A mixture of local and quadratic approximation variable selection algorithm in nonconcave penalized regression

Assi N'Guessan, Ibrahim Sidi Zakari, Assi Mkhadri

A mixture of local and quadratic approximation variable selection algorithm in nonconcave penalized regression

Sidi Zakari Ibrahim* — Mkhadri Abdallah* — N'Guessan Assi**

* Departement of Mathematics
University Cadi Ayyad
40000 Marrakech
MAROC
ibrahim.sidizakari@edu.uca.ma, mkhadri@uca.ma

** Departement of Mathematics
University of Sciences and Technologies of Lille
59650 Villeneuve d'Ascq
FRANCE
assi.nguessan@polytech-lille.fr

ABSTRACT. We consider the problem of variable selection via penalized likelihood using nonconvex penalty functions. To maximize the non-differentiable and nonconcave objective function, an algorithm based on local linear approximation and which adopts a naturally sparse representation was recently proposed. However, although it has promising theoretical properties, it inherits some drawbacks of Lasso in high dimensional setting. To overcome these drawbacks, we propose an algorithm (MLLQA) for maximizing the penalized likelihood for a large class of nonconvex penalty functions. The convergence property of MLLQA and oracle property of one-step MLLQA estimator are established. Some simulations and application to a real data set are also presented.

RÉSUMÉ. Nous considérons le problème de sélection de variables via la vraisemblance pénalisée en utilisant des fonctions de pénalité non convexes. Afin de maximiser la fonction objectif qui est non différentiable et non concave, un algorithme basé sur une approximation linéaire locale et fournissant un estimateur éparse été récemment proposé. Cependant, il hérite de certains inconvénients du Lasso en grande dimension. Afin d'y remédier, nous proposons un algorithme (MLLQA) pour maximiser la vraisemblance pénalisée pour une large classe de fonctions de pénalité non convexes. La propriété de convergence du MLLQA ainsi que la propriété oracle de l'estimateur obtenu après une itération ont été établies. Des simulations ainsi qu'une application sur données réelles sont également présentées.

KEYWORDS : Regression, Variable selection, SCAD penalty, LARS, LLA and LQA algorithms.

MOTS-CLÉS : Régression, Sélection de variables, pénalité SCAD, algorithmes LARS, LLA et LQA.

Received, July 5, 2012
Revised, January 16, 2013
Accepted, January 16, 2013

ARIMA Journal, vol. 16 (2013), pp. 29-46.

1. Introduction

Variable selection plays an important role in statistical modeling. In genomics and preteomics studies, functional MRI, tumor classification and signal processing[14], it is very common that a large number p of candidate predictors are included in the model. However, when p is large, selection of a small number of predictors that contribute to the response leads often to a parsimonious model. It amounts to assuming that the true model has a sparse representation, i. e. some components of the parameter vector $\boldsymbol{\beta}$ of regression coefficients are exactly zero. In this setting, variable selection can improve on both estimation accuracy and interpretation. Our objective is to find the set $\mathcal{A}$ of the nonzero components of $\boldsymbol{\beta}$ and to estimate the true corresponding coefficients.

Recently, variable selection for high dimensional data has received a lot of attention. In the last decade interest has focused on penalized regression methods which implement both variable selection and coefficient shrinkage in a single procedure. The most well known of these procedures are Lasso[12, 1] and SCAD[3], which have good computational and statistical properties. The Lasso sparse estimates minimize the penalized least squares with ℓ_1 penalty. While SCAD is presented as a unified approach, via nonconcave penalized likelihood which simultaneously performs variable selection and coefficient estimation. By a judicious choice of nonconvex penalty function, SCAD keeps many merits of the best subset selection and ridge regression. A similar nonconvex penalty MCP[13] has been proposed to overcome the Lasso bias[10]. SCAD and MCP enjoy the oracle property, that is, the SCAD and MCP estimators can perform as well as the oracle if the penalization parameter is appropriately chosen.

The SCAD (and also MCP) penalty is nonconvex, and consequently it is hard to compute the solution of the optimization problem. To facilitate the use of Newton-Raphson algorithm, Fan and Li[3] proposed to approximate the nonconvex penalty by the local quadratic approximation (LQA). However, the drawback of this approximation is that the estimate of the regression coefficient has to end up being 0 once it reached 0 at any step of the LQA algorithm. So, the LQA algorithm inherits the drawback of backward stepwise variable selection: if a covariate is eliminated at any step in the LQA algorithm, it will necessarily be deleted from the final selected model. To alleviate this problem, Hunter and Li[6] proposed a minorize-maximize (MM) algorithm to compute the nonconcave penalized estimator. In this algorithm, the LQA approximation is improved with a small perturbation $\epsilon > 0$ to overcome the non-differentiability at zero.

On the other hand, Zou and Li[15] proposed a local linear approximation (LLA) algorithm that recasts the computation of nonconcave penalized likelihood problems into a sequence of penalized ℓ_1-likelihood problems. The LLA algorithm enjoys some significant advantages over LQA and the perturbed LQA and produces a sparse estimates via continuous penalization. Moreover, the efficient LARS algorithm[2] for solving Lasso is used to compute the one-step LLA estimator. Consequently, the LLA algorithm will inherit similar limitations of Lasso in high dimensional setting: for $p > n$, it selects at most n variables before it puts all coefficients to zero and a second limitation is that group of variables can not enter in the same time with Lasso.

In this paper, we propose an efficient one-step sparse estimation procedure in nonconcave penalized likelihood models, which is based on the mixture of local linear and quadratic approximation penalties (MLLQA). The new iterative MLLQA enjoys the advantages of both LLA and the perturbed LQA algorithms. As with LLA, MLLQA does not delete any small coefficient and it produces a sparse estimates via continuous penal-

ization. Its convergence property is shown and the oracle property of one-step MLLQA estimator is established. Computationally, we take advantage of the efficient coordinate descent algorithm for Lasso penalized regression to compute the one-step MLLQA estimator in high dimension.

In Section 2, we present the local linear and quadratic approximation algorithms for SCAD penalty. In Section 3, we present our mixture of the local linear and quadratic approximation algorithm for SCAD penalty and study its various properties. In particular, we show that the MLLQA algorithm is an instance of MM algorithms[5] which converges to a stationary point of the likelihood solutions. In Section 4, we study the statistical properties of the one-step MLLQA estimator. In particular, we show that the one-step MLLQA estimator enjoys the oracle property: consistence of selection and asymptotical normality. Numerical study is presented in Section 5 and we end with a brief discussion in Section 6.

2. Linear and quadratic approximation algorithms

In this Section, we consider the problem of variable selection in generalized linear model based on penalized likelihood approach. Two useful nonconvex penalties (SCAD and MCP) and various local linear and quadratic approximation algorithms for computing the maximum penalized likelihood are briefly presented.

2.1. Penalized likelihood with concave penalty

Let $(\mathbf{x}^i, y_i), i = 1, \ldots, n$ be n i. i. d. predictive-response observation pairs that are assumed to be a random sample where $\mathbf{x}^i \in I\!R^p$, and $y_i \in I\!R$. We assume that the observation y_i depend on $\mathbf{x}^i$ through a linear combination of $(\mathbf{x}^i)^t\boldsymbol{\beta}, \boldsymbol{\beta} \in I\!R^p$ and t stands for the transpose. That is, we assume that given $\mathbf{x}^i$, y_i has the density $f_i(g((\mathbf{x}^i)^t\beta), y_i)$ where g is a known link function. The conditional log-likelihood given $\mathbf{x}^i$ can be written as

$$\ell_i(\boldsymbol{\beta}) = \ell_i(\boldsymbol{\beta}, \phi) = \ell_i((\mathbf{x}^i)^t\boldsymbol{\beta}, y_i, \phi) \tag{1}$$

where ϕ is a dispersion parameter which is assumed to be known. Our objective is the estimation of the parameter vector $\boldsymbol{\beta}$ and the identification of the subset model.

We consider the estimating of parameter vector $\boldsymbol{\beta}$ by maximizing the penalized log-likelihood

$$P\ell(\boldsymbol{\beta}) = \sum_{i=1}^{n} \ell_i(\boldsymbol{\beta}) - n\sum_{j=1}^{p} J_\lambda(|\beta_j|), \tag{2}$$

for a penalty function $J_\lambda(.)$. In the linear model case, the penalty $J_\lambda(|\beta_j|) = \lambda|\beta_j|^\gamma, \gamma \geq 0$ leads the bridge estimator ([4]). In the same setting, when $\gamma = 1$ the penalty yields the Lasso estimator ([12]). Fan and Li (2001) proposed the SCAD penalty which is a continuously differentiable concave function defined by: $J_\lambda(0) = 0$ and for $|\beta_j| > 0$

$$J_\lambda^{'}(|\beta_j|) = \lambda\mathbf{I}(|\beta_j| \leq \lambda) + \frac{(a\lambda - |\beta_j|)_+}{a-1}\mathbf{I}(|\beta_j| > \lambda), \tag{3}$$

where $(z)_+ = \max(z, 0), a = 3.7, J_\lambda^{'}(r) \geq 0$ for $r > 0$ and $\mathbf{I}(|x| \leq \lambda) = 1$ if $|x| \leq \lambda$ and 0 otherwise. So, with the penalty (3) the penalized likelihood function (2) is a nonconcave function. The hard thresholding estimator corresponds to the penalty $J_\lambda(|\beta_j|) = \lambda^2 - (|\beta_j| - \lambda)^2\mathbf{I}(|\beta_j| < \lambda)$. Moreover, a new nonconvex penalty MCP ([13]) derived from

SCAD penalty can be easily understood by considering its derivative $J'_{\lambda_1,a}(z) = \lambda_1(1 - |z|/(a\lambda_1))_+ \text{sgn}(z)$ where $\text{sgn}(z) = -1, 0$ or 1 if $z < 0, = 0$ or > 0, respectively. It begins by applying the same rate of penalization as Lasso, but continuously relaxes that penalization, until $|z| > a\lambda_1$, until the rate of penalization drops to zero. In the literature, the penalty J_λ produces estimates with basic properties such that: unbiasedness, sparsity and continuity ([13]).

2.2. Local approximation algorithms

The function (2) is non-differentiable at the origin and nonconcave with respect to $\boldsymbol{\beta}$. Suppose given an initial value $\boldsymbol{\beta}^{(0)}$ that is close to the true value of $\boldsymbol{\beta}$. To run easily the Newton-Raphson algorithm, Fan and Li[3] propose the following quadratic approximation

$$[J_\lambda(|\beta_j|)]' = J'_\lambda(|\beta_j|)\text{sign}(\beta_j) \approx \left\{ J'_\lambda(|\beta_j^{(0)}|)/|\beta_j^{(0)}| \right\} \beta_j. \tag{4}$$

It leads to $J_\lambda(|\beta_j|) \approx J_\lambda(|\beta_j^{(0)}|) + \frac{1}{2}\left\{ J'_\lambda(|\beta_j^{(0)}|)/|\beta_j^{(0)}| \right\} (\beta_j^2 - {\beta_j^{(0)}}^2)$ for $\beta_j \approx \beta_j^{(0)}$. Then the iterative procedure LQA (Local Quadratic Approximation) solves

$$\boldsymbol{\beta}^{(k+1)} = \text{argmax}_{\boldsymbol{\beta}} \left\{ \sum_{i=1}^{n} \ell_i(\boldsymbol{\beta}) - n \sum_{i=1}^{n} \frac{J'_\lambda(|\beta_j^{(k)}|)}{2|\beta_j^{(k)}|} \beta_j^2 \right\}. \tag{5}$$

When $\beta_j^{(k)}$ is close to zero, i. e. $|\beta_j^{(k)}| < \epsilon_0$ (pre-specified value), then $\hat{\beta}_j = 0$ and delete the jth component of $\mathbf{x}^i$ from the iteration. However, LQA has two drawbacks: the choice of ϵ_0 and similarity with the backward stepwise variable selection. Hunter and Li[6] studied the convergence property of the LQA algorithm. They described a minorize-maximize (MM) algorithm[5] to compute the penalized nonconcave likelihood estimator. In this algorithm, the latter approximation (4) is improved with a small perturbation τ_0 to handle the non-differentiability at 0. This prevents the estimation from being trapped at 0. Then, the new iterative perturbed LQA algorithm solves

$$\boldsymbol{\beta}^{(k+1)} = \text{argmax}_{\boldsymbol{\beta}} \left\{ \sum_{i=1}^{n} \ell_i(\boldsymbol{\beta}) - n \sum_{j=1}^{p} \frac{J'_\lambda(|\beta_j^{(k)}|)}{2(|\beta_j^{(k)}| + \tau_0)} \beta_j^2 \right\}, \tag{6}$$

for a fixed size perturbation τ_0. Hunter and Li[6] noted that a suitable choice of the size of τ_0 is essential for the good degree of sparsity of the solution as well as the speed of convergence. To overcome the limitations of LQA algorithms, Zou and Li[15] described a new algorithm based on local linear approximation (LLA) to the penalty function:

$$J_\lambda(|\beta_j|) \approx J_\lambda(|\beta_j^{(0)}|) + J'_\lambda(|\beta_j^{(0)}|)(|\beta_j| - |\beta_j^{(0)}|) \quad \text{for} \quad \beta_j \approx \beta_j^{(0)}. \tag{7}$$

Then, the iterative LLA procedure becomes

$$\boldsymbol{\beta}^{(k+1)} = \text{argmax}_{\boldsymbol{\beta}} \left\{ \sum_{i=1}^{n} \ell_i(\boldsymbol{\beta}) - n \sum_{j=1}^{p} J'_\lambda(|\beta_j^{(k)}|)|\beta_j| \right\}. \tag{8}$$

As with Lasso, the ℓ_1 penalty in the LLA algorithm naturally leads to a sparse representation of the estimates of the vector parameter $\boldsymbol{\beta}$. So the LLA algorithm shares the good properties of Lasso in terms of computational efficiency, and therefore the efficient least

angle regression shrinkage (LARS) algorithm[2] can be used to solve the equation (8). Zou and Li[15] confirm that the LLA algorithm is numerically stable, and so, the limitations of backward variable selection can be avoided in the LLA algorithm. However, the LLA algorithm inherits the drawbacks of Lasso in high dimensional setting in the presence of strong correlated variables. Moreover, when $p > n$ Lasso will selects at least n variables[14].

In the same setting, other new algorithms have been recently proposed to find a minimizer of the SCAD (or MCP) penalized likelihood function[7, 8, 11]. However, even if these new procedures provide simple and efficient computational algorithms, but they inherit the drawbacks of LARS in high dimension with correlated predictors.

3. Mixture of Local Linear and Quadratic Approximations

In this Section, we propose our MLLQA procedure and establish its convergence property.

3.1. MLLQA procedure

To overcome the drawbacks of the LLA algorithm in high dimensional setting with correlated variables, we propose a new unified algorithm which is a mixture of local linear and quadratic approximations. Indeed, from the approximation (4), we obtain that $J_\lambda^{''}(|\beta_j|) \approx J_\lambda^{'}(|\beta_j^{(0)}|)/|\beta_j^{(0)}|$ for $\beta_j \approx \beta_j^{(0)}$. Then, we consider the following local quadratic approximation of the penalty function

$$J_\lambda(|\beta_j|) \approx J_\lambda(|\beta_j^{(0)}|) + J_\lambda^{'}(|\beta_j^{(0)}|)(|\beta_j| - |\beta_j^{(0)}|) + \frac{J_\lambda^{'}(|\beta_j^{(0)}|)}{2|\beta_j^{(0)}|}(|\beta_j|^2 - |\beta_j^{(0)}|^2)$$

for $\beta_j \approx \beta_j^{(0)}$. Finally, the iterative mixture of local linear and quadratic approximations (MLLQA) procedure is defined by

$$\boldsymbol{\beta}^{(k+1)} = \text{argmax}_{\boldsymbol{\beta}} \left\{ \sum_{i=1}^{n} \ell_i(\boldsymbol{\beta}) - n \sum_{j=1}^{p} J_\lambda^{'}(|\beta_j^{(k)}|)|\beta_j| - \frac{n}{2} \sum_{j=1}^{p} \frac{J_\lambda^{'}(|\beta_j^{(k)}|) + \tau_0}{|\beta_j^{(k)}| + \tau_0} |\beta_j|^2 \right\}. \tag{9}$$

The small perturbation τ_0 is introduced, in the numerator and denominator of the 3th term of (9), to handle the non-differentiability at 0 and in order to ensure convergence of our algorithm as we will see further, respectively. Consequently, the penalty in (9) is a combination of the weighted ℓ_1 and ℓ_2 norms. So, MLLQA is similar to the Elastic Net[14] which is more adapted to strong correlated variables in high dimensional linear regression setting. Thus, MLLQA inherits the good properties of LLA algorithm and corrects its difficulties in high dimension.

3.2. Convergence property of MLLQA algorithm

Following Schifano et al.[11], we assume that $J_\lambda^{'}(0+) \in [C_\lambda^{-1}, C_\lambda]$ for some finite $C_\lambda > 0$. So, $J_\lambda(.)$ satisfies the condition $(P1)$ in Schifano et al.[11], which implies that $J_\lambda^{'}(r) > 0$ for $r \in (0, K_\lambda)$, where $K_\lambda > 0$ may be finite or infinite. The positivity of the

right derivative at zero ensures that $\sum_{j=1}^{p} J_\lambda(|\beta_j|)$ is not identically zero for $|\beta_j| > 0$. Denote

$$H(\boldsymbol{\beta}|\boldsymbol{\beta}^{(k)}) = \left\{ \sum_{i=1}^{n} \ell_i(\boldsymbol{\beta}) - n \sum_{j=1}^{p} \Upsilon_{\tau_0}(\beta_j|\beta_j^{(k)}) \right\} \tag{10}$$

where

$$\Upsilon_{\tau_0}(\beta_j|\beta_j^{(k)}) = J_\lambda(|\beta_j^{(k)}|) + J_\lambda^{'}(|\beta_j^{(k)}|)(|\beta_j| - |\beta_j^{(k)}|) + \frac{J_\lambda^{'}(|\beta_j^{(k)}|) + \tau_0}{2(|\beta_j^{(k)}| + \tau_0)}(|\beta_j|^2 - |\beta_j^{(k)}|^2).$$

The following theorem states that MLLQA algorithm is an instance of *MM* algorithms and has the ascent property.

Theorem 3.1. *For a differentiable concave penalty function $J_\lambda(.)$ on $[0, \infty)$, we have*

$$P\ell(\boldsymbol{\beta}) \geq H(\boldsymbol{\beta}|\boldsymbol{\beta}^{(k)}) \quad \textit{and} \quad P\ell(\boldsymbol{\beta}^{(k)}) = H(\boldsymbol{\beta}^{(k)}|\boldsymbol{\beta}^{(k)}). \tag{11}$$

Furthermore, the MLLQA has the ascent property, i.e, for all k=0,1,2,...

$$P\ell(\boldsymbol{\beta}^{(k+1)}) \geq P\ell(\boldsymbol{\beta}^{(k)}). \tag{12}$$

Proof of Theorem 3.1. We recall that

$$P\ell(\boldsymbol{\beta}) - H(\beta|\beta^{(k)}) = n\{\sum_{i=1}^{n}(J_\lambda(|\beta_j^{(k)}|) + J_\lambda^{'}(|\beta_j^{(k)}|)(|\beta_j| - |\beta_j^{(k)}|) + \frac{J_\lambda^{'}(|\beta_j^{(k)}|) + \tau_0}{2(|\beta_j^{(k)}| + \tau_0)}(|\beta_j|^2 - |\beta_j^{(k)}|^2) - J_\lambda(|\beta_j|))\}.$$

By the concavity of $J_\lambda(.)$, we have

$$J_\lambda(|\beta_j^{(k)}|) + J_\lambda^{'}(|\beta_j^{(k)}|)(|\beta_j| - |\beta_j^{(k)}|) - J_\lambda(|\beta_j|) \geq 0 \quad \text{for} \quad j = 1, ..., p.$$

When $\beta_j^{(k)} = 0$, we use the right derivative. The quadratic term

$$\frac{J_\lambda^{'}(|\beta_j^{(k)}|) + \tau_0}{2(|\beta_j^{(k)}| + \tau_0)}(|\beta_j|^2 - |\beta_j^{(k)}|^2)$$

is always positive due to the approximation $(|\beta_j| - |\beta_j^{(k)}|)^2 \approx |\beta_j|^2 - |\beta_j^{(k)}|^2$ for $\beta_j \approx \beta_j^{(k)}$. We hence have $P\ell(\boldsymbol{\beta}) \geq H(\boldsymbol{\beta}|\boldsymbol{\beta}^{(k)})$ and it's easy to verify that $P\ell(\boldsymbol{\beta^{(k)}}) = H(\boldsymbol{\beta}^{(k)}|\boldsymbol{\beta}^{(k)})$.
The second inequality holds by the fact that $\boldsymbol{\beta}^{(k+1)} = \text{argmax}_{\boldsymbol{\beta}} H(\boldsymbol{\beta}|\boldsymbol{\beta}^{(k)})$, which leads to $P\ell(\boldsymbol{\beta}^{(k+1)}) \geq H(\boldsymbol{\beta}^{(k+1)}|\boldsymbol{\beta}^{(k)}) \geq H(\boldsymbol{\beta}^{(k)}|\boldsymbol{\beta}^{(k)}) = P\ell(\boldsymbol{\beta}^{(k)})$. ■
Let $M(\boldsymbol{\beta}^{(k)})$ denote the map defined by the MLLQA algorithm from $\boldsymbol{\beta}^{(k)}$ to $\boldsymbol{\beta}^{(k+1)}$. Note that the penalty function has continuous first derivative and solving $\boldsymbol{\beta}^{(k+1)}$ is a convex optimization problem, so *M* is a continuous map. We assume that the set $\mathfrak{S}$ of stationary points for $\xi(\boldsymbol{\beta}) = -P\ell(\boldsymbol{\beta})$ is both non empty and finite.

Theorem 3.2. *Let $\boldsymbol{\beta}^{(k+1)} = argmin_{\boldsymbol{\beta}} - H(\boldsymbol{\beta}|\boldsymbol{\beta}^{(k)})$. Then, using the condition (iii) of Theorem 2.1 in Schifano et al.[11], $\boldsymbol{\beta}^{(k+1)}$ converges to a stationary point of $\xi(\boldsymbol{\beta}) = -P\ell(\boldsymbol{\beta})$.*

Proof of Theorem 3.2. We recall that

$$-P\ell(\boldsymbol{\beta}) = -\ell(\boldsymbol{\beta}) + n\sum_{j=1}^{p} J_\lambda(|\beta_j|).$$

Let $\xi(\boldsymbol{\beta}) = -P\ell(\boldsymbol{\beta}) = g(\boldsymbol{\beta}) + nS_\lambda(\boldsymbol{\beta})$. The negative log-likelihood function $g(.)$ is strictly convex and $S_\lambda(\boldsymbol{\beta}) = \sum_{j=1}^{p} J_\lambda(|\beta_j|)$ satisfies the assumptions needed for Theorem 2.1 of Schifano et al.[11]. On the other hand, let
$U_\lambda(\boldsymbol{\beta}|\boldsymbol{\beta}^{(k)}) = \sum_{j=1}^{p} \tilde{u}_\lambda(|\beta_j|, |\beta_j^{(k)}|)$, where

$$\tilde{u}_\lambda(r,s) = J_\lambda(s) + J_\lambda^{'}(s)(r-s) + \frac{J_\lambda^{'}(s)+\tau_0}{2(s+\tau_0)}(r-s)^2$$

for $r \approx s$, with r and s are taken in a compact set of $(0,\infty)$. Then, we have $U_\lambda(\boldsymbol{\beta}|\boldsymbol{\beta}^{(k)}) - S_\lambda(\boldsymbol{\beta}^{(k)}) > 0$. This strict inequality is obtained by the concavity of $J_\lambda(.)$ on $(0,\infty)$ which leads to $J_\lambda(r) \leq J_\lambda(s) + J_\lambda^{'}(s)(r-s)$ for each $r,s > 0$ and the fact that $(J_\lambda^{'}(s)+\tau_0)(r-s)^2/2(s+\tau_0) > 0$ for each $r \approx s$. Hence, $-H(\boldsymbol{\beta}|\boldsymbol{\beta}^{(k)})$ strictly locally majorizes $\xi(\boldsymbol{\beta})$ in an open neighborhood containing $\boldsymbol{\beta}^{(k+1)}$. We mention here that as far as strictly local majorization holds at each iteration, we don't need to use the function $h(\boldsymbol{\beta},\alpha)$ used in Theorem 2.1 of Schifano et al.[11] to majorize $g(\boldsymbol{\beta})$. In fact, one can consider that, $\sum_{j=1}^{p} \frac{J_\lambda^{'}(|\beta_j^{(k)}|)+\tau_0}{2(|\beta_j^{(k)}|+\tau_0)}(|\beta_j| - |\beta_j^{(k)}|)^2 > 0$ instead of $h(\boldsymbol{\beta},\boldsymbol{\beta}^{(k)})$. This is the reason of introducing the perturbation to the numerator, as previously mentionned. Finally, strict convexity of $-H(\boldsymbol{\beta}|\boldsymbol{\beta}^{(k)})$ in $\boldsymbol{\beta}$ leads to unique minimum $\beta^{(k+1)}$. With locally strict majorization, we conclude that the MM algorithm derived from $-H(\boldsymbol{\beta}|\boldsymbol{\beta}^{(k)})$ converges to a stationary point of $\xi(\boldsymbol{\beta})$. ■

4. Statistical study of one-step MLLQA estimator

In this Section, we establish the oracle property of the one-step MLLQA estimator in the case of linear regression models based on the penalized least squares and in the most general penalized likelihood setting.

4.1. Linear regression case

In the case of linear models, the one step MLLQA estimator $\hat{\boldsymbol{\beta}}$ verifies

$$\hat{\boldsymbol{\beta}} = \text{argmin}_{\boldsymbol{\beta}} \left\{ \frac{1}{2} \parallel \mathbf{y} - X\boldsymbol{\beta} \parallel^2 + n\sum_{j=1}^{p} J_\lambda^{'}(|\beta_j^{(0)}|)|\beta_j| + \frac{n}{2}\sum_{j=1}^{p} \frac{J_\lambda^{'}(|\beta_j^{(0)}|)+\tau_0}{|\beta_j^{(0)}|+\tau_0}|\beta_j|^2 \right\}. \tag{13}$$

We remark that solving (13) is similar to elastic net problem. In fact unlike the elastic net based on the choice of two regularization parameters, here we deal with a single parameter

λ due to the behavior of the SCAD (and also MCP) penalty. It is easy to see that solving problem (13) is equivalent to find

$$\hat{\boldsymbol{\beta}} = \text{argmin}_{\boldsymbol{\beta}} \left\{ \frac{1}{2} \parallel \mathbf{y}^* - X^* \boldsymbol{\beta} \parallel^2 + n \sum_{j=1}^{p} J_\lambda^{'}(|\beta_j^{(0)}|)|\beta_j|, \right\} \tag{14}$$

where

$$\mathbf{y}^* = \begin{pmatrix} \mathbf{y} \\ \mathbf{0} \end{pmatrix}, X^* = \begin{pmatrix} X \\ S \end{pmatrix}$$

and S is a diagonal matrix with $S_{jj} = \sqrt{n \frac{J_\lambda^{'}(|\beta_j^{(0)}|)+\tau_0}{|\beta_j^{(0)}|+\tau_0}}$, $j = 1, ...p$.

Thus, maximizing $P\ell(\boldsymbol{\beta})$, as defined in (2) via the one-step MLLQA algorithm, is equivalent to use one-step LLA on an augmented data.
It's now interesting to see if $\hat{\boldsymbol{\beta}}$ enjoys oracle properties. So, we assume the two following regularity conditions (A.1) and (A.2) used in [15].

(A.1). $y_i = \mathbf{x}_i^t \boldsymbol{\beta}_0 + \epsilon_i$, where $\epsilon_1, ..., \epsilon_n$ are independent and identically distributed random variables with mean 0 and variance σ^2,

(A.2).

$$\frac{1}{n} X^t X \to C = \begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix}$$

where C is a positive definite matrix, $\boldsymbol{\beta}_0 = (\beta_{01}, ..., \beta_{0p})^t = (\boldsymbol{\beta}_{10}^t, \boldsymbol{\beta}_{20}^t)^t$ and $\boldsymbol{\beta}_{20} = 0$.

Theorem 4.1. *Assume that the previous assumptions (A.1) and (A.2) are satisfied and that if $\lambda_n \to 0$, $\sqrt{n}\lambda_n \to \infty$ and $\sqrt{n}\tau_0 \to 0$ as $n \to \infty$, then with probability tending to one we obtain that:*

(a) Sparsity: $\hat{\boldsymbol{\beta}}_2 = 0$.

(b) Asymptotic normality: $\sqrt{n}(\hat{\boldsymbol{\beta}}_1 - \boldsymbol{\beta}_{10}) \to \text{N}(0, \sigma^2 C_{11}^{-1})$.

We omit the proof of Theorem 4.1 since it is similar to that of Theorem 4.2 defined in Section 4.2.

4.2. Generalized linear model case

In the penalized likelihood setting, we assume that the log-likelihood function $\ell(\boldsymbol{\beta}) = \sum_{i=1}^{n} \ell_i(\boldsymbol{\beta})$ is twice differentiable according to $\boldsymbol{\beta}$. For a given initial value $\boldsymbol{\beta}^{(0)}$, we can use the following local approximation:

$$\ell(\boldsymbol{\beta}) \approx \ell(\boldsymbol{\beta}^{(0)}) + \nabla \ell(\boldsymbol{\beta}^{(0)})^t (\boldsymbol{\beta} - \boldsymbol{\beta}^{(0)}) + \frac{1}{2} (\boldsymbol{\beta} - \boldsymbol{\beta}^{(0)})^t \nabla^2 \ell(\boldsymbol{\beta}^{(0)}) (\boldsymbol{\beta} - \boldsymbol{\beta}^{(0)}). \tag{15}$$

Starting from $\boldsymbol{\beta}^{(0)} = \hat{\boldsymbol{\beta}}$(mle) the maximum likelihood estimator, with $\nabla \ell(\boldsymbol{\beta}^{(0)}) = 0$, then $\hat{\boldsymbol{\beta}}$ verifies

$$\begin{aligned} \hat{\boldsymbol{\beta}} = \text{argmin}_{\boldsymbol{\beta}} \{ & \frac{1}{2} (\boldsymbol{\beta} - \boldsymbol{\beta}^{(0)})^t [-\nabla^2 \ell(\boldsymbol{\beta}^{(0)})] (\boldsymbol{\beta} - \boldsymbol{\beta}^{(0)}) \\ & + n \sum_{j=1}^{p} J_\lambda^{'}(|\beta_j^{(0)}|)|\beta_j| + \frac{n}{2} \sum_{j=1}^{p} \frac{J_\lambda^{'}(|\beta_j^{(0)}|) + \tau_0}{|\beta_j^{(0)}| + \tau_0} |\beta_j|^2 \}, \end{aligned} \tag{16}$$

which can be written as

$$\hat{\boldsymbol{\beta}} = \text{argmin}_{\boldsymbol{\beta}} \left\{ \frac{1}{2}(\boldsymbol{\beta} - \boldsymbol{\beta}^{(0)})^t[-\nabla^2\ell(\boldsymbol{\beta}^{(0)})](\boldsymbol{\beta} - \boldsymbol{\beta}^{(0)}) \right. \\ \left. + \frac{n}{2}\boldsymbol{\beta}^t\mathbf{Q}_{\tau_0}\boldsymbol{\beta} + n\sum_{j=1}^{p} J'_\lambda(|\beta_j^{(0)}|)|\beta_j| \right\} \tag{17}$$

where $\mathbf{Q}_{\tau_0} = \text{diag}(\mathbf{Q}_{\tau_0 11}, \ldots, \mathbf{Q}_{\tau_0 pp})$ and $\mathbf{Q}_{\tau_0 jj} = \frac{J'_\lambda(|\beta_j^{(0)}|)+\tau_0}{|\beta_j^{(0)}|+\tau_0}$.

We denote $I(\boldsymbol{\beta}_0)$ the $p \times p$ Fisher information matrix and the submatrix $I_1(\boldsymbol{\beta}_{10}) = I(\boldsymbol{\beta}_{10}, 0)$, the Fisher information knowing $\boldsymbol{\beta}_{20} = 0$. As advocated in [9], under some regularity conditions, we have $n^{-1}\nabla^2\ell(\hat{\boldsymbol{\beta}}(mle)) \to_P -I(\boldsymbol{\beta}_0)$, and $\sqrt{n}(\boldsymbol{\beta}_0 - \hat{\boldsymbol{\beta}}(mle)) \to_D W = N(0, I^{-1}(\boldsymbol{\beta}_0))$. The following theorem assesses the oracle property of the one step MLLQA estimator for penalized likelihood.

Theorem 4.2. *Under the previous assumptions, if* $\lambda_n \to 0$, $\sqrt{n}\lambda_n \to \infty$ *and* $\sqrt{n}\tau_0 \to 0$ *as* $n \to \infty$*, then with probability tending to one,* $\hat{\boldsymbol{\beta}}$ *satisfies:*

(a) Sparsity: $\hat{\boldsymbol{\beta}}_2 = 0$.

(b) Asymptotic normality: $\sqrt{n}(\hat{\boldsymbol{\beta}}_1 - \boldsymbol{\beta}_{10}) \to \text{N}(0, I_1^{-1}(\boldsymbol{\beta}_{10}))$

According to the two previous theorems, we see that oracle properties require for the penalty function to be twice differentiable, λ_n is chosen as in Theorem 2 of [3] and we have used a supplementary condition $\sqrt{n}\tau_0 \to 0$ as $n \to \infty$. This is justified by the fact that our algorithm is based on linear and quadratic approximation of the SCAD penalty function, which uses the second order derivatives. We recall that results for the one step LLA require less regularity conditions than results given in Fan and Li[3].

REMARK. — In their earlier work, Fan and Li[3] showed that continuity for the nonconcave penalized likelihood estimates is guaranteed by the condition that the minimum of the function $|\theta| + J'_\lambda(|\theta|)$ must be attained at 0. Since our MLLQA one step estimator is based on a mixture of linear and quadratic approximation of the penalty function, continuity of $\hat{\beta}$ only requires that $J'_\lambda(|\theta|)$ is continuous for $|\theta| > 0$, as with the one step LLA estimator[15]. Since the computation of the sparse one-step MLLQA estimator is based on the LLA algorithm which uses an L_1 penalized criterion. The quadratic term of the approximation only contributes in the non penalized part of the objective function.

Proof of Theorem 4.2. We only demonstrate oracle properties for the penalized likelihood estimates. The proof for linear regression model is similar. The following proof is based on a slightly modified version of Theorem 5 in [15]. Let us define

$$\begin{aligned} K_n(u) &= \frac{1}{2}(\frac{u}{\sqrt{n}} + \boldsymbol{\beta}_0 - \boldsymbol{\beta}^{(0)})^t[-\nabla^2\ell(\boldsymbol{\beta}^{(0)})](\frac{u}{\sqrt{n}} + \boldsymbol{\beta}_0 - \boldsymbol{\beta}^{(0)}) + \\ &\quad n\sum_{j=1}^{p} J'_{\lambda_n}(|\beta_j^{(0)}|)|\beta_{0j} + \frac{u_j}{\sqrt{n}}| + \frac{n}{2}\sum_{j=1}^{p}\frac{J'_{\lambda_n}(|\beta_j^{(0)}|) + \tau_0}{|\beta_j^{(0)}| + \tau_0}(\beta_{0j} + \frac{u_j}{\sqrt{n}})^2. \end{aligned}$$

Then, we have

$$\begin{aligned} K_n(u) - K_n(0) &= \frac{1}{2}(\frac{u^t}{\sqrt{n}}[-\nabla^2\ell(\boldsymbol{\beta}^{(0)})]\frac{u}{\sqrt{n}} + (\boldsymbol{\beta}_0 - \boldsymbol{\beta}^{(0)})^t[-\nabla^2\ell(\boldsymbol{\beta}^{(0)})]\frac{u}{\sqrt{n}} \\ &+ n\sum_{j=1}^{p} J'_{\lambda_n}(|\beta_j^{(0)}|)(|\beta_{0j} + \frac{u_j}{\sqrt{n}}| - |\beta_{0j}|) \\ &+ \frac{n}{2}\sum_{j=1}^{p}\frac{J'_{\lambda_n}(|\beta_j^{(0)}|) + \tau_0}{|\beta_j^{(0)}| + \tau_0}((\beta_{0j} + \frac{u_j}{\sqrt{n}})^2 - \beta_{0j}^2) \\ &\equiv T_1 + T_2 + T_3 + T_4. \end{aligned}$$

Moreover, it's easy to see that

$$\hat{u}(n) = \text{argmin}_u[K_n(u) - K_n(0)]$$

leads to $\hat{\boldsymbol{\beta}} = \boldsymbol{\beta}_0 + \frac{\hat{u}(n)}{\sqrt{n}}$ with $\hat{\boldsymbol{\beta}}$ the one step MLLQA estimator.

By Slutsky's theorem and using the same argument as in the proof of Theorem 5 in Zou and Li[15], it follows that

$$T_1 = \frac{1}{2}(\frac{u^t}{\sqrt{n}}[-\nabla^2\ell(\boldsymbol{\beta}^{(0)})]\frac{u}{\sqrt{n}} \to_P \frac{1}{2}u^t I(\boldsymbol{\beta}_0)u,$$

$$T_2 = (\boldsymbol{\beta}_0 - \boldsymbol{\beta}^{(0)})^t[-\nabla^2\ell(\boldsymbol{\beta}^{(0)})]\frac{u}{\sqrt{n}} = \sqrt{n}(\boldsymbol{\beta}_0 - \boldsymbol{\beta}^{(0)})^t[\frac{-\nabla^2\ell(\boldsymbol{\beta}^{(0)})}{n}]u \to_D -W^t I(\boldsymbol{\beta}_0)u$$

and

$$T_3 \to_P \begin{cases} 0 & \text{if } u_{20} = 0 \\ \infty & \text{otherwise.} \end{cases} \tag{18}$$

The last term can be written as

$$T_4 = \frac{1}{2}\sum_{j=1}^{p}\sqrt{n}\frac{J'_{\lambda_n}(|\beta_j^{(0)}|) + \tau_0}{|\beta_j^{(0)}| + \tau_0}(\frac{(\beta_{0j} + \frac{u_j}{\sqrt{n}})^2 - \beta_{0j}^2}{\frac{1}{\sqrt{n}}}) = \frac{1}{2}\sum_{j=1}^{p} T_{4j}.$$

First, it can be seen that

$$\frac{(\beta_{0j} + \frac{u_j}{\sqrt{n}})^2 - \beta_{0j}^2}{\frac{1}{\sqrt{n}}} \to 2u_j\beta_{0j}I(\beta_{0j} \neq 0) + \frac{u_j^2}{\sqrt{n}}I(\beta_{0j} = 0).$$

We now examine the behavior of $\sqrt{n}(J'_{\lambda_n}(|\beta_j^{(0)}|) + \tau_0)/(|\beta_j^{(0)}| + \tau_0)$.

When $\beta_{0j} \neq 0$, we have $|\beta_j^{(0)}| \to_P |\beta_{0j}| > 0$ and $|\beta_j^{(0)}| + \tau_0$ remains bounded away from zero. Moreover, using the fact that $J'_{\lambda_n}(\theta) = 0$ if $\theta > a\lambda_n$ and if $\sqrt{n}\tau_0 \to 0$ and $\lambda_n \to 0$ as $n \to \infty$ we conclude that $\sqrt{n}\frac{J'_{\lambda_n}(|\beta_j^{(0)}|) + \tau_0}{|\beta_j^{(0)}| + \tau_0}u_j\beta_{0j} \to_P 0$.

When $\beta_{0j} = 0$, $T_{4j} = 0$ if $u_j = 0$ else we have $|\beta_j^{(0)}| = O_P(1/\sqrt{n})$. On the other hand, $J'_{\lambda_n}(\theta) = \lambda_n$ for all $0 < \theta < \lambda_n$ and $\sqrt{n}\lambda_n \to \infty$ and $\sqrt{n}\tau_0 \to 0$ as $n \to \infty$, then

$$T_{4j} = (\sqrt{n}\lambda_n + \sqrt{n}\tau_0)\frac{u_j^2}{\sqrt{n}(|\beta_j^{(0)}| + \tau_0)} \to_P \infty.$$

So

$$T_4 \to_P \begin{cases} 0 & \text{if } u_{20} = 0 \\ \infty & \text{otherwise.} \end{cases} \tag{19}$$

By taking $W = (W_{10}^t, W_{20}^t)$, from T_1, T_2, T_3 and T_4 convergence results we conclude that for each fixed u,

$$K_n(u) - K_n(0) \to_d K(u) \equiv \begin{cases} \frac{1}{2} u_{10}^t I_1(\boldsymbol{\beta}_{10}) u_{10} - W_{10}^t u_{10} & \text{if} \quad u_{20} = 0 \\ \infty & \text{otherwise.} \end{cases} \tag{20}$$

. The unique minimum of $K(u)$ is $u = (u_{10} = I_1^{-1}(\boldsymbol{\beta}_{10}) W_{10}, u_{20} = 0)$. Since $K_n(u) - K_n(0)$ is a convex function of u, we conclude by epiconvergence as in [15], that

$$\hat{u}(n)_{10} \to_d I_1^{-1}(\boldsymbol{\beta}_{10}) W_{10} \tag{21}$$

$$\hat{u}(n)_{20} \to_d 0. \tag{22}$$

Considering $W_{10} = N(0, I_1(\boldsymbol{\beta}_{10}))$, (21) is equivalent to $\sqrt{n}(\hat{\boldsymbol{\beta}}_1 - \boldsymbol{\beta}_{10}) \to N(0, I_1^{-1}(\boldsymbol{\beta}_{10}))$ and (22) implies that $\sqrt{n}\hat{\boldsymbol{\beta}}_2 \to_P 0$.

Now we have to show that $P(\hat{\boldsymbol{\beta}}_2 = 0) \to 1$, which is stronger statement than (22). We just have to show that if $\beta_{0j} = 0$, then $P(\hat{\beta}_j \neq 0) \to 0$. Assume $\hat{\beta} j \neq 0$, by (KKT) conditions of (16), we must have

$$\frac{1}{\sqrt{n}}([-\nabla^2 \ell(\boldsymbol{\beta}^{(0)})](\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}^{(0)}))_j = \sqrt{n}\lambda(J'_{\lambda_n}(|\beta_j^{(0)}|) + \frac{J'_{\lambda_n}(|\beta_j^{(0)}|) + \tau_0}{|\beta_j^{(0)}| + \tau_0}|\hat{\beta}_j|). \tag{23}$$

Since $\frac{J'_{\lambda_n}(|\beta_j^{(0)}|)+\tau_0}{|\beta_j^{(0)}|+\tau_0}|\hat{\beta}_j| \geq 0$, (23) leads to

$$\frac{1}{\sqrt{n}}([-\nabla^2 \ell(\boldsymbol{\beta}^{(0)})](\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}^{(0)}))_j \geq \sqrt{n}\lambda J'_{\lambda_n}(|\beta_j^{(0)}|).$$

When $\beta_{0j} = 0$, $\lambda\sqrt{n}J'_{\lambda_n}(|\beta_j^{(0)}|)$ goes to ∞ in probability. Moreover, the left hand side of (23) can be written as $([\frac{-\nabla^2 \ell(\boldsymbol{\beta}^{(0)})}{n}]\sqrt{n}(\hat{\boldsymbol{\beta}} - \boldsymbol{\beta}_0))_j - ([\frac{-\nabla^2 \ell(\boldsymbol{\beta}^{(0)})}{n}]\sqrt{n}(\boldsymbol{\beta}^{(0)} - \boldsymbol{\beta}_0))_j$. From (21) and (22), the first term converges in law to some normal, and so does the second term. Thus

$$P(\hat{\beta}_j \neq 0) \leq P(\text{KKT condition} \quad (23) \quad \text{holds}) \to 0.\blacksquare$$

5. Numerical experiments

In this section, we study the performances of our one step MLLQA and its competitors on simulated and real data sets. As competitors we consider one step LLA, LQA, Perturbed LQA (PLQA), LASSO and ENET. For the choice of the initial parameter vector $\boldsymbol{\beta}^{(0)}$, we use the Maximum Likelihood Estimator (MLE) in the classical case when $n > p$, otherwise we consider the ℓ_2-Penalized MLE. In all of the experiments, computations are conducted using R software.

5.1. Simulation study

In all of the examples the correlation matrix (Σ) is defined by $\Sigma_{ij} = \rho^{|i-j|}$, $1 \leq i, j \leq n$ and $\rho \in \{0, 0.5, 0.75, 0.9\}$. For illustration, we consider the setting of linear regression and logistic regression models. In all of the simulated examples, the perturbation $\tau_0 = 10^{-6}$ for one step MLLQA, and the same value is considered for the PLQA with ϵ chosen according to equation (3.12) in Hunter and Li[6]. The tuning parameters are selected by ten-fold cross validation. The statistics considered here are the prediction error (MSE_y), the false positive (FP), which is the number of true zero coefficients incorrectly estimated as nonzero, and false negative (FN), which is the number of true nonzero coefficients incorrectly estimated to zero value. All simulations are performed 100 times and standard error related to an estimation, presented in brackets, is obtained by 500 bootstrap resampling.

5.1.1. Linear regression case

Example 1: ($n > p$). We consider the following model $y_i = \boldsymbol{x}_i^t \beta + \epsilon_i$ where $\boldsymbol{x}_i = (x_{i1}, ..., x_{i8})$ is a multinormal vector with correlation matrix (Σ), $\boldsymbol{\beta} = (3, 1.5, 0, 0, 2, 0, 0, 0)$ and $\epsilon_i \sim N(0, 1)$ $1 \leq i \leq n$. The sample size n is set to be $n = 60$ for training sample and the test sample has size $ntest = 2 \times n$. The initial vector $\boldsymbol{\beta}^{(0)}$ corresponds to the Ordinary Least Square estimate. Table1 summarizes the results for $\rho \in \{0, 0.5\}$.

" Table 1 about here "

As presented in Table 1, one step MLLQA performs slightly better in terms of MSE_y and FP followed by the one step LLA, where the difference in terms of FP rate with the four other methods is clear. While the methods are all comparable in terms of FN. In addition, the performance of LASSO and ENET are relatively similar but slightly better than the performance of LQA and PLQA.

Example 2: $p > n$. We consider the high dimensional model $y = \boldsymbol{x}^t \beta + \epsilon$ where ϵ follows a standard gaussian distribution and the predictors are generated as in the previous example. The number of predictors is fixed to $p = 120$ and the vector $\boldsymbol{\beta}$ has nine nonzero components of differents signs and the remaining components set to be equal to zero, so $\boldsymbol{\beta} = (\underbrace{3, 3, -1/3, ..., 3, 3, -1/3}_{9}, \underbrace{0, ..., 0}_{111})$. Sample sizes considered for both training and test sets are $n = p/3 = 40$ and $n = p/2 = 60$. The results are summarized in Table 2 and Table 3 for $\rho \in \{0.5, 0.75, 0.9\}$. For this example, we use a ridge regression estimate for $\beta^{(0)}$ as initial value instead of the classical OLS estimate.

As can be seen from Table 2 for $\rho = 0.5$, ENET and LASSO performs slightly better than MLLQA in terms of MSE_y, respectively. Moreover, MLLQA does better in terms of FP and FN rates with large difference in terms of FP. However, MLLQA performs better than all other methods in terms of MSE_y and FP for $\rho \in \{0.75, 0.9\}$. It is followed by LLA in terms of FP and ENET in terms of MSE_y. Unlike Example 1, the difference between MLLQA and LLA appears to be slightly greater in terms of errors and it is relatively low in terms of FP. In terms of FN, PLQA and LQA perform surprisingly better than all other methods, but their performance is bad in all other three measures.

Table 3 displays the results for the case where the sample size $n = 60$ and $p = 120$. In all terms and for all ρ values, it can be seen that the one step MLLQA is better than its competitors, except in terms of FN rate where PLQA and LQA are slightly better than MLLQA. Furthermore, we note that when the sample size increases the gap between LLA and MLLQA is increased in terms of prediction and estimation errors particularly

for $\rho = 0.9$. Globally, it can be seen that the FN rates of all methods decrease when the sample size increases.

Example 3: $p > n$

We consider the high dimensional model $y = \boldsymbol{x}^t\beta + \sigma \times \epsilon$ where ϵ follows a standard gaussian distribution and the predictors are generated as in the previous examples with correlation matrix (Σ) defined by $\Sigma_{ij} = 0.9^{|i-j|}$, $1 \leq i, j \leq n$. The number of predictors is fixed to $p = 120$ with the nine first components of β with nonzero values and the remaining components set to be equal to zero, so that $\beta = (\underbrace{10, ..., 10}_{9}, \underbrace{0, ..., 0}_{111})$. Sample size considered is $n = p/2 = 60$ for both training and test sets and $\sigma \in \{3, 5\}$.

Results in Table 4 show that one step MLLQA is also the winner in terms of FP for $\sigma \in \{3, 5\}$ followed by one step LLA. Considering MSE_y, ENET performs better for $\sigma = 3$ followed respectively by one step MLLQA and LASSO; for $\sigma = 5$ our one step MLLQA is the winner followed by ENET and LASSO. Globally for this example, one step LLA, LQA and PLQA seem not having good performances except for FN where PLQA and LQA perform better.

" Table 2 about here "

" Table 3 about here "

" Table 4 about here "

5.1.2. Logistic regression

The response variable is generated from a binomial distribution with parameter

$$\Pi = P(y = 1|X = \mathbf{x}) = e^{\mathbf{x}^t\beta}/(1 + e^{\mathbf{x}^t\beta}).$$

The predictors $\mathbf{x}$ are generated following the similar model as in examples before. On the other hand, the sample size considered is $n = 200$ for training and test sets. The parameter vector considered is $\beta = (3, 1.5, 0, 0, 2, 0, 0, -1)$. The misclassification error rate, FP and FN rates are the three measures used for comparing the performance of different methods on test data set. In this example, best results are obtained for weighted methods (LLA, LQA, PLQA, MLLQA) by using $\exp(\hat{\lambda})$ rather than $\hat{\lambda}$ when evaluating weights related to each coefficient, where $\hat{\lambda}$ is the optimal tuning parameter selected by tenfold cross validation.

For this example, the results in Table 5 show that the one step MLLQA is the winner in terms of misclassification error rate followed by the LASSO, while one step LLA is the winner in terms of false positive. Moreover, LASSO, the one step MLLQA and ENET have a zero false negative rate, which is not the case for one step LLA, LQA and PLQA.

" Table 5 about here "

5.2. Real data experiments: p»n

We propose to test the performance of our method on ARCENE dataset which is one of the five NIPS 2003 feature selection challenge data sets. The data set is obtained from

UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. For this two-class classification problem with continuous input variables, the aim is to distinguish cancer versus normal patterns from mass-spectrometric data. The samples include patients with cancer (ovarian or prostate cancer), and healthy or control patients. ARCENE was obtained by merging three mass-spectrometry data sets to obtain enough training and test data for a benchmark. The original features indicate the abundance of proteins in human sera having a given mass value. Based on those features one must separate cancer patients from healthy patients. The sample size considered here is $n = 100$ for training and test sets with $p = 10000$ features. Before performing variables selection, we firstly select $p = 500, 1000, 1500$ and 2000 variables with the smallest p-value and compare our methods on the selected subset of variables. This pre-selection step is a common approach used in many papers with ultra high dimensional data sets.

According to results in Table 6, the one step MLLQA is very competitive in terms of misclassification error rate and has a tendency to select small number of variables in all settings (followed by LLA or LASSO). While, ENET is competitive in terms of misclassification error, but it has a tendency to introduce a lot of variables. The LQA and PLQA perform badly in terms of misclassification rate and select more variables than all other methods (except in $p = 500$).

" Table 6 about here "

6. Conclusion

We have proposed an efficient one-step sparse estimation procedure in nonconcave penalized likelihood models, which is based on the mixture of local linear and quadratic approximation penalties (MLLQA). The new iterative MLLQA enjoys the advantages of both LLA and the perturbed LQA algorithms. Its convergence property is shown. As with LLA, MLLQA does not delete any small coefficient and it produces a sparse estimates via continuous penalization. Computationally, we take advantage of the efficient coordinate descent algorithm for LASSO penalized regression to compute the one-step MLLQA estimator. Moreover, the oracle property of one-step MLLQA estimator is established. Empirically, the proposed method provides smaller models with better prediction accuracy in comparison with its principal competitors.

7. References

[1] CHEN, S., DONOHO, D., SAUNDERS, M., "Atomic decomposition by basis pursuit", *SIAM J. on Sci. Comp.*, vol. 20, num. 1, 1998.

[2] EFRON, B., HASTIE, T., JOHNSTONE, I., TIBSHIRANI, R. "Least angle regression", *Annals of Statistics*, num. 32, 2004.

[3] FAN, J., LI, R. "Variable selection via nonconcave penalized likelihood and its oracle properties", *Journal American Statistical Association*, num. 96, 2001.

[4] FRANK, I. E., FRIEDMAN, J. H., "A statistical view of some chemometrics regression tools (with discussion)", *Technometrics*, num. 35, 1993.

[5] HUNTER, D., LANGE, K., "A tutorial on MM algorithms", *American. Statistician*, num. 58, 2004.

[6] HUNTER, D., LI, R., "Variable selection using MM algorithm", *The Annals of Statistics*, num. 33, 2005.

[7] KIM, Y., CHOI, H., OH, H., "Smoothly clipped absolute deviation on high dimensions", *J. Amer. Statist. Ass.*, num. 103, 2008.

[8] KWON, S., CHOI, H., Kim, Y., "Quadratic approximation on SCAD penalized estimation", *Comp. Statist. and Data Anal.*, num. 55, 2011.

[9] LEHMANN, E., Casella, G., "Theory of point estimation, Second edition, Springer", 2011.

[10] RADCHENKO, P., JAMES, G. M., "Variable inclusion and shrinkage algorithms", *Journal of the Americain Statistical Association.*, vol. 103, num. 483, 2008.

[11] SCHIFANO, E. D., STRAWDERMA, R. L., Wells, M., T., "Majorization-Minimization algorithms for nonsmoothly penalized objective functions", *Electronic Journal of Statistics*, vol. 4, 2010.

[12] TIBSHIRANI, R., "Regression shrinkage and selection via the Lasso", *Journal of the Royal Statistical Society, B.*, num. 58, 1994.

[13] ZHANG, C.-H., "Nearly unbiased variable selection minimax concave penalty", *Annals of Statistics*, num. 38, 2010.

[14] ZOU, H., HASTIE, T., "Regularization and variable selection via the elastic-net", *Journal of the Royal statistical Society, B.*, num. 67, 2005.

[15] ZOU, H., LI, R., "One-step sparse estimates in nonconcave penalized likelihood models", *Annals of Statistics*, vol. 36, num. 4, 2008.

Method	MSE_y	Number of Zeros FP	FN
		$\rho = 0$	
LASSO	1.144(0.018)	2.50(0.17)	0(-)
LLA	1.139(0.018)	1.91(0.16)	0(-)
LQA	1.156(0.017)	4.26(0.09)	0(-)
PLQA	1.152(0.019)	4.24(0.09)	0(-)
MLLQA	**1.132**(0.018)	**1.77**(0.16)	0(-)
ENET	1.142(0.019)	2.49(0.15)	0(-)
		$\rho = 0.5$	
LASSO	1.119(0.015)	2.36(0.15)	0(-)
LLA	1.112(0.016)	1.67(0.15)	0(-)
LQA	1.118(0.015)	3.90(0.10)	0(-)
PLQA	1.114(0.016)	3.84(0.10)	0(-)
MLLQA	**1.108**(0.016)	**1.58**(0.15)	0(-)
ENET	1.119(0.015)	2.38(0.14)	0(-)

Table 1. *Simulation results for Example 1.*

Method	MSE_y	Number of Zeros FP	FN
		$\rho = 0.5$	
LASSO	2.35(0.10)	16.26(0.56)	2.62(0.06)
LLA	3.22(0.70)	7.18(0.60)	2.41(0.12)
LQA	22.02(2.92)	18.72(1.66)	1.68(0.18)
PLQA	20.96(2.95)	20.60(1.75)	**1.56**(0.16)
MLLQA	2.40(0.17)	**4.60**(0.57)	2.30(0.09)
ENET	**2.33**(0.08)	12.95(0.52)	2.54(0.06)
		$\rho = 0.75$	
LASSO	1.91(0.06)	13.78(0.47)	2.45(0.07)
LLA	2.51(0.48)	2.99(0.45)	2.19(0.12)
LQA	14.44(2.54)	19.94(1.52)	1.47(0.13)
PLQA	12.55(2.13)	21.03(1.87)	**1.26**(0.10)
MLLQA	**1.74**(0.06)	**1.33**(0.23)	1.85(0.11)
ENET	1.82(0.06)	9.20(0.57)	2.20(0.08)
		$\rho = 0.9$	
LASSO	1.71(0.05)	14.87(0.51)	2.15(0.07)
LLA	4.00(1.01)	1.38(0.24)	2.05(0.17)
LQA	5.68(0.59)	26.26(1.73)	1.29(0.09)
PLQA	5.05(0.57)	28.88(1.61)	**1.08**(0.09)
MLLQA	**1.47**(0.05)	**0.44**(0.10)	1.37(0.09)
ENET	1.62(0.05)	6.97(0.48)	1.46(0.08)

Table 2. *Simulation results for Example 2:* $n = 40$*,* $p = 3 * n$*.*

Method	MSE_y	Number of Zeros FP	FN
		$\rho = 0.5$	
LASSO	1.87(0.05)	23.17(0.81)	2.35(0.07)
LLA	3.19(0.80)	5.98(0.77)	2.32(0.14)
LQA	13.81(2.13)	21.07(1.89)	**1.17**(0.15)
PLQA	14.57(2.36)	18.12(1.62)	1.34(0.13)
MLLQA	**1.57**(0.05)	**3.76**(0.58)	2.08(0.11)
ENET	1.78(0.05)	14.40(0.70)	2.60(0.06)
		$\rho = 0.75$	
LASSO	1.72(0.04)	21.36(0.62)	2.47(0.06)
LLA	2.94(0.69)	3.21(0.54)	1.97(0.14)
LQA	5.87(0.74)	18.44(1.74)	**1.20**(0.10)
PLQA	6.12(0.93)	21.38(2.10)	1.25(0.10)
MLLQA	**1.34**(0.03)	**1.18**(0.24)	1.73(0.10)
ENET	1.57(0.04)	11.45(0.64)	2.31(0.07)
		$\rho = 0.9$	
LASSO	1.54(0.03)	21.22(0.53)	2.28(0.07)
LLA	2.49(0.45)	1.38(0.28)	1.61(0.14)
LQA	3.19(0.27)	26.93(2.12)	1.46(0.09)
PLQA	3.18(0.26)	27.69(2.06)	**1.44**(0.09)
MLLQA	**1.31**(0.04)	**0.46**(0.12)	1.52(0.10)
ENET	1.40(0.03)	12.71(0.37)	1.69(0.08)

Table 3. *Simulation results for Example 2:* $n = 60$, $p = 2 * n$.

Method	MSE_y	Number of Zeros FP	FN
		$\sigma = 3$	
LASSO	14.75(0.36)	25.90(0.48)	2.36(0.07)
LLA	18.19(4.02)	4.15(0.13)	3.05(0.06)
LQA	33.28(7.96)	47.39(1.61)	1.39(0.12)
PLQA	32.69(8.67)	47.43(1.80)	**1.39**(0.11)
MLLQA	14.15(3.17)	**3.57**(0.06)	3.04(0.04)
ENET	**12.14**(0.32)	12.03(0.73)	2.70(0.05)
		$\sigma = 5$	
LASSO	40.47(0.93)	24.07(0.45)	2.42(0.07)
LLA	49.33(5.70)	4.49(0.29)	3.14(0.09)
LQA	100.23(8.16)	33.98(1.86)	1.21(0.11)
PLQA	92.82(8.75)	35.50(1.77)	**1.18**(0.14)
MLLQA	**32.81**(0.67)	**3.59**(0.10)	2.98(0.02)
ENET	34.80(0.76)	11.02(0.93)	2.73(0.05)

Table 4. *Simulation results for Example 3.*

Method	Misclassification error	FP	FN
LASSO	24.14(0.53)	2.48(0.10)	0(-)
LLA	24.17(0.55)	**2.47**(0.11)	0.01(0.01)
LQA	27.12(0.65)	2.95(0.07)	0.02(0.01)
PLQA	27.12(0.68)	2.64(0.08)	0.02(0.01)
MLLQA	**23.90**(0.55)	2.75(0.10)	0(-)
ENET	24.26(0.54)	3.19(0.08)	0(-)

Table 5. *Simulation results for logistic regression model.*

Method	Misclassification error	♯ Selected variables
	$p = 500$	
LASSO	36	18
LLA	36	18
LQA	40	65
PLQA	38	123
MLLQA	**33**	**11**
ENET	37	68
	$p = 1000$	
LASSO	29	22
LLA	29	**20**
LQA	44	119
PLQA	44	119
MLLQA	**27**	21
ENET	28	108
	$p = 1500$	
LASSO	30	43
LLA	31	41
LQA	44	174
PLQA	44	174
MLLQA	29	**36**
ENET	**27**	114
	$p = 2000$	
LASSO	**30**	47
LLA	32	50
LQA	45	150
PLQA	44	244
MLLQA	31	**46**
ENET	32	122

Table 6. *Results for ARCENE dataset.*