant $d < 1$. Kim et al. [2008] and Kwon and Kim [2011] proved that the oracle property can be extended to the cases when $p = O(n^k)$ for some constant $k > 1$. They also proved that under the Gaussian error, the oracle property still holds even when $p = O(\exp(n^c))$ for some constant $c < 1$. For the MCP, Zhang [2010] proved the oracle property.

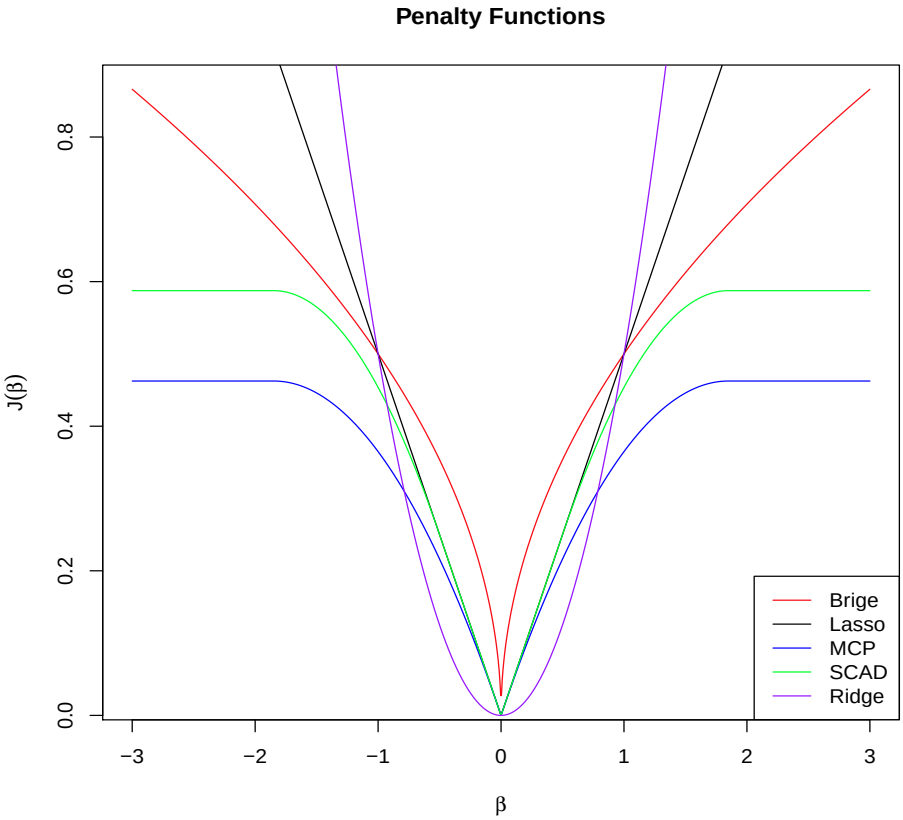

그림 2.2: Plot of penalty functions : the bridge with $\gamma = 0.5$ and the SCAD and MCP with $a = 3.7$, $\lambda = 0.5$

2.2 Penalties for highly correlated covariates

For highly correlated covariates, we can overcome various disadvantages of the standard sparse regularized methods mentioned in Introduction by using a combination of sparse and non-sparse penalties.

Zou and Hastie [2005] propose the Elastic net(Enet) penalty for the selection of correlated covariates. The Enet penalty is devised to combine the advantages of the LASSO and ridge penalties. The naive Enet estimator is defined as

$$\hat{\boldsymbol{\beta}}^{nE} = \arg\min_{\boldsymbol{\beta}} \Big\{ \mathcal{L}(\boldsymbol{\beta}) + \sum_{j=1}^{p} J_\lambda(|\beta_j|) \Big\}.$$

where $J_\lambda(\cdot),\ \lambda = (\lambda_1, \lambda_2)$ is

$$J_\lambda(\boldsymbol{\beta}) = \lambda_1 \sum_{j=1}^{p} |\beta_j| + \lambda_2 \sum_{j=1}^{p} \beta_j^2,$$

In particular, the ridge penalty in the Enet makes the regression coefficients corresponding to highly correlated covariates be similar. So the Enet promotes to select all covariates in a group of highly correlated covariates.

A problem with the naive Enet estimator is that the estimator is doubly regularized and so the bias would be too large. To resolve this problem, Zou and Hastie [2005] proposed the following post-scaling procedure. Let $\hat{\boldsymbol{\beta}}^{nE}(\lambda)$ be the naive Enet estimator. Then, the Enet estimator is defined as

$$\hat{\boldsymbol{\beta}}^{Enet}(\lambda) = (1 + \lambda_2)\hat{\boldsymbol{\beta}}^{nEnet}(\lambda).$$

Zou and Hastie [2005] proved that the Enet estimator converges to the univariate regression estimator as $\lambda_2 \to \infty$ with $\lambda_1 = 0$, where the univariate regression estimator defined by

$$\begin{aligned} \hat{\beta}_j^{ur} &= \arg\min_{\gamma} \frac{1}{n} \sum_{i=1}^{n} (y_i - \gamma x_{ij})^2 \\ &= \frac{1}{n} \sum_{i=1}^{n} y_i x_{ij}. \end{aligned}$$

The univariate estimator is considered to be the least sparse estimator. That is, the post-scaling procedure eliminates the bias due to the l_2 penalty.

A problem with the Enet is that the univariate regression estimator may not be optimal in cases where covariates are highly correlated. For example, when there are two covariates that are exactly the same, the univariate regression estimator is twice as large as the optimal estimator. In contrast, the post-scaled weighted l_2 penalized estimator converges to the optimal estimator. See Section 2.3 for details.

Huang et al. [2010] propose the Mnet that combines the MCP and ridge penalty. The Mnet penalty given as

$$J_\lambda(\boldsymbol{\beta}) = J_{\lambda_1}(\boldsymbol{\beta}) + \lambda_2 \sum_{j=1}^{p} \beta_j^2,$$

where$J_{\lambda_1(\boldsymbol{\beta})}$ is MCP, replaces the l_1 penalty in the Enet by the MCP penalty to reduce the bias introduced by the l_1 penalty. Huang et al. [2010] proved the oracle property.

Huang et al. [2011] proposed the sparse Laplacian penalty which is a linear combination of the MCP and Laplacian penalty. Let $\mathbf{A}$ be a given $p \times p$ symmetric matrix whose (j,k) entries are a_{jk}. Huang et al. [2011] considered the Laplacian penalty given as $\boldsymbol{\beta}^{'}(\mathbf{D}-\mathbf{A})\boldsymbol{\beta}$, where $\mathbf{D} = \text{diag}(d_1, \ldots, d_p)$ with $d_j = \sum_{k=1}^{p} |a_{jk}|$. It turns out that

$$\boldsymbol{\beta}^{'}(\mathbf{D}-\mathbf{A})\boldsymbol{\beta} = \sum_{1 \le j < k \le p} |a_{jk}|(\beta_j - s_{jk}\beta_k)^2, \tag{2.1}$$

where $s_{jk} = \text{sign}(a_{jk})$. That is, the Laplacian penalty acts directly on the squared signed differences of pairs of regression coefficients. Here, a_{jk}s are called the adjacency measures which can be thought to be degrees of the signed similarity of $\mathbf{x}^j$ and $\mathbf{x}^k$, where $\mathbf{x}^j = (x_{1j}, x_{2j}, ..., x_{nj})^{'}$ Examples include $a_{jk} = \text{corr}(\mathbf{x}^j, \mathbf{x}^k)$, $a_{jk} = I(\|\mathbf{x}^j - \mathbf{x}^k\|/\sqrt{n} \le r)$ for some $r \ge 0$, and $a_{jk} = \max\{\text{corr}(\mathbf{x}^j, \mathbf{x}^k), 0\}^{\alpha}$ for $\alpha > 0$. Here, $\text{corr}(\mathbf{x}^j, \mathbf{x}^k)$ is the sample Pearson correlation of $\mathbf{x}^j$ and $\mathbf{x}^k$. In the following, we denote (2.1) by $Lp(\boldsymbol{\beta} : \mathbf{A})$. Huang et al. [2011] also explained that when two covariates are exactly the same,the Laplacian penalized estimator converges to the ordinary least square estimator assuming that the corresponding regression coefficients are equal, which is better in prediction than the univariate regression estimator.

The sparse Laplacian penalty is defined as a linear combination of the MCP

and the Laplacian penalty such that

$$J_\lambda = J_{\lambda_1}(\boldsymbol{\beta}) + \frac{1}{2}\lambda_2 \sum_{1\le j\le k\le p} |a_{jk}|(\beta_j - s_{jk}\beta_k)^2.$$

Huang et al. [2011] proved that the sparse Laplacian estimator is sign consistent and equal to the oracle Laplacian shrinkage estimator.

Kim and Jeon [2014] proposed the weighted Mnet penalty to improve the selection power of the Mnet. The weighted Mnet penalty is combination of the weighted MCP and weighted l_2 penalty. That is, the weighted Mnet penalty is defined as

$$J_\lambda = J_{\lambda_1}(\boldsymbol{\beta}_{\boldsymbol{w},\lambda_2}) + \lambda_2 \sum_{j=1}^{p} w_j \beta_j^2$$

where $\lambda = (\lambda_1, \lambda_2), \boldsymbol{\beta}_{\boldsymbol{w},\lambda_2} = (1+\lambda_2)(w_1\beta_1, \ldots, w_p\beta_p)'$, J_{λ_1} is the MCP and w_j are positive constants. It can control the variable selectivity and coefficient shrinkage separately. The weighted Mnet uses $\boldsymbol{\beta}_{\boldsymbol{w},\lambda_2}$ instead of $\boldsymbol{\beta}$ in the MCP to eliminate the shrinkage effect caused by λ_2 and w_js from variable selection. For example, suppose that $\mathbf{x}^1$ is orthogonal to the other covariates. Then, the weighted Mnet estimator selects $\mathbf{x}^1$ (i.e. $\hat{\beta}_1^{wM} \neq 0$) whenever $|\hat{\beta}_1^{ls}| > a\lambda_1$ regardless of the values of λ_2 and w_js.

2.3 The weighted Mnet penalty for the squared loss

In this section, we review favorable theoretical properties of the weighted Mnet estimator under the squared loss. There is an interesting relation between the weighted l_2 penalty and the Laplacian penalty of Huang et al. [2011]. Theorem 1 of Kim and Jeon [2014] proved that the weighted l_2 penalty with a certain choice of weights becomes the Laplacian penalty of Huang et al. [2011]. The next Remark reviewed the Theorem 1 of Kim and Jeon [2014]

Remark. Suppose $w_j = 1 + \sum_{k \neq j} |\text{corr}(\mathbf{x}^j, \mathbf{x}^k)|$, and let $\hat{\boldsymbol{\beta}}_{sl}^{w2}(\lambda)$ be the post-scaled weighted l_2 penalized estimator defined as

$$\hat{\boldsymbol{\beta}}_{sl}^{w2}(\lambda) = (1+\lambda_2)\text{argmin}_{\boldsymbol{\beta}} \left\{ \frac{1}{n} \sum_{i=1}^{n} (y_i - \mathbf{x}_i^{'}\boldsymbol{\beta})^2 + \lambda \sum_{j=1}^{p} w_j \beta_j^2 \right\}.$$

Then

$$\hat{\boldsymbol{\beta}}_{sl}^{w2}(\lambda) = \text{argmin}_{\boldsymbol{\beta}} \frac{1}{n} \sum_{i=1}^{n} (y_i - \mathbf{x}_i^{'}\boldsymbol{\beta})^2 + \frac{\lambda}{1+\lambda} Lp(\boldsymbol{\beta} : \mathbf{A}),$$

where $\mathbf{A}$ is the $p \times p$ matrix whose (j,k) entry is $\text{corr}(\mathbf{x}^j, \mathbf{x}^k)$.

proof of Remark. Let

$$\tilde{\boldsymbol{\beta}} = \text{argmin}_{\boldsymbol{\beta}} \frac{1}{n} \sum_{i=1}^{n} (y_i - \mathbf{x}_i^{'}\boldsymbol{\beta})^2 + \frac{\lambda}{1+\lambda} \boldsymbol{\beta}^{'} (\mathbf{D} - \mathbf{A}) \boldsymbol{\beta}.$$

Note that

$$\sum_{i=1}^{n}(y_i - \mathbf{x}_i^{'}\beta)^2 = \mathbf{y}^{'}\mathbf{y} - 2\mathbf{y}^{'}\mathbf{X}\beta + \beta^{'}\mathbf{X}^{'}\mathbf{X}\beta.$$

Since the covariates are standardized, we have $\mathbf{X}^{'}\mathbf{X}/n = \mathbf{A}$. By letting $\eta = \lambda/(1+\lambda)$, we have

$$\begin{aligned}
& \frac{1}{n}\sum_{i=1}^{n}(y_i - \mathbf{x}_i^{'}\boldsymbol{\beta})^2 + \eta\boldsymbol{\beta}^{'}(\mathbf{D}-\mathbf{A})\boldsymbol{\beta} \\
= \; & \frac{1}{n}\left\{\mathbf{y}^{'}\mathbf{y} - 2\mathbf{y}^{'}\mathbf{X}\beta + \beta^{'}\mathbf{X}^{'}\mathbf{X}\beta - \eta\beta^{'}\mathbf{X}^{'}\mathbf{X}\beta\right\} + \eta\sum_{j=1}^{p} w_j\beta_j^2 \\
= \; & \frac{1}{n}\left\{\mathbf{y}^{'}\mathbf{y} - 2\mathbf{y}^{'}\mathbf{X}\beta + (1-\eta)\beta^{'}\mathbf{X}^{'}\mathbf{X}\beta +\right\}\eta\sum_{j=1}^{p} w_j\beta_j^2 \\
= \; & \frac{1}{n}\left\{\frac{1}{1-\eta}\sum_{i=1}^{n}\left(y_i - (1-\eta)\mathbf{x}_i^{'}\boldsymbol{\beta}\right)^2 - \frac{\eta}{1-\eta}\sum_{i=1}^{n} y_i^2\right\} + \eta\sum_{j=1}^{p} w_j\beta_j^2.
\end{aligned}$$

By transforming $\boldsymbol{\gamma} = (1-\eta)\boldsymbol{\beta}$, we have $\tilde{\boldsymbol{\beta}} = \hat{\boldsymbol{\gamma}}/(1-\eta)$, where

$$\hat{\boldsymbol{\gamma}} = \text{argmin}_{\boldsymbol{\gamma}}\frac{1}{n}\sum_{i=1}^{n}(y_i - \mathbf{x}_i^{'}\boldsymbol{\gamma})^2 + \frac{\eta}{1-\eta}\sum_{j=1}^{p} w_j\beta_j^2.$$

Since $1/(1-\eta) = (1+\lambda)$ and $\eta/(1-\eta) = \lambda$, $\tilde{\boldsymbol{\beta}} = \hat{\boldsymbol{\beta}}^{w2}(\lambda)$.

This implies that the weighted l_2 penalty shares most desirable properties of the Laplacian penalty. An advantage of the weighted l_2 penalty over the Laplacian penalty is that the former is simpler to use because the penalty is always strictly convex. The weighted MCP is employed to make the estimator sparse.

Another advantage of the weighted Mnet penalty is to reduce the bias induced by the l_2 penalty. Zou and Hastie [2005] showed that the post-scaled l_2 penalized estimator converges to the univariate regression estimator. However, the univariate regression estimator may not be optimal when there are highly correlated covariates.

$\hat{\boldsymbol{\beta}}_{sl}^{w2}(\lambda)$ can be rewritten that

$$\hat{\boldsymbol{\beta}}_{sl}^{w2}(\lambda) = \text{argmin}_{\boldsymbol{\beta}} \boldsymbol{\beta}^{'} \left(\frac{\mathbf{X}^{'}\mathbf{X}/n + \lambda \mathbf{W}}{1+\lambda} \right) \boldsymbol{\beta} - 2\mathbf{y}^{'}\mathbf{X},$$

where $\mathbf{y} = (y_1, \ldots, y_n)^{'}, \mathbf{X} = (\mathbf{x}^1, \ldots, \mathbf{x}^p)$ and $\mathbf{W} = \text{diag}(w_1, \ldots, w_p)$. Hence, as $\lambda \to \infty$, $\hat{\beta}_{sl}^{w2}(\lambda) \to \hat{\beta}_{sl}^{w2}(\infty)$, where

$$\begin{aligned} \hat{\boldsymbol{\beta}}_{sl}^{w2}(\infty) &= \text{argmin}_{\boldsymbol{\beta}} \boldsymbol{\beta}^{'}\mathbf{W}\boldsymbol{\beta} - 2\mathbf{y}^{'}\mathbf{X}\boldsymbol{\beta} \\ &= \frac{\sum_{i=1}^{n} y_i x_{ij}}{w_j}, \end{aligned}$$

which Kim and Jeon [2014] called the weighted univariate regression estimator. And they proved that $\hat{\boldsymbol{\beta}}_{sl}^{w2}(\infty)$ becomes the optimal estimator for highly correlated covariates, by choosing the weights appropriately in the post-scaled weighted l_2 penalty.

There is an simple example in Kim and Jeon [2014]. Consider an extreme situation where $p = 2$ and $\mathbf{x}^1 = \mathbf{x}^2$, where $\mathbf{x}^j = (x_{1j}, \ldots, x_{nj})^{'}$. Suppose $\mathbf{y} = \mathbf{x}^1 + \mathbf{x}^2 + \epsilon$. That is, the true regression coefficient vector $\boldsymbol{\beta}^*$ is $\boldsymbol{\beta}^* = (1, 1)^{'}$. Since $\mathbf{x}^1 = \mathbf{x}^2$, it is easy to see that $E(\hat{\beta}_j^{ur} | \mathbf{x}^1, \mathbf{x}^2) = 2$ for $j = 1, 2$. Hence, the

predictor $\hat{y}_i^{ur} = \hat{\beta}_1^{ur} x_{i1} + \hat{\beta}_2^{ur} x_{i2}$ based on the univariate regression estimator has $E(\hat{y}_i^{ur} | x_{i1}, x_{i2}) = 2(x_{i1} + x_{i2})$, which is twice as large as the expectation of the optimal prediction, which is $x_{i1} + x_{i2}$. In fact, the optimal estimator of $\boldsymbol{\beta}$ is given as $\hat{\beta}_j^{ur}/2$ for $j = 1, 2$, which is the ordinary least square estimator assuming equal regression coefficients. In contrast, the post-scaled weighted l_2 penalized estimator converges to the optimal estimator by use of the weighted l_2 penalty.

제 3 장

Theorical Properties : Weighted Mnet Penalty for Twice Differentiable Convex Losses

3.1 Definition and estimator of the weighted Mnet

The weighted Mnet penalty is a linear combination of the weighted MCP and weighted l_2 penalty. The weighted Mnet estimator for a twice differentiable

convex loss is defined as

$$\hat{\boldsymbol{\beta}}^{wM}(\lambda) = (1+\lambda_2) argmin_{\boldsymbol{\beta}\in\Theta}\{\frac{1}{n}\sum_{i=1}^{n} l(y_i, \mathbf{x}_i'\boldsymbol{\beta}) + J_{\lambda_1}(\boldsymbol{\beta}_{w,\lambda_2}) + \lambda_2 \sum_{j=1}^{p} w_j \beta_j^2\}$$

where $l(\cdot,\cdot)$ is a twice differentiable convex losses.

3.2 Compare to Laplacian penalty on twice differentiable losses

Let $\mathcal{L}(\boldsymbol{\beta}) = \sum_{i=1}^{n} l(y_i, \mathbf{x}_i'\boldsymbol{\beta})/n$, where the loss $l(\cdot,\cdot)$ is twicely differentiable. By the second order Taylor expansion of $\mathcal{L}(\boldsymbol{\beta})$ around $\boldsymbol{\beta}^c$, we have

$$\mathcal{L}(\boldsymbol{\beta}) \approx \mathcal{L}(\boldsymbol{\beta}^c) + (\boldsymbol{\beta}-\boldsymbol{\beta}^c)'\nabla(\boldsymbol{\beta}^c) + \frac{1}{2}(\boldsymbol{\beta}-\boldsymbol{\beta}^*)'\mathbf{H}(\boldsymbol{\beta}^c)(\boldsymbol{\beta}-\boldsymbol{\beta}^c), \tag{3.1}$$

where $\nabla(\boldsymbol{\beta}) = \partial\mathcal{L}(\boldsymbol{\beta})/\partial\boldsymbol{\beta}$ and $\mathbf{H}(\boldsymbol{\beta}) = \partial^2\mathcal{L}(\boldsymbol{\beta})/\partial\boldsymbol{\beta}\partial\boldsymbol{\beta}$. Let $\tilde{\mathcal{L}}(\boldsymbol{\beta})$ be the approximated risk defined on the right hand side of (3.1). Then, similarly to Remark in section 2.3, we can prove the following Theorem.

Theorem 1 *Let* $\mathbf{A} = \mathbf{H}(\boldsymbol{\beta}^c)$ *is nonsigular and* $w_k = \sum_{l=1}^{p} |h_{kl}|$ *with* h_{kl} *being the* (k,l) *entries of* $\mathbf{H}(\boldsymbol{\beta}^c)$.

$$\text{argmin}_{\boldsymbol{\beta}}\tilde{\mathcal{L}}(\boldsymbol{\beta}) + \frac{1}{2}\frac{\lambda}{(1+\lambda)} Lp(\boldsymbol{\beta}:\mathbf{A}) = (1+\lambda)\left\{\text{argmin}_{\boldsymbol{\beta}}\tilde{\mathcal{L}}(\boldsymbol{\beta}) + \frac{1}{2}\lambda\sum_{k=1}^{p} w_k\beta_k^2\right\}, \tag{3.2}$$

Motivated by (3.2), we propose to set $w_k = \sum_{l=1}^{p} 1 + \phi(\tilde{h}_{kl})$ for some function ϕ such as $\phi(x) = |x|, \phi(x) = I(|x| > 0.6)$ or $\phi(x) = (\max\{x, 0\})^6$, where $\tilde{h}_{kl}$ are the (k, l) entries of $\mathbf{H}(\hat{\boldsymbol{\beta}}^{(0)})$ for some initial estimator $\hat{\boldsymbol{\beta}}^{(0)}$ such as the LASSO estimator.

Theorem 1 implies that the weighted l_2 penalty can be compared with the Laplacian penalty. An obvious advantage of the weighted l_2 penalty over the Laplacian penalty is that the former is computationally simpler to use because it is strictly convex and hence the optimization problem becomes easier. This advantage becomes more important for generalized linear models.

Remark. A disadvantage of the weighted l_2 penalty is that the set of solutions for the Laplacian penalty is larger than that of the weighted l_2 penalty. That is, by Theorem 1, we have

$$\{\hat{\boldsymbol{\beta}}^{w2}(\lambda) : \lambda > 0\} = \{\hat{\boldsymbol{\beta}}^{lp}(\lambda) : 0 < \lambda < 1\} \subset \{\hat{\boldsymbol{\beta}}^{lp}(\lambda) : \lambda > 0\},$$

where

$$\hat{\boldsymbol{\beta}}^{lp}(\lambda) = \text{argmin}_{\boldsymbol{\beta}} \tilde{\mathcal{L}}(\boldsymbol{\beta}) + \frac{1}{2}\lambda Lp(\boldsymbol{\beta} : \mathbf{A}),$$

$$\hat{\boldsymbol{\beta}}^{w2}(\lambda) = \text{argmin}_{\boldsymbol{\beta}} \tilde{\mathcal{L}}(\boldsymbol{\beta}) + \frac{1}{2}\lambda \sum_{k=1}^{p} w_k \beta_k^2.$$

However, the optimal λ for the Laplacian penalty is in most cases less than 1 and hence this disadvantage is not a serious problem.

Theorem 1 suggests that the weights in the weighted l_2 penalty to squared loss can be chosen through adjacency measures . That is, motivated by Theorem 1, we can set $w_j = 1 + \sum_{k \neq j} |a_{jk}|$ for given adjacency measures a_{jk}. Note that except for $a_{jk} = \text{corr}(\mathbf{x}^j, \mathbf{x}^k)$, the Laplacian penalty with adjacency measures a_{jk} is different from the weighted l_2 penalty with $w_j = 1 + \sum_{k \neq j} |a_{jk}|$.

3.3 Oracle property

3.3.1 Oracle weighted ridge estimator

Let $\boldsymbol{\beta}^* = (\beta_1^*, ..., \beta_p^*)'$ be the true regression coefficients. We assume that there exists a nonempty subset $\mathcal{A} = \{j : |\beta_j^*| \neq 0\}$ that is the index set of the true signal coefficients. That is $|\beta_j^*| \neq 0$ for $j \in \mathcal{A}$ and $|\beta_j^*| = 0$ for $j \in \mathcal{A}^c$. For a given $\lambda_2 \geq 0$, we can define the oracle ridge and the orace weighted ridge estimators as

$$\hat{\boldsymbol{\beta}}^o(\lambda_2) = (\arg \min_{\beta_j = 0, j \in \mathcal{A}^c} \frac{1}{n} \sum_{i=1}^{n} l(y_i, \mathbf{x}_i' \beta) + \lambda_2 \sum_{j=1}^{p} w_j \beta_j^2)$$

$$\hat{\boldsymbol{\beta}}^{ow}(\lambda_2) = (1 + \lambda_2)(\arg \min_{\beta_j = 0, j \in \mathcal{A}^c} \frac{1}{n} \sum_{i=1}^{n} l(y_i, \mathbf{x}_i' \beta) + \lambda_2 \sum_{j=1}^{p} w_j \beta_j^2).$$

Notice that $\hat{\boldsymbol{\beta}}^o(\lambda_2)$ and $\hat{\boldsymbol{\beta}}^{ow}(\lambda_2)$ are the penalized estimators obtained only with the covariates in the signal group. In this sense, we refer $\hat{\boldsymbol{\beta}}^o(\lambda_2)$ and $\hat{\boldsymbol{\beta}}^{ow}(\lambda_2)$ as

the oracle ridge and oracle weighted ridge estimators, respectively. Specially, it is similar to the oracle MLE in Kwon and Kim [2011] when $\lambda_2 = 0$. In practice, the oracle ridge and oracle weighted ridge estimators are not available since we do not know the signal covariates. However, we will prove that the weighted Mnet estimator is asymptotically equivalent to the oracle weighted ridge estimator provided that we choose λ_1 appropriately. This result explains the role of the regularization parameters λ_1 and λ_2. The selection of the signal groups is done by λ_1 and the shrinkage within the signal covariates is done by λ_2.

3.3.2 Oracle property

In this section, we give sufficient conditions with which the oracle weighted ridge estimator and the weighted Mnet estimator are asymptotically equivalent, which is called 'oracle property'. We assume the following regularity conditions.

A1 Conditions for penalty

For class of penalty functions $J_\lambda(t)$, there exists $a > 0$ such that

1. $\dot{J}_\lambda(\cdot)$ is nonnegative, non-increasing and continuous over $(0, \infty)$
2. $\lim_{t \to 0+} \dot{J}_\lambda(t) = \lambda$ and $\dot{J}_\lambda(t) = 0, t \geq a\lambda$

3. $\dot{J}_\lambda(t) \geq (\lambda - t/a)_+, t > 0$

A2 Let $\sigma(\boldsymbol{\beta}) = \{j : \beta_j \neq 0\}$. There exist positive integer s and positive numbers $\rho_* < \rho^*$ such that

$$\begin{aligned}\Theta &= \{\beta \in R^p : |\sigma(\boldsymbol{\beta})| \leq s, \\ &\quad \rho_{\min}(H(\beta)_{\sigma(\boldsymbol{\beta})\cup\mathcal{A}}) \geq \rho_* > 0, \\ &\quad \rho_{\max}(H(\beta)_{\sigma(\boldsymbol{\beta})\cup\mathcal{A}}) \leq \rho^*\}\end{aligned}$$

where $\rho_{\min}(\cdot)$ is the smallest eigenvalue and $\rho_{\max}(\cdot)$ is the largest eigenvalue. Also, we assume that $\hat{\boldsymbol{\beta}}^o(\lambda_2)$ and $\hat{\boldsymbol{\beta}}^{ow}(\lambda_2)$ are inside Θ.

A3
1. $\min_{j\in\mathcal{A}} |\hat{\beta}_j^o| >> \lambda_1 >> \sqrt{\log p/n}$
2. $\sup_{j\in\mathcal{A}^c} |\frac{\partial R(\hat{\beta}_j^o)}{\partial \hat{\beta}_j^o}| = O_p(\sqrt{\log p/n})$
3. $\lambda_2 w^* ||\hat{\beta}^o|| << \lambda_1$, where $w^* = max_k w_k$

Condition **A1** specifies the class of nonconvex penalties which includes the SCAD and MCP. **A2** assumes that the design matrix is well posed, which is similar to the sparse Riesz condition for linear regression models (Zhang and Huang [2008]). Condition **A3** assumes that the true signal coefficients are sufficiently large and the bias induced by λ_2 and w is not too large.

Theorem 2 *Under the conditions A1-A3. We have*

$$Pr\{\hat{\boldsymbol{\beta}}^{wM}(\lambda_2) = \hat{\boldsymbol{\beta}}^{ow}(\lambda_2)\} \to 1$$

Theorem 2 show that the weighted Mnet estimator converge to oracle weighted ridge estimator with probability 1. Our regularity conditions are more intuitively understandable than those for the Mnet or sparse Laplacian penalties.

제 4 장

Optimization Algorithm

In this section, we propose an efficient algorithm for the weighted Mnet. We review various optimization algorithms for penalized methods in section 4.1. Because the weighted Mnet penalty is a combination of the weighted MCP and weighted l_1 penalty, it is a nonconvex penalty. To solve the nonconvex problem, we apply the convex concave procedure of Yuille and Rangarajan [2003]. In section 4.2, we review the convex concave procedure and we apply it to the weighted Mnet estimator in section 4.3.

1. Initialize $\mathcal{A} = \varnothing$ and $\boldsymbol{\beta} = 0$

2. For a given constant $0 < \gamma \leq 1$, update $\mathcal{A}$ until convergence

   (a) $\mathbf{r} = \mathbf{y} - \gamma \mathbf{X}_{\mathcal{A}} \boldsymbol{\beta}_{\mathcal{A}}$

   (b) $j = \arg\min_k \frac{1}{n} l(r, \mathbf{X}^k \boldsymbol{\beta}_k)$

   (c) update $\mathcal{A}$, $\mathcal{A} = \mathcal{A} \cup \{j\}$

   (d) $\boldsymbol{\beta}_{\mathcal{A}} = \arg\min_\gamma \frac{1}{n} l(r, \mathbf{X}_{\mathcal{A}} \gamma)$

그림 4.1: LARS algorithm

4.1 Optimization algorithms for sparse penalized methods.

4.1.1 Least Angle Regression(LARS) algorithm

One of the popularly used optimization algorithms for the LASSO is the Least Angle Regression(LARS) algorithm by Efron et al. [2004], which is briefly described at Figure 4.1.

The LARS adds to the active set only one coefficient in each iteration as the forward selection. However, the LARS updates only the added coefficient while the forward selection updates all of the coefficients in the active set

fully. In addition, the LARS chooses an added coefficient by comparing the correlations between the current residuals. This is sharply contrast with the forward selection which is a hard decision rule in the sense that covariates either survive or die. Therefore, The LARS algorithm can be considered as a continuous version of the forward selection procedure. Note that the LARS does not have a deletion step. If the deletion step is added, Efron et al. [2004] show that the algorithm becomes the path finding algorithm of LASSO.

4.1.2 Coordinate descent algorithm

The coordinate descent algorithm optimizes the objective function through a sequence of one-dimensional optimizations. The coordinate descent algorithm for penalized methods is often called the shooting algorithm. The shooting algorithm is proposed by Fu [1998] and improved by Friedman et al. [2007]. The main idea of the shooting algorithm is to solve the problem by updating each coordinate iteratively until convergence. That is, the shooting algorithm updates each coordinate by minimizing the penalized empirical expected loss along the selected coordinate with all other coordinates fixed. When the objective function is convex and differentiable, the subproblem can be solved quickly and hence the coordinate descent algorithm is simple and efficient.

For example, consider the l_1 penalized quadratic function such as

$$Q_{\lambda_1}(\boldsymbol{\beta}) \equiv \boldsymbol{\beta}^T \mathbf{A}\boldsymbol{\beta} + \boldsymbol{\beta}^T \mathbf{b} + \mathbf{c} + \lambda_1 ||\boldsymbol{\beta}||_1$$

where $\mathbf{A}$ is $p \times p$ nonnegative matrix, $\mathbf{b}$ is $p \times 1$ vector, $\mathbf{c}$ is some constant, and $||\cdot||_1$ is L_1 norm. For a given current solution $\boldsymbol{\beta}_{-j} = (\beta_1, ...\beta_{j-1}, 0, \beta_{j+1}, ...\beta_p)^T$, we update β_j by minimizing $Q_{\lambda_1}(\beta_j|\boldsymbol{\beta}_{-j})$, where

$$Q_{\lambda_1}(\beta_j|\boldsymbol{\beta}_{-j}) = \beta_k \mathbf{A}_{jj}\beta_j + \boldsymbol{\beta}_j^T \mathbf{A}\boldsymbol{\beta}_{-j} + \boldsymbol{\beta}_{-j}^T \mathbf{A}\boldsymbol{\beta}_j + \boldsymbol{\beta}_j^T \mathbf{b} + \mathbf{c}_2 + \lambda_1 ||\beta_j||_1.$$

The solution $\hat{\beta}_j$ is given by

$$\begin{aligned} \hat{\beta}_j &= \arg\min_{\beta_j} Q_{\lambda_1}(\beta_j|\boldsymbol{\beta}_{-j}) \\ &= \begin{cases} 0, & |\beta_j| \leq \lambda_1 \ ; \\ \frac{sign(\beta_j)(|-1/2\mathbf{b}-\mathbf{A}\boldsymbol{\beta}_{-j}|-\lambda_1)}{\mathbf{A}_{jj}}, & |\beta_j| > \lambda_1. \end{cases} \end{aligned}$$

To sum up, the coordinate descent algorithm for the l_1 penalized quadratic function is given in Figure 4.2.

When design matrix is orthogonal and the squared loss is used, the coordinate descent algorithm can be sued for the weighted Mnet estimator, which is a modification of the coordinate descent algorithm of the Mnet of Huang et al. [2010]. The coordinate descent algorithm updates $\boldsymbol{\beta}$ by updating each coordinate $\beta_j, j = 1, \ldots, p$ sequentially. That is, for a given current solution

1. Find the initial estimator $\hat{\boldsymbol{\beta}}^c$.

2. Update $\hat{\boldsymbol{\beta}}^c$ by updating $\hat{\beta}_j$ of $(1), j = 1, ..., p$ sequentially until all $\hat{\beta}_j$ are converged.

   (a) $\hat{\beta}_j$ is given by

$$\begin{aligned} \hat{\beta}_j &= \arg\min_{\beta_j} Q_{\lambda_1}(\beta_j | \boldsymbol{\beta}_{-j}) \\ &= \begin{cases} 0, & |\beta_j| \le \lambda_1 \ ; \\ \frac{\text{sign}(\beta_j)(|-1/2\mathbf{b}-\mathbf{A}\boldsymbol{\beta}_{-j}|-\lambda_1)}{\mathbf{A}_{jj}}, & |\beta_j| > \lambda_1. \end{cases} \end{aligned}$$

   (b) updating $\hat{\beta}_j^c = \tilde{\beta}_j$.

그림 4.2: Example : Coordinate descent algorithm the quadratic function with L_1 penalty

1. Find the initial estimator $\hat{\boldsymbol{\beta}}^c$.

2. Update $\hat{\boldsymbol{\beta}}^c$ by updating $\hat{\beta}_j$, $j = 1, ..., p$ sequentially

   until all $\hat{\beta}_j$ are converged.

   (a) $\hat{\beta}_j$ is given by

$$\hat{\beta}_j = \begin{cases} \text{sign}(z_j)\frac{a(|z_j|-\lambda_1/2)_+}{a(1+w_j\lambda_2)-1/2} & \text{if } |z_j| \le a\lambda_1(1+\lambda_2) \\ \frac{z_j}{1+\lambda_2} & \text{if } |z_j| > a\lambda_1(1+\lambda_2) \end{cases},$$

   where $z_j = \sum_{i=1}^n y_i x_{ij}$.

   (b) updating $\hat{\beta}_j^c = \tilde{\beta}_j$.

그림 4.3: Example : Coordinate descent algorithm for the weighted Mnet estimator with orthogonal design to squared loss

$\hat{\boldsymbol{\beta}}^c$, we update β_j by minimizing

$$\sum_{i=1}^{n}(y_i - \sum_{k\neq j} x_{ik}\hat{\beta}_k^c - x_{ij}\beta_j)^2 + J_{\lambda_1}(|\beta_j|) + \lambda_2 w_j \beta_j^2.$$

When $J_{\lambda_1}(|\beta_j|)$ is the MCP, the colsed form solution of the aobve objective function is available. The coordinate descent algorithm for the weighted Mnet estimator with the squared loss is presented in Figure 4.3.

4.2 Concave Convex Procedure(CCCP)

In this section, we introduce the Concave-Convex procedure (CCCP) proposed by Yuille and Rangarajan [2003]. It is an optimization algorithm for nonconvex function that can be decomposed to the sum of convex and concave functions. The main idea is to update the solution by the minimizer of the tight convex upper bound of the objective function at the current solution. Since the objective function decompose to the sum of convex and concave functions, we can easily find the tight convex upper bound of the objective function using the hyperplane of the concave part at the current solution. To explain details, suppose that we are to minimize a nonconvex function $Q(\boldsymbol{\beta})$. Suppose $Q(\boldsymbol{\beta})$ is a sum of convex and concave functions such that $Q(\boldsymbol{\beta}) = Q_{vex}(\boldsymbol{\beta}) + Q_{cav}(\boldsymbol{\beta})$. For a given current solution $\hat{\boldsymbol{\beta}}^c$, the tight convex upper bound is given by

$$U(\boldsymbol{\beta}) = Q_{vex}(\boldsymbol{\beta}) + \{\partial Q_{cav}(\hat{\boldsymbol{\beta}}^c)/\partial \boldsymbol{\beta}\}^T \boldsymbol{\beta}$$

Then we update the solution by the minimizer of $U(\boldsymbol{\beta})$. Since $U(\boldsymbol{\beta})$ is a convex function, we can easily find the minimizer using various convex optimization algorithms. This procedure is repeated until the solution converge. It always converges to a local minimizer by the decent property of the CCCP algorithm by Yuille and Rangarajan [2003].

For example, when the squared loss is used, the CCCP algorithm can

be used for the weighted Mnet estimator as follows. Let $\tilde{J}_\lambda(\boldsymbol{\beta}) = J_\lambda(\boldsymbol{\beta}) - \lambda \sum_{j=1}^{p} |\beta_j|$. We assume that $\tilde{J}_\lambda(\boldsymbol{\beta})$ is concave, which is satisfied for the SCAD penalty and MCP. For a given current solution $\boldsymbol{\beta}^c$, we update the solution by minimizing

$$\sum_{i=1}^{n} (y_i - \mathbf{x}_i^{'}\boldsymbol{\beta})^2 + \nabla_{\lambda_1}(\boldsymbol{\beta}^c)^{'}\boldsymbol{\beta} + \sum_{j=1}^{p} w_j \beta_j^2 + \lambda_1 \sum_{j=1}^{p} |\beta_j|, \tag{4.1}$$

where $\nabla_{\lambda_1}(\boldsymbol{\beta}) = \partial J_{\lambda_1}(\boldsymbol{\beta})/\partial\boldsymbol{\beta}$. Since (4.1) is a sum of a quadratic form and the LASSO penalty, the minimizer can be obtained efficiently by the LARS algorithm of Efron et al. [2004] or the coordinate descent algorithm of Friedman et al. [2007].

4.3 Optimization algorithm for the weighted Mnet estimator

In this section, we introduce the optimization algorithm for the weighted Mnet estimator with a general twice differentiable convex loss. Recall that the weighted Mnet estimator is defined as

$$\hat{\boldsymbol{\beta}}^{wM} = \arg\min_{\beta} \mathcal{L}(\boldsymbol{\beta}) + J_{\lambda}(\boldsymbol{\beta})$$

where $\mathcal{L}(\boldsymbol{\beta}) = \frac{1}{n}\sum_{i=1}^{n} l(y_i, \mathbf{x}_i'\beta_i)$, $J_{\lambda}(\boldsymbol{\beta}) = J_{\lambda_1}(\boldsymbol{\beta}_{w,\lambda_2}) + \lambda_2 \sum_{j=1}^{p} w_j \beta_j^2$, $J_{\lambda_1}(\cdot)$ is the MCP and $\boldsymbol{\beta}_{w,\lambda_2} = (1+\lambda_2)(w_1\beta_1, ..., w_p\beta_p)'$

Note that the weighted MCP penalty function can be decomposed by the sum of convex and concave functions, so we can rewrite the minimizing problem as

$$\hat{\boldsymbol{\beta}}^{wM} = \arg\min_{\boldsymbol{\beta}} \underbrace{\mathcal{L}(\boldsymbol{\beta}) + \lambda_1 \sum_{j=1}^{p} ||\beta_j|| + \lambda_2 \sum_{j=1}^{p} w_j \beta_j^2}_{\text{convex function}} + \underbrace{\tilde{J}_{\lambda}(\boldsymbol{\beta})}_{\text{concave function}} \tag{4.2}$$

where $\tilde{J}_{\lambda}(\boldsymbol{\beta}) = J_{\lambda_1}(\beta_{w,\lambda_2}) - \lambda_1 \sum_{j=1}^{p} ||\beta_j||$. Hence, we can find a local minimizer of (4.2) by using of the Convex Concave Procedure (CCCP) as follows. Let $\nabla \tilde{J}(\boldsymbol{\beta})$ be the sub-gradient of $\tilde{J}(\boldsymbol{\beta})$. For a given current solution $\hat{\boldsymbol{\beta}}^c$, we update the solution by minimizing

$$\mathcal{L}(\boldsymbol{\beta}) + \nabla \tilde{J}(\hat{\boldsymbol{\beta}}^c)^T \boldsymbol{\beta} + \lambda_1 \|\boldsymbol{\beta}\|_1, \tag{4.3}$$

1. Find the initial estimator $\hat{\boldsymbol{\beta}}^c$.

2. Do followings until convergence.

   (a) Let

   $$U_\lambda(\boldsymbol{\beta}) = \mathcal{L}(\boldsymbol{\beta}) + \nabla \tilde{J}(\hat{\boldsymbol{\beta}}^c)^T \boldsymbol{\beta} + \lambda_1 ||\boldsymbol{\beta}||_1.$$

   (b) Find the minimizer

   $$\hat{\boldsymbol{\beta}} = \arg\min_{\boldsymbol{\beta}} U_\lambda(\boldsymbol{\beta}).$$

   (c) Update the $\hat{\boldsymbol{\beta}}^c$ by $\hat{\boldsymbol{\beta}}$.

그림 4.4: General algorithm for computing weighted Mnet esimator

which can be solved relatively easily since the objective function is convex. We will discuss how to optimize the above function later on. We iterate this procedure until the solution converges. Since the decent property of the CCCP, it always converges to a local solution. Figure 4.4 summarizes the algorithm.

If the loss function $L(\boldsymbol{\beta})$ is a piecewise quadratic function of $\boldsymbol{\beta}$, we can apply the idea of LARS algorithm [Efron et al., 2004] and [Rosset and Zhu, 2007] directly to minimize 4.3. For general loss function, we can minimize 4.3 using the local quadratic approximation (LQA) as follows. For a given initial estimator $\hat{\boldsymbol{\beta}}^c$, we can approximate the empirical expected loss function $\mathcal{L}(\boldsymbol{\beta})$

as a quadratic function

$$\tilde{\mathcal{L}}(\boldsymbol{\beta}|\hat{\boldsymbol{\beta}}^c) = \mathcal{L}(\hat{\boldsymbol{\beta}}^c) + \nabla\mathcal{L}(\hat{\boldsymbol{\beta}}^c)^T(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}^c) + (\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}^c)^T\nabla^2\mathcal{L}(\hat{\boldsymbol{\beta}}^c)(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}^c)/2 \quad (4.4)$$

where $\nabla\mathcal{L}(\boldsymbol{\beta})$ and $\nabla^2\mathcal{L}(\boldsymbol{\beta})$ are the first and second derivatives of $\mathcal{L}(\boldsymbol{\beta})$, respectively. Now, using the LARS algorithm or coordinate descent algorithm, we can find the minimizer $\hat{\beta}^a$ of the quadratically approximated object function 4.5

$$\tilde{Q}(\boldsymbol{\beta}|\hat{\boldsymbol{\beta}}^c) = \tilde{\mathcal{L}}(\boldsymbol{\beta}|\hat{\boldsymbol{\beta}}^c) + \nabla\tilde{J}(\hat{\boldsymbol{\beta}}^c)^T\boldsymbol{\beta} + \lambda||\boldsymbol{\beta}||_1. \quad (4.5)$$

We update $\hat{\beta}^c$ by $\hat{\beta}^a$ amd repeat this procedure until convergence to obtain the minimizer of 4.3. Figure 4.5 summarizes the optimization algorithm for the weighted Mnet estimator.

1. Find an initial estimator $\hat{\boldsymbol{\beta}}^c$.

2. Do followings until convergence.

   (a) Let

   $$\tilde{Q}(\boldsymbol{\beta}|\hat{\boldsymbol{\beta}}^c) = \tilde{\mathcal{L}}(\boldsymbol{\beta}|\hat{\boldsymbol{\beta}}^c) + \nabla \tilde{J}(\hat{\boldsymbol{\beta}}^c)^T\boldsymbol{\beta} + \lambda||\boldsymbol{\beta}||_1.$$

   where,

   $$\begin{aligned}\tilde{L}(\boldsymbol{\beta}|\hat{\boldsymbol{\beta}}^c) &= \mathcal{L}(\hat{\boldsymbol{\beta}}^c) + \nabla\mathcal{L}(\hat{\boldsymbol{\beta}}^c)^T(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}^c) \\ &+ (\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}^c)^T\nabla^2\mathcal{L}(\hat{\boldsymbol{\beta}}^c)(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}^c)/2\end{aligned}$$

   (b) Find the minimizer $\hat{\boldsymbol{\beta}}^a = \arg\min \tilde{Q}(\boldsymbol{\beta}|\hat{\boldsymbol{\beta}}^c)$

   (c) Return $\hat{\boldsymbol{\beta}}^c = \hat{\boldsymbol{\beta}}^a$

그림 4.5: Local quadratic approximation (LQA) algorithm for the weighted Mnet estimator

제 5 장

Numerical studies

In this chapter, we investigate finite sample performance of the weighted Mnet estimator through simulation and real data analysis. We compare the weighted Mnet estimator with the Mnet and sparce Laplacian (SLP) estimators. Recall that the weighted Mnet, Mnet and sparse Laplacian estimators are defined as

The Mnet with the SLP, naive Enet, Enet and Mnet estimators that are defined as

$$\boldsymbol{\beta}^{SLP}(\lambda) = \text{argmin}_{\boldsymbol{\beta}} \left\{ \mathcal{L}(\boldsymbol{\beta}) + \lambda_2 \boldsymbol{\beta}^{'}(\mathbf{D} - \mathbf{A})\boldsymbol{\beta} + J_{\lambda_1}(\boldsymbol{\beta}) \right\},$$

$$\boldsymbol{\beta}^{nEnet}(\lambda) = \text{argmin}_{\boldsymbol{\beta}} \left\{ \mathcal{L}(\boldsymbol{\beta}) + \lambda_2 \sum_{k=1}^{p} \beta_k^2 + \lambda_1 \sum_{k=1}^{p} |\beta_k| \right\},$$

$$\boldsymbol{\beta}^{Enet}(\lambda) = (1+\lambda_2)\text{argmin}_{\boldsymbol{\beta}} \left\{ \mathcal{L}(\boldsymbol{\beta}) + \lambda_2 \sum_{k=1}^{p} \beta_k^2 + \lambda_1 \sum_{k=1}^{p} |\beta_k| \right\},$$

and

$$\boldsymbol{\beta}^{Mnet}(\lambda) = \text{argmin}_{\boldsymbol{\beta}} \left\{ \mathcal{L}(\boldsymbol{\beta}) + \lambda_2 \sum_{k=1}^{p} \beta_k^2 + J_{\lambda_1}(\boldsymbol{\beta}) \right\}.$$

We consider three adjacency measures for the Laplacian penalty $a_{jk} = \text{corr}(\mathbf{x}^j, \mathbf{x}^k)$ (Lap1), $a_{jk} = \max\{0, \text{corr}(\mathbf{x}^j, \mathbf{x}^k)\}^6$ (Lap2) and $a_{jk} = |\text{corr}(\mathbf{x}^j, \mathbf{x}^k)|$ $I(|\text{corr}(\mathbf{x}^j, \mathbf{x}^k)| > 0.6)$ (Lap3). The corresponding three weights for the weighted Mnet penalty (WMnet1, WMnet2 and WMnet3) are $w_k = \sum_{l=1}^{p} \phi(\tilde{h}_{kl})$ for some function ϕ such as $\phi(x) = |x|$(WMnet1), $\phi(x) = (\max\{x, 0\})^6$, (WMnet2) and $\phi(x) = I(|x| > 0.6)$ (WMnet3), where $\tilde{h}_{kl}$ are the (k, l) entries of hessian matrix $\mathbf{H}(\hat{\boldsymbol{\beta}}^{(0)})$ for some initial estimator $\hat{\boldsymbol{\beta}}^{(0)}$. For the initial estimator, we use the LASSO estimator.

5.1 Simulation studies

For simulation studies, we consider the logistic regression model:

$$y|\mathbf{x} \sim \text{Bernoulli}\Big(\frac{\exp(\mathbf{x}^T\boldsymbol{\beta})}{1 + \exp(\mathbf{x}^T\boldsymbol{\beta})}\Big).$$

For all examples, we set the sample size $n = 100$ and the dimension of covariates $p = 500$. We consider the following three models for simulation.

Example 1 We set $n = 100$ and $p = 500$. The correlations of covariates j and k are set to $r^{-|j-k|}$ with $r = 0.7$. We divide the 500 covariates into 100 blocks

of size 5. We randomly select five blocks and assign values generated from the uniform distribution on $[0.25, 0.75]$ to the regression coefficients in those five blocks. The other regression coefficients are set to 0.

Example 2 Let $n = 100, p = 500$ and $\beta_k^* = 1I(k \leq 15)$. We generate $\mathbf{x}$ by

$$x_k = z_1 + \epsilon_k^x, z_1 \sim N(0,1), k = 1, \ldots, 5$$

$$x_k = z_2 + \epsilon_k^x, z_2 \sim N(0,1), k = 6, \ldots, 10$$

$$x_k = z_3 + \epsilon_k^x, z_3 \sim N(0,1), k = 11, \ldots, 15$$

and $x_k \sim N(0,1)$ independently for $k = 16, \ldots, 500$, where $\epsilon_k^x \sim N(0, 0.01)$ independently.

Example 3 Let $n = 100, p = 500$ and $\beta_k^* = 1I(k \leq 15)$. We generate $\mathbf{x}$ by

$$x_k = z_1 + \epsilon_k^x, z_1 \sim N(0,1), k = 1, \ldots, 2$$

$$x_k = z_2 + \epsilon_k^x, z_2 \sim N(0,1), k = 3, \ldots, 7$$

$$x_k = z_3 + \epsilon_k^x, z_3 \sim N(0,1), k = 8, \ldots, 15$$

and $x_k \sim N(0,1)$ independently for $k = 16, \ldots, 500$, where $\epsilon_k^x \sim N(0, 0.01)$ independently.

표 5.1: Example 1 result

Methods	**Error**	**s.e.**	**T.P.**	**F.P.**	**n.hood**
Enet	0.21660	0.00440	11.3	18.16	0.25582
Mnet	0.21966	0.00445	8.92	11.36	0.25982
Lap1	0.21482	0.00412	12.14	23.12	0.24721
Lap2	0.21681	0.00444	8.94	12	0.26142
Lap3	0.21569	0.00449	9.14	12.84	0.25540
WMnet 1	0.21492	0.00431	9.2	10.14	0.24924
WMnet 2	0.21518	0.00442	8.9	10.1	0.25263
WMnet 3	0.21683	0.00438	8.72	10.48	0.25758

Based on 50 replications of simulation, Tables 5.1, 5.2 and 5.3 report the test negative loglikelihood value (n.hood) and the misclassification error (Error) based on independent test sample 10000, the standard error of the misclassification error (s.e.), the average number of signal variables included in the selected model (T.P.) and the average number of noisy variables included in the selected model (F.P.). The regularization parameters is selected based on the validation sets of size 100.

In Exmaple 2, neighborhood covariates are highly correlated. In Example 3, the first 15 covariates are signal. In addition, the signal covariates form

표 5.2: Example 2 result

Methods	**Error**	**s.e.**	**T.P.**	**F.P.**	**n.hood**
Enet	0.07193	0.00376	11.1	5.96	0.17419
Mnet	0.05125	0.00309	6.2	2.1	0.13999
Lap1	0.05478	0.00311	5.74	2.38	0.14567
Lap2	0.05147	0.00294	6.54	2.16	0.14042
Lap3	0.05156	0.00297	6.18	2.14	0.14050
WMnet 1	0.04844	0.00292	6.68	2.18	0.13270
WMnet 2	0.04945	0.00280	7.68	1.98	0.13785
WMnet 3	0.04926	0.00280	7.92	2.06	0.13853

the three groups of highly correlated covariates, where the sizes of the groups are equal. Example 4 is similar to Example 3 except that the sizes of the highlycorrelated groups of covariates are different. For Example 2, the weighted Mnet shows similar predcition peformance to the sparse Laplacian estimator, while the Mnet is slightly inferior. For Example 3 and 4, the weighted Mnet outperforms the sparse Laplacian and Mnet estimators in terms of prediction accuracy as well as variable selectivity.

표 5.3: Example 3 result

Methods	**Error**	**s.e.**	**T.P.**	**F.P.**	**n.hood**
Enet	0.07544	0.00328	11.14	4.98	0.15574
Mnet	0.06362	0.00328	5.76	2.38	0.13872
Lap1	0.06624	0.00338	5.96	2.96	0.14445
Lap2	0.06404	0.00333	6.56	2.22	0.13945
Lap3	0.06430	0.00329	6.56	2.14	0.13906
WMnet 1	0.06135	0.00337	6.22	2.2	0.12830
WMnet 2	0.06145	0.00350	8.16	2.02	0.13054
WMnet 3	0.06119	0.00345	8.16	2.04	0.13084

5.2 Real data analysis

We analyze the three real data sets sonar, voice and arrhythmia which are found in UCI machine learning repository (http://archive.ics.uci.edu/ml/). These data are the data for classificaion and the dscriptions of the data sets are given in Table 5.4.

The sonar data consist of two level responses, that are 'mines' and 'rocks'. The covariates are bouncing sonar signals under various conditions.

Voice data assess whether voice rehabilitation treatment lead to phonations considered 'acceptable' or 'unacceptable' which is the binary class classification

표 5.4: Real data description

data set	**n**	**p**	**y(0/1)**	**training set n**
sonar	208	60	111/97	104
voice	126	310	42/84	63
arrhythmia	452	274	207/245	226

problem.

The arrhythmia data distinguish between the presence and absence of cardiac arrhythmia and classify it in one of the 16 groups.

We divide each data set into three disjoint data at random such that 50% of the data set is a training data. It leads to voice and arrhythmia data to be high-dimensional data, which p is larger than n.

The regularization parameters is selected based on negative log likelihood of the validation sets.

In case of voice and arrhythmia data, the number of variables are less than observations in the training set. It means those data are high dimensional data.

Tables 5.5, 5.6 and 5.7 report the test negative loglikelihood value (n.hood) and the misclassification error (Error), the standard error of the misclassification error (s.e.), the average number of nonzero variables included in the selected model (nonzero coeff.) based on 50 repetitions of random partition.

표 5.5: Sonar data analysis result

Methods	**Error**	**s.e.**	**nonzero coeff.**	**n.hood**
Enet	0.25742	0.00615	13.52	0.27361
Mnet	0.25742	0.00645	13.62	0.27545
Lap1	0.25839	0.00602	10.36	0.28102
Lap2	0.26032	0.00600	9	0.28512
Lap3	0.26290	0.00610	8.92	0.28560
WMnet 1	0.25645	0.00605	13.24	0.27335
WMnet 2	0.26967	0.00691	19.88	0.27288
WMnet 3	0.25645	0.00605	13.32	0.27353

In sonar data analysis, WMnet methods are the better than other methods in negative loglikelihood value over test data. The WMnet1 and WMnet3 methods are the better than other methods in prediction accuracy.

In voice data analysis, The WMnet1 and Lap1 methods are the better than other method in prediction accuracy and negative loglikelihood value. The WMnet1 give similar performance with less less variable than Lap1 Lap2 and Lap3 are the worse than other competitors

In arrhythmia data analysis, Lap1 is the better than other in prediction accuracy, but WMnet methods are the better than other methods in negative

표 5.6: Voice data analysis result

Methods	**Error**	**s.e.**	**nonzero coeff.**	**n.hood**
Enet	0.16757	0.00711	11.867	0.22940
Mnet	0.16396	0.00685	11.733	0.2325
Lap1	0.15496	0.00685	18.333	0.22140
Lap2	0.16757	0.00832	11.667	0.27142
Lap3	0.16577	0.00776	16.333	0.28282
WMnet 1	0.15496	0.00757	11.133	0.22959
WMnet 2	0.16036	0.00729	11.6	0.2314
WMnet 3	0.16397	0.00757	11.2	0.24815

loglikelihood value over test data.

표 5.7: Arrhythmia data analysis result

Methods	**Error**	**s.e.**	**nonzero coeff.**	**n.hood**
Enet	0.25482	0.00395	21.02	0.27129
Mnet	0.25437	0.00388	21.14	0.27135
Lap1	0.24978	0.00491	26.2	0.27884
Lap2	0.25615	0.00475	18.98	0.29286
Lap3	0.25511	0.00473	19.68	0.26220
WMnet 1	0.25541	0.00419	20.56	0.27191
WMnet 2	0.25244	0.00446	23.58	0.27163
WMnet 3	0.25482	0.00413	20.96	0.27121

제 6 장

Concluding remarks

In this thesis, we studied theoretical properties of the weighted Mnet estimator with twice differentiable general losses. First, we showed that the weighted l_2 penalty is approximately equal to Laplacian penalty. It means that the weighted Mnet penalty preserves most desirable properties of the sparse Laplacian penalty. Second, we showed the oracle property of the weighted Mnet estimator on high dimensional models. That is the weighted Mnet estimator is asymptotically consistent estimator. Our regularity conditions for the oracle property are weaker and more transparent than those for the Mnet and sparse Laplacian penalties.

Numerical studies showed that the weighted Mnet estimator is a promising alternative to the other competitors. In particular, when there are several

groups of highly correlated covariates and the group sizes differ, the weighted Mnet estimator outperforms the Mnet and sparse Laplacian estimators.

Appendix

Proof of the Theorem 1

By letting $\eta = \lambda/(1+\lambda)$, and $\mathbf{A} = H(\boldsymbol{\beta}^c)$, we have

$$
\begin{aligned}
& \tilde{\boldsymbol{\beta}} = \text{argmin}_{\boldsymbol{\beta}} \tilde{\mathcal{L}}(\boldsymbol{\beta}) + \frac{1}{2}\frac{\lambda}{1+\lambda}\boldsymbol{\beta}'(\mathbf{D}-\mathbf{A})\boldsymbol{\beta}. \\
= \;& \text{argmin}_{\boldsymbol{\beta}} \tilde{\mathcal{L}}(\boldsymbol{\beta}) + \frac{1}{2}\frac{\lambda}{1+\lambda}\boldsymbol{\beta}'(\mathbf{D}-\mathbf{A})\boldsymbol{\beta}. \\
= \;& \text{argmin}_{\boldsymbol{\beta}} \mathcal{L}(\boldsymbol{\beta}^c) + \nabla(\boldsymbol{\beta}^c)(\boldsymbol{\beta}-\boldsymbol{\beta}^c) + \frac{1}{2}(\boldsymbol{\beta}-\boldsymbol{\beta}^c)'H(\boldsymbol{\beta}^c)(\boldsymbol{\beta}-\boldsymbol{\beta}^c) \\
& + \frac{1}{2}\eta\boldsymbol{\beta}'(\mathbf{D}-\mathbf{A})\boldsymbol{\beta} \\
= \;& \text{argmin}_{\boldsymbol{\beta}} \mathcal{L}(\boldsymbol{\beta}^c) + \nabla(\boldsymbol{\beta}^c)(\boldsymbol{\beta}-\boldsymbol{\beta}^c) + \frac{1}{2}(\boldsymbol{\beta}-\boldsymbol{\beta}^c)'H(\boldsymbol{\beta}^c)(\boldsymbol{\beta}-\boldsymbol{\beta}^c) \\
& - \frac{1}{2}\eta\boldsymbol{\beta}'H(\boldsymbol{\beta}^c)\boldsymbol{\beta} + \frac{1}{2}\eta\sum_{j=1}^{p} w_j\beta_j^2 \\
= \;& \text{argmin}_{\boldsymbol{\beta}} \nabla(\boldsymbol{\beta}^c)\boldsymbol{\beta} + \frac{1}{2}\boldsymbol{\beta}'H(\boldsymbol{\beta}^c)\boldsymbol{\beta} - \boldsymbol{\beta}^c H(\boldsymbol{\beta}^c)\boldsymbol{\beta} - \frac{1}{2}\eta\boldsymbol{\beta}'H(\boldsymbol{\beta}^c)\boldsymbol{\beta} + \frac{1}{2}\eta\sum_{j=1}^{p} w_j\beta_j^2 \\
& + \mathcal{L}(\boldsymbol{\beta}^c) + \nabla(\boldsymbol{\beta}^c)\boldsymbol{\beta}^c + \frac{1}{2}\boldsymbol{\beta}^c H(\boldsymbol{\beta}^c)\boldsymbol{\beta}^c \\
= \;& \text{argmin}_{\boldsymbol{\beta}} (\nabla(\boldsymbol{\beta}^c) - \boldsymbol{\beta}^c H(\boldsymbol{\beta}^c))\boldsymbol{\beta} + \frac{1}{2}(1-\eta)\boldsymbol{\beta}'H(\boldsymbol{\beta}^c)\boldsymbol{\beta} + \frac{1}{2}\eta\sum_{j=1}^{p} w_j\beta_j^2 \\
& + \mathcal{L}(\boldsymbol{\beta}^c) - \nabla(\boldsymbol{\beta}^c)\boldsymbol{\beta}^c + \frac{1}{2}\boldsymbol{\beta}^c H(\boldsymbol{\beta}^c)\boldsymbol{\beta}^c
\end{aligned}
$$

Let $A(\boldsymbol{\beta}^c) = (\nabla(\boldsymbol{\beta}^c) - \boldsymbol{\beta}^c H(\boldsymbol{\beta}^c))$

$$
\begin{aligned}
&= \text{argmin}_{\boldsymbol{\beta}} \frac{1}{2} \frac{1}{1-\eta} H(\boldsymbol{\beta}^c)^{-1} \{2(1-\eta) H(\boldsymbol{\beta}^c) A(\boldsymbol{\beta}^c) \boldsymbol{\beta} + (1-\eta)^2 H(\boldsymbol{\beta}^c) \boldsymbol{\beta} H(\boldsymbol{\beta}^c) \boldsymbol{\beta})\} \\
&\quad + \frac{1}{2} \eta \sum_{j=1}^{p} w_j \beta_j^2 \\
&\quad + \mathcal{L}(\boldsymbol{\beta}^c) - \nabla(\boldsymbol{\beta}^c) \boldsymbol{\beta}^c + \frac{1}{2} \boldsymbol{\beta}^c H(\boldsymbol{\beta}^c) \boldsymbol{\beta}^c \\
&= \text{argmin}_{\boldsymbol{\beta}} \frac{1}{2} \frac{1}{1-\eta} H(\boldsymbol{\beta}^c)^{-1} \{A(\boldsymbol{\beta}^c) + (1-\eta) H(\boldsymbol{\beta}^c) \boldsymbol{\beta})\}^2 + \frac{1}{2} \eta \sum_{j=1}^{p} w_j \beta_j^2 \\
&\quad + \mathcal{L}(\boldsymbol{\beta}^c) - \nabla(\boldsymbol{\beta}^c) \boldsymbol{\beta}^c + \frac{1}{2} \boldsymbol{\beta}^c H(\boldsymbol{\beta}^c) \boldsymbol{\beta}^c - \frac{1}{2} \frac{1}{1-\eta} A(\boldsymbol{\beta}^c)^2 \\
&= \text{argmin}_{\boldsymbol{\beta}} \frac{1}{2} \frac{1}{1-\eta} H(\boldsymbol{\beta}^c)^{-1} \{A(\boldsymbol{\beta}^c) + (1-\eta) H(\boldsymbol{\beta}^c) \boldsymbol{\beta})\}^2 + \frac{1}{2} \eta \sum_{j=1}^{p} w_j \beta_j^2
\end{aligned}
$$

By transforming $\gamma = (1-\eta)\boldsymbol{\beta}$

$$
\begin{aligned}
\hat{\gamma} &= \text{argmin}_{\gamma} \frac{1}{1-\eta} \{\frac{1}{2} H(\boldsymbol{\beta}^c)^{-1} \{A(\boldsymbol{\beta}^c) + H(\boldsymbol{\beta}^c) \gamma)\}^2 + \frac{1}{2} \frac{\eta}{1-\eta} \sum_{j=1}^{p} w_j \gamma_j^2\} \\
&= \text{argmin}_{\gamma} (1+\lambda) \{\frac{1}{2} H(\boldsymbol{\beta}^c)^{-1} \{A(\boldsymbol{\beta}^c) + H(\boldsymbol{\beta}^c) \gamma)\}^2 + \frac{1}{2} \lambda \sum_{j=1}^{p} w_j \gamma_j^2\}
\end{aligned}
$$

Note that

$$
\begin{aligned}
& \text{argmin}_{\boldsymbol{\beta}} \tilde{\mathcal{L}}(\boldsymbol{\beta}) + \frac{1}{2}\lambda \sum_{k=1}^{p} w_k \beta_k^2 \\
= \; & \text{argmin}_{\boldsymbol{\beta}} \mathcal{L}(\boldsymbol{\beta}^c) + \nabla(\boldsymbol{\beta}^c)(\boldsymbol{\beta} - \boldsymbol{\beta}^c) + \frac{1}{2}(\boldsymbol{\beta} - \boldsymbol{\beta}^c)' H(\boldsymbol{\beta}^c)(\boldsymbol{\beta} - \boldsymbol{\beta}^c) \\
& + \frac{1}{2}\lambda \sum_{k=1}^{p} w_k \beta_k^2 \\
= \; & \text{argmin}_{\boldsymbol{\beta}} (\nabla(\boldsymbol{\beta}^c) - \boldsymbol{\beta}^c H(\boldsymbol{\beta}^c))\boldsymbol{\beta} + \frac{1}{2}\boldsymbol{\beta}' H(\boldsymbol{\beta}^c)\boldsymbol{\beta} + \frac{1}{2}\lambda \sum_{k=1}^{p} w_k \beta_k^2 \\
& + \mathcal{L}(\boldsymbol{\beta}^c) - \nabla(\boldsymbol{\beta}^c)\boldsymbol{\beta}^c + \frac{1}{2}\boldsymbol{\beta}^c H(\boldsymbol{\beta}^c)\boldsymbol{\beta}^c \\
= \; & \text{argmin}_{\boldsymbol{\beta}} \frac{1}{2} H(\boldsymbol{\beta}^c)^{-1} \{A(\boldsymbol{\beta}^c) + H(\boldsymbol{\beta}^c)\boldsymbol{\beta})\}^2 + \frac{1}{2}\lambda \sum_{j=k}^{p} w_k \beta_k^2 \\
& + \mathcal{L}(\boldsymbol{\beta}^c) - \nabla(\boldsymbol{\beta}^c)\boldsymbol{\beta}^c + \frac{1}{2}\boldsymbol{\beta}^c H(\boldsymbol{\beta}^c)\boldsymbol{\beta}^c - \frac{1}{2} A(\boldsymbol{\beta}^c)^2 \\
\hat{\boldsymbol{\beta}}^{w2} \; = \; & \text{argmin}_{\boldsymbol{\beta}} \frac{1}{2} H(\boldsymbol{\beta}^c)^{-1} \{A(\boldsymbol{\beta}^c) + H(\boldsymbol{\beta}^c)\boldsymbol{\beta})\}^2 + \frac{1}{2}\lambda \sum_{j=k}^{p} w_k \beta_k^2
\end{aligned}
$$

We have $\hat{\boldsymbol{\beta}}^{w2} = \hat{\gamma}(1+\lambda)$

Proof of the Theorem 2

Some notations are follows.

- $\nabla^* = \sup_{j \in \mathcal{A}^{*c}} |\nabla(\hat{\beta}^o)_j|$.
- $C_\lambda(\boldsymbol{\beta}) = \frac{1}{n}\sum_{i=1}^n l(y_i, \mathbf{x}_i'\boldsymbol{\beta}) + \lambda_2 \sum_{k=1}^p w_k \beta_k^2 + J_{\lambda_1}(\boldsymbol{\beta}_{\boldsymbol{w},\lambda_2})$
- $w_* = \min_k w_k$

Proof.

Taylor expansion implies that

$$\begin{aligned}
\mathcal{L}(\boldsymbol{\beta}) - \mathcal{L}(\hat{\boldsymbol{\beta}}^o(\lambda_2)) &= \frac{1}{n}\sum_{i=1}^n l(y_i, \mathbf{x}_i'\boldsymbol{\beta}) - \frac{1}{n}\sum_{i=1}^n l(y_i, \mathbf{x}_i'\hat{\boldsymbol{\beta}}^o(\lambda_2)) \\
&= (\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}^o(\lambda_2))'\nabla(\hat{\boldsymbol{\beta}}^o(\lambda_2)) + \frac{1}{2}(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}^o(\lambda_2))' H(\tilde{\boldsymbol{\beta}})(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}^o(\lambda_2))
\end{aligned}$$

where for some $\tilde{\boldsymbol{\beta}} \in \Theta$.

Hence,

$$\begin{aligned}
C_\lambda(\boldsymbol{\beta}) - C_\lambda(\hat{\boldsymbol{\beta}}^o(\lambda_2)) &= (\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}^o(\lambda_2))'\nabla_{\lambda_2}(\hat{\boldsymbol{\beta}}^o(\lambda_2)) + \frac{1}{2}(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}^o(\lambda_2))' H_{\lambda_2}(\tilde{\boldsymbol{\beta}})(\boldsymbol{\beta} - \hat{\boldsymbol{\beta}}^o(\lambda_2)) \\
&\quad + \sum_{j=1}^p \Big\{ J_{\lambda,w}(|\beta_j|) - J_{\lambda,w}(|\hat{\beta}^o(\lambda_2)_j|) \Big\} \\
&\leq \sum_{j=1}^p U_j,
\end{aligned}$$

where $\nabla_{\lambda_2}(\boldsymbol{\beta}) = \nabla(\boldsymbol{\beta}) + 2\lambda_2 \mathbf{W}\beta, H_{\lambda_2}(\boldsymbol{\beta}) = H(\boldsymbol{\beta}) + 2\lambda_2\mathbf{W}$, $J_{\lambda,w}(|\beta_j|) = J_{\lambda_1}\big\{(1+\lambda_2)|w_j\beta_j|\big\}$ and

$$U_j = (\beta_j - \hat{\beta}^o(\lambda_2)_j)\nabla_{\lambda_2}(\hat{\beta}^o(\lambda_2))_j + \rho_*(\beta_j - \hat{\beta}^o(\lambda_2)_j)^2 + J_{\lambda,w}(|\beta_j|) - J_{\lambda,w}(|\hat{\beta}^o(\lambda_2)_j|).$$

Consider a case where $j \in \mathcal{A}^{*c}$.

Condition A1-3 implies that

$$J_{\lambda,w}(|\beta_j|) \geq \frac{\lambda_1}{2}(1+\lambda_2)|w_j\beta_j|$$

for $(1+\lambda_2)|w_j\beta_j| \leq a\lambda_1/2$.

Lemma 1 proves that

$$\sup_{j\in\mathcal{A}^{*c}} |\nabla_{\lambda_2}(\hat{\beta}^o(\lambda_2))_j| \leq \nabla^* + 2\lambda_2 w^* \frac{\rho^* + 2\lambda_2 w^*}{\rho_* + \lambda_2 w_*} \|\hat{\beta}^o\|$$

and hence $\lambda_1 >> \sup_{j\in\mathcal{A}^{*c}} |\nabla_{\lambda_2}(\hat{\beta}^o(\lambda_2))_j|$.

When $(1+\lambda_2)|w_j\beta_j| \leq a\lambda_2/2$,

$$U_j \geq \left\{ \frac{\lambda_1(1+\lambda_2)w_j}{2} - \nabla_{\lambda_2}(\hat{\beta}^o(\lambda_2))_j \right\} |\beta_j| + \rho_*\beta_j^2 > 0.$$

On the other hand, when $(1+\lambda_2)|w_j\beta_j| > a\lambda_1/2$, we have

$$U_j \geq -|\nabla_{\lambda_2}(\hat{\beta}^o(\lambda_2))_j||\beta_j| + \rho_*\beta_j^2 \geq |\beta_j| \left(\frac{\rho_* w_* \lambda_1}{2(1+\lambda_2)w^*} - |\nabla_{\lambda_2}(\hat{\beta}^o(\lambda_2))_j| \right) > 0$$

since $\lambda_1 >> \sup_{j\in\mathcal{A}^{*c}} |\nabla_{\lambda_2}(\hat{\beta}^o(\lambda_2))_j|$.

Consider a case of $j \in \mathcal{A}^*$.

Note that $\nabla(\hat{\beta}^o(\lambda_2))_{\lambda_2,j} = 0$.

Lemma 2 implies

$$\min_{j \in \mathcal{A}_*} |\hat{\beta}^o(\lambda_2)_j - \hat{\beta}^o_j| \leq \frac{2\lambda_2 w^*}{\rho_* + \lambda_2 w_*} \|\hat{\beta}^o\|.$$

Since $\min_{j \in \mathcal{A}_*} |\hat{\beta}^o_j| >> \lambda_1 >> \lambda_2 w^* \|\hat{\beta}^o\|$, we have $\min_{j \in \mathcal{A}_*} |\hat{\beta}^o(\lambda_2)_j| >> \lambda_1$.

When $(1+\lambda_2)|w_j \beta_j| \leq a\lambda_1$,

$$\begin{aligned} w_j &\geq (\rho_* + 2\lambda_2 w_*)(|\hat{\beta}^o(\lambda_2)_j| - |\beta_j|)^2 - \lambda_1(1+\lambda_2)w^* \big||\hat{\beta}^o(\lambda_2)_j| - |\beta_j|\big| \\ &\geq (\rho^* + 2\lambda_2)\big||\hat{\beta}^o(\lambda_2)_j| - |\beta_j|\big| \left\{ \big||\hat{\beta}^o(\lambda_2)_j| - |\beta_j|\big| - \frac{\lambda_1(1+\lambda_2)w^*}{\rho_* + 2\lambda_2} \right\} > 0 \end{aligned}$$

since $\big||\hat{\beta}^o(\lambda_2)_j| - |\beta_j|\big| >> \lambda_1$.

When $(1+\lambda_2)w_j|\beta_j| > a\lambda_1$, $w_j = (\rho_* + 2\lambda_2)(\beta_j - \hat{\beta}^o(\lambda_2)_j)^2 > 0$ unless $\beta_j = \hat{\beta}^o(\lambda_2)_j$. To sum up, we have shown that $\hat{\beta}^{wM}(\lambda) = (1+\lambda_2)\hat{\beta}^o(\lambda_2)$ with probability converging to 1.

Lemma 1

$$\sup_{j\in\mathcal{A}^{*c}} |\nabla_{\lambda_2}(\hat{\beta}^o(\lambda_2))_j| \leq \nabla^* + 2\lambda_2 w^* \frac{\rho^* + 2\lambda_2 w^*}{\rho_* + \lambda_2 w_*} \|\hat{\beta}^o\|.$$

Proof. For $j \in \mathcal{A}^{*c}$, Taylor expansion yields

$$\nabla_{\lambda_2}(\hat{\beta}^o(\lambda_2))_j = \nabla(\hat{\beta}^o)_j + e_j' H_{\lambda_2}(\tilde{\beta})(\hat{\beta}^o(\lambda_2) - \hat{\beta}^o)$$

for some $\tilde{\beta} \in \Theta$. Hence

$$\nabla_{\lambda_2}(\hat{\beta}^o(\lambda_2))_j| \leq \nabla^* + (\rho^* + 2\lambda_2 w^*)\|\hat{\beta}^o(\lambda_2) - \hat{\beta}^o\|.$$

The proof can be complete by Lemma 2.

Lemma 2

$$\|\hat{\beta}^o(\lambda_2) - \hat{\beta}^o\| \leq \frac{2\lambda_2 w^*}{\rho_* + \lambda_2 w_*} \|\hat{\beta}^o\|.$$

Proof. Let $Q_{\lambda_2}(\beta) = \mathcal{L}(\beta) + \lambda_2 \sum_{j=1}^p \beta_j^2$. Taylor expansion yields that there exists $\tilde{\beta} \in \Theta$ such that

$$Q_{\lambda_2}(\hat{\beta}^o(\lambda_2)) - Q_{\lambda_2}(\hat{\beta}^o) = (\hat{\beta}^o(\lambda_2) - \hat{\beta}^o)' H(\tilde{\beta})(\hat{\beta}^o(\lambda_2) - \hat{\beta}^o) + \lambda_2 \left\{ \|\hat{\beta}^o(\lambda_2)\|_w^2 - \|\hat{\beta}^o\|_w^2 \right\} < 0,$$

where $\|\beta\|_w^2 = \sum_{j=1}^p w_j \beta_j^2$. Note that

$$\|\hat{\beta}^o(\lambda_2)\|^2 -_w \|\hat{\beta}^o\|_w^2 \geq \|\hat{\beta}^o(\lambda_2) - \hat{\beta}^o\|_w^2 - 2\|\hat{\beta}^o\|_w \|\hat{\beta}^o(\lambda_2) - \hat{\beta}^o\|_w.$$

Hence, (6) implies

$$(\rho_* + \lambda_2 w_*)\|\hat{\beta}^o(\lambda_2) - \hat{\beta}^o\|^2 - 2\lambda_2 w^*\|\hat{\beta}^o\|\|\hat{\beta}^o(\lambda_2) - \hat{\beta}^o\| \leq 0,$$

which in turn implies

$$\|\hat{\beta}^o(\lambda_2) - \hat{\beta}^o\| \leq \frac{2\lambda_2 w^*}{\rho_* + \lambda_2 w_*}\|\hat{\beta}^o\|.$$

참고 문헌

H. Akaike. Information theory and an extension of the maximum likelihood principle. 1:267–281, 1973.

P. Bickel, Y. Ritov, and A. Tsybakov. Simultaneous analysis of lasso and danzig selector. Annals of Statistics, 37:1705–1732, 2009.

H.D. Bondell and B.J. Reich. Simultaneous regression shrinkage, variable selection, and supervised clustering of predictors with oscar. Biometrics, 64: 115–123, 2008.

L. Breiman. Heuristics of instability and stabilization in model selection. The Annals of Statistics, 24(6):2350–2383, 1996.

B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani. Least angle regression. The Annals of statistics, 32(2):407–499, 2004.

J. Fan and R. Li. Variable selection via nonconcave penalized likelihood and

its oracle properties. Journal of the American Statistical Association, 96: 1348–1360, 2001.

J. Fan and H. Peng. Nonconcave penalized likelihood with a diverging number of parameters. The Annals of Statistics, 32:928–961, 2004.

J.H. Friedman, Hofling H. Hastie, T., and R. Tibshirani. Pathwise coordinate optimization. Annals of Applied Statistics, 1:302–332, 2007.

Wenjiang J Fu. Penalized regressions: the bridge versus the lasso. Journal of computational and graphical statistics, 7(3):397–416, 1998.

A.E. Hoerl and R.W. Kennard. Ridge regression: Biased estimation for nonorthogonal problems. Technometrics, pages 55–67, 1970.

J. Huang, J.L. Horowitz, and S. Ma. Asymptotic properties of bridge estimators in sparse high-dimensional regression models. The Annals of Statistics, 36(2):587–613, 2008.

J. Huang, P. Breheny, S. Ma, and C.H. Zhang. The mnet method for variable selection. Technical Report 403, Department of Statistics and Actuarial Science, University of Iowa, 2010.

J. Huang, S. Ma, H. Li, and C.H. Zhang. The sparse Laplacian shrinkage

estimator for high-dimensional regression. Annals of Statistics, 39:2021–2046, 2011.

Y. Kim and J. Jeon. weighted mnet penalty for highly correlated. Department of Statistics, Seoul National University, 2014.

Y. Kim, H. Choi, and H.S. Oh. Smoothly clipped absolute deviation on high dimensions. Journal of the American Statistical Association, 103(484):1665–1673, 2008.

K. Knight and W. Fu. Asymptotics for lasso-type estimators. Annals of Statistics, pages 1356–1378, 2000.

S. Kwon and Y. Kim. Large sample properties of the smoothly clipped absolute deviation penalized maximum likelihood estimation on high dimensions. Statisca Sinica, 2011.

C.L. Mallows. Some comments on cp. Technometrics, pages 661–675, 1973.

N. Meinshausen and P. Bühlmann. High-dimensional graphs and variable selection with the lasso. The Annals of Statistics, 34(3):1436–1462, 2006.

Garvesh Raskutti, Martin J Wainwright, and Bin Yu. Minimax rates of estimation for high-dimensional linear regression over-balls. Information Theory, IEEE Transactions on, 57(10):6976–6994, 2011.

S. Rosset and J. Zhu. Piecewise linear regularized solution paths. The Annals of Statistics, 35(3):1012–1030, 2007.

G. Schwarz. Estimating the dimension of a model. The annals of statistics, 6 (2):461–464, 1978.

Y. She. Sparse regression with exact clustering. Electronic Jorunal of Statistics, 4:1055–1096, 2010.

X. Shen and H. Huang. Grouping pursuit in regression. Journal of the American Statistical Association, 105:727–739, 2010.

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

Ryan Joseph Tibshirani. The solution path of the generalized lasso. Stanford University, 2011.

X. Wu, X. Shen, and C.J. Geyer. Adaptive regularization using the entire solution surface. Biometrika, 96:513–527, 2009.

A.L. Yuille and A. Rangarajan. The concave-convex procedure. Neural Computation, 15(4):915–936, 2003.

C.-H. Zhang. Nearly unbiased variable selection under minimax concave penalty. The Annals of Statistics, 38:894–942, 2010.

Cun-Hui Zhang and Jian Huang. The sparsity and bias of the lasso selection in high-dimensional linear regression. The Annals of Statistics, pages 1567–1594, 2008.

P. Zhao and B. Yu. On model selection consistency of lasso. The Journal of Machine Learning Research, 7:2541–2563, 2006.

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

H. Zou and T. Hastie. Regularization and variable selection via the elastic net. Journal of the Royal Statistical Society, Ser. B, 67:301–320, 2005.

국문초록

많은 회귀분석 문제에서 설명변수들 간에 자연스럽게 상관관계가 있는 것을 볼 수 있다. 유의미한 설명변수을 적절히 선택하는 것은 모형에 대한 이해나 예측성능을 향상시키는데 도움을 준다. 본 연구는 고차원자료의 일반화 선형모형에서 변수를 선택하는 문제에 대한 연구이다.

본 학위논문에서는 상관관계가 높은 고차원 일반화 선형모형 자료에서 WMnet 벌점함수의 이론적 성질과 수치적 연구를 통해 유용성을 보였다. 이론적 성질로는 의미없는 그룹에서의 변수들을 버리고, 유의미한 변수만을 가지고 구한 추정량인 '신의 추정량'과 WMnet 추정량이 같아질 확률이 점근적으로 1로 수렴함을 증명하였다. 또한 WMnet 추정량을 구하기 위한 일반화된 알고리즘을 기술하였다. 수치적 연구를 통해서는 상관관계가 높은 자료에 대해 WMnet방법이 다른 기존의 방법들에 비해 좋은 성능을 나타냄을 보일 수 있다.

주요어 : WMnet 제한함수, 신의 성질, 일반화 선형모형

학 번 : 2009 – 30069