HHS Public Access

Author manuscript

Technometrics. Author manuscript; available in PMC 2016 August 04.

Published in final edited form as:

Technometrics. 2016 ; 58(3): 285–293. doi:10.1080/00401706.2015.1054436.

Orthogonalizing EM: A design-based least squares algorithm

Shifeng Xiong[1], **Bin Dai**[2], **Jared Huling**[3], and **Peter Z. G. Qian**[3]

[1]Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100190

[2]Tower Research Capital, 377 Broadway, New York, NY 10013

[3]Department of Statistics, University of Wisconsin-Madison, Madison, WI 53706

Abstract

We introduce an efficient iterative algorithm, intended for various least squares problems, based on a design of experiments perspective. The algorithm, called orthogonalizing EM (OEM), works for ordinary least squares and can be easily extended to penalized least squares. The main idea of the procedure is to orthogonalize a design matrix by adding new rows and then solve the original problem by embedding the augmented design in a missing data framework. We establish several attractive theoretical properties concerning OEM. For the ordinary least squares with a singular regression matrix, an OEM sequence converges to the Moore-Penrose generalized inverse-based least squares estimator. For ordinary and penalized least squares with various penalties, it converges to a point having grouping coherence for fully aliased regression matrices. Convergence and the convergence rate of the algorithm are examined. Finally, we demonstrate that OEM is highly efficient for large-scale least squares and penalized least squares problems, and is considerably faster than competing methods when n is much larger than p. Supplementary materials for this article are available online.

Keywords

Design of experiments; Computational statistics; Missing data; Optimization; Orthogonal design; SCAD; MCP; The Lasso

1 INTRODUCTION

Observational data with massive sample size appear in an increasing multitude of fields. In a growing number of areas such as marketing, physics, computational biology, engineering, and web applications, data have tens of millions of observations or more. While much recent research has been devoted to high-dimensional data, in this paper we develop an experimental design method that targets big data. Our method is shown to have state-of-the-art performance for data with a large number of observations. We investigate the theoretical

*Corresponding author: Peter Z. G. Qian. peterq@stat.wisc.edu.

SUPPLEMENTARY MATERIALS

This article has supplementary materials made available online. It consists of the following files:

OEM SupplementaryMaterial: This is a pdf file providing proofs of theorems, extra examples, and further numerical results including plots and simulation results.

properties of our method and demonstrate its efficiency in the computation of standard statistical methods.

This project aims at developing a new data augmentation method for fitting ordinary and penalized least squares and in big data. Unlike approximate methods, OEM solves the least squares problem exactly. OEM starts with a trivial observation in fitting ordinary or penalized least squares with orthogonal matrices. If the design matrix is orthogonal, then the ordinary least squares problem has a *closed-form* solution without any matrix inversion. More importantly, this holds regardless of the sample size and in this sense, scales well for big data. But the hard truth is that most large observational data in practice do not have orthogonal design matrices. In this paper, we propose an experimental design method to circumvent this difficulty.

Consider a regression model

$$\mathbf{y}=\boldsymbol{X}\boldsymbol{\beta}+\boldsymbol{\varepsilon}, \quad (1)$$

where $\boldsymbol{X} = (x_{ij})$ is an $n \times p$ regression matrix, $\mathbf{y} \in \mathbb{R}^n$ is the response vector, $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)'$ is the vector of regression coefficients, and $\boldsymbol{\varepsilon}$ is the vector of random errors with zero mean. The ordinary least squares (OLS) estimator of $\boldsymbol{\beta}$ is the solution to

$$\min_{\boldsymbol{\beta}} \|\mathbf{y}-\boldsymbol{X}\boldsymbol{\beta}\|^2, \quad (2)$$

where $\|\cdot\|$ denotes the Euclidean norm. If $\boldsymbol{X}$ is a part of a *known* $m \times p$ orthogonal matrix

$$\boldsymbol{X}_{\mathrm{c}}=\begin{pmatrix} \boldsymbol{X} \\ \boldsymbol{\Delta} \end{pmatrix}, \quad (3)$$

where $\boldsymbol{\Delta}$ is an $(m-n)\times p$ matrix, (2) can be efficiently computed by the Healy-Westmacott procedure (Healy and Westmacott 1956). Let

$$\mathbf{y}_{\mathrm{c}}=(\mathbf{y}', \mathbf{y}'_{\mathrm{miss}})' \quad (4)$$

be the vector of complete responses with missing data $\mathbf{y}_{\mathrm{miss}}$ of $m - n$ points. In each iteration, the procedure imputes the value of $\mathbf{y}_{\mathrm{miss}}$, and updates the OLS estimator for the complete data $(\boldsymbol{X}_{\mathrm{c}}, \mathbf{y}_{\mathrm{c}})$. This update involves no matrix inversion since $\boldsymbol{X}_{\mathrm{c}}$ is (column) orthogonal. Dempster, Laird, and Rubin (1977) showed that this procedure is an EM algorithm.

The major limitation of the procedure is the assumption that $\boldsymbol{X}$ must be embedded in a *pre-specified* orthogonal matrix $\boldsymbol{X}_c$. We propose a new algorithm, called *orthogonalizing EM* (OEM) algorithm, to remove this restriction and extend to other directions. The first step, called active orthogonization, orthogonalizes an arbitrary regression matrix by elaborately adding more rows. The second step imputes the responses of the new rows. The third step solves the OLS problem in (2) for the complete orthogonal design. The second and third steps iterate until convergence. The OEM procedure implicity adds rows and only requires the computation of the largest eigenvalue of $\boldsymbol{X}'\boldsymbol{X}$ in order to achieve convergence.

For the OLS problem in (2), OEM works with an arbitrary regression matrix $\boldsymbol{X}$. OEM can be extended to *penalized* least squares problems by adding penalties or constraints to β in (2). For penalized problems, the only difference in the OEM algorithm is the form of the solutions to the second and third steps. We demonstrate that OEM

1. Is faster than competing methods when $n \gg p$ for ordinary and penalized least squares
2. Converges to the Moore-Penrose generalized inverse for rank-deficient matrices faster than standard methods for both when $n > p$ and $p > n$
3. Computes solutions to penalization paths for multiple penalties efficiently
4. Results in estimates with *grouping coherence*, in that it yields the same coefficients for fully aliased columns in $\boldsymbol{X}$
5. Converges faster for penalized least squares than ordinary least squares

The remainder of this paper will be presented as follows. Section 2 discusses the active orthogonalization procedure. Section 3 presents OEM for OLS. Section 4 extends OEM to penalized least squares. Section 5 provides convergence properties of OEM. Section 6 presents numerical examples to compare OEM with other algorithms for penalized least squares. Section 7 concludes with some discussion.

2 ACTIVE ORTHOGONALIZATION

For an arbitrary $n \times p$ matrix $\boldsymbol{X}$ in (1), we propose *active orthogonalization* to actively orthogonalize an arbitrary matrix by elaborately adding more rows. Let $\boldsymbol{S}$ be a $p \times p$ diagonal matrix

$$\boldsymbol{S} = \operatorname{diag}(s_1, \ldots, s_p), \quad (5)$$

where $s_j > 0$ for $j = 1, \ldots, p$. Consider the eigenvalue decomposition $\boldsymbol{V}'\boldsymbol{\Gamma}\boldsymbol{V}$ of $\boldsymbol{S}^{-1}\boldsymbol{X}'\boldsymbol{X}\boldsymbol{S}^{-1}$ (Wilkinson 1965), where $\boldsymbol{V}$ is an orthogonal matrix and $\boldsymbol{\Gamma}$ is a diagonal matrix whose diagonal elements, $\gamma_1 \geq \cdots \geq \gamma_p$, are the nonnegative eigenvalues of $\boldsymbol{S}^{-1}\boldsymbol{X}'\boldsymbol{X}\boldsymbol{S}^{-1}$. For $d \geq \gamma_1$, let

$$t=\#\{j:\gamma_j=d,\ j=1,\ldots,p\} \quad (6)$$

denote the number of the γ_j equal d. For example, if $d=\gamma_1=\gamma_2$ and $\gamma_1>\gamma_j$ for $j=3,\ldots,p$, then $t=2$. If $d>\gamma_1$, then $t=0$. Define

$$\boldsymbol{B}=\mathrm{diag}(d-\gamma_{t+1},\ldots,d-\gamma_p) \quad (7)$$

and

$$\boldsymbol{\Delta}=\boldsymbol{B}^{1/2}\boldsymbol{V}_1\boldsymbol{S}, \quad (8)$$

where $\boldsymbol{V}_1$ is the submatrix of $\boldsymbol{V}$ consisting of the last $p-t$ rows. Put $\boldsymbol{X}$ and $\boldsymbol{\Delta}$ row by row together to form a complete matrix $\boldsymbol{X}_c$.

Lemma 1—The matrix $\boldsymbol{X}_c$ above is column orthogonal.

Lemma 1 indicates that at most $p-t$ rows need to be added to make $\boldsymbol{X}$ orthogonal. The following lemma shows that $p-t$ is actually the minimum value.

Lemma 2—Suppose that $\boldsymbol{\Delta}$ is an $l\times p$ matrix satisfying $\boldsymbol{X}'\boldsymbol{X}+\boldsymbol{\Delta}'\boldsymbol{\Delta}=\mathrm{diag}(d_1,\cdots,d_p)$ with $d_1,\cdots,d_p>0$. For any $d>0$, let $\boldsymbol{S}=\mathrm{diag}(\sqrt{d_1/d},\ldots,\sqrt{d_p/d})$ and $\gamma_1\geq\cdots\geq\gamma_p$ denote the eigenvalues of $\boldsymbol{S}^{-1}\boldsymbol{X}'\boldsymbol{X}\boldsymbol{S}^{-1}$. If $d\geq\gamma_1$, then $l\geq p-t$, where $t=\#\{j:\gamma_j=d, j=1,\ldots,p\}$.

We now consider the underlying geometry of active orthogonalization, which should provide intuition behind the OEM algorithm. For a vector $\mathbf{x}\in\mathbb{R}^m$, let $P_\omega\mathbf{x}$ denote its projection onto a subspace ω of $\mathbb{R}^m$. Lemma 1 implies that for the column vectors $\mathbf{x}_1,\ldots,\mathbf{x}_p\in\mathbb{R}^n$ of $\boldsymbol{X}$ in (1), there exists a set of mutually orthogonal vectors $\mathbf{x}_{c1},\ldots,\mathbf{x}_{cp}\in\mathbb{R}^{n+p-t}$ of $\boldsymbol{X}_c$ in (3) satisfying $P_{\mathbb{R}^n}\mathbf{x}_{ci}=\mathbf{x}_i$, for $j=1,\ldots,p$.

Remark 1—In (8) $\boldsymbol{\Delta}$ has $p-t$ rows, which does not rely on the number of rows in $\boldsymbol{X}$. Lemmas 1 and 2 indicate that $p-t$ rows need to be added to make $\boldsymbol{X}$ orthogonal.

Remark 2—The form of $\boldsymbol{S}$ in (5) can be chosen flexibly. One possibility is $\boldsymbol{S}=\boldsymbol{I}_p$ with

$$\boldsymbol{X}'\boldsymbol{X}+\boldsymbol{\Delta}'\boldsymbol{\Delta}=d\boldsymbol{I}_p \quad (9)$$

with $d\geq\gamma_1$, and $\boldsymbol{X}_c$ is standardized such that the Euclidean norm of each column is d. Note that larger t corresponds to fewer rows needed, and t depends on $\boldsymbol{S}$ and d. We can use the optimal $\boldsymbol{S}$ and d that maximize the value of t. Clearly $d=\gamma_1$ leads to the maximum t for a

given $\boldsymbol{S}$, whereas the optimal $\boldsymbol{S}$ has no obvious form. From the algorithmic aspect, a more reasonable strategy is to select them that optimize the convergence rate of the proposed algorithm; see the appendix for details on the convergence rate. However, it adds extra computational burden to solve such optimization problems, and we usually use explicit form of $\boldsymbol{S}$ like $\boldsymbol{S} = \boldsymbol{I}_p$. For standardized $\boldsymbol{X}$, another intuitive selection in (10) reduces to $\boldsymbol{S} = \boldsymbol{I}_p$.

The augmented rows $\boldsymbol{\Delta}$ need not be computed explicitly. Instead, we can compute $\boldsymbol{\Delta}'\boldsymbol{\Delta}$, which has a simple form, $d\boldsymbol{S}^2 - \boldsymbol{X}'\boldsymbol{X}$. As a direct result, only the largest eigenvalue of $\boldsymbol{S}^{-1}\boldsymbol{X}'\boldsymbol{X}\boldsymbol{S}^{-1}$ is required. Such a task can be very efficiently achieved using modern numerical linear algebra techniques.

Example 1—Suppose that $\boldsymbol{X}$ in (1) is orthogonal. Take $d = \gamma_1$ and

$$\boldsymbol{S}=\operatorname{diag}\left[\left(\sum_{i=1}^{n}x_{i1}^2\right)^{1/2},\ldots,\left(\sum_{i=1}^{n}x_{ip}^2\right)^{1/2}\right]. \quad (10)$$

Note that $\boldsymbol{S}^{-1}\boldsymbol{X}'\boldsymbol{X}\boldsymbol{S}^{-1}$ is an identity matrix. Consequently, $t = p$, and $\boldsymbol{\Delta}$ in (8) is empty, which indicates that active orthogonalization will not overshoot.

Example 2—Let

$$\boldsymbol{X}=\begin{pmatrix} 0 & 0 & 3/2 \\ -4/3 & -2/3 & 1/6 \\ 2/3 & 4/3 & 1/6 \\ -2/3 & 2/3 & -7/6 \end{pmatrix}.$$

If $\boldsymbol{S} = \boldsymbol{I}_3$ and $d = \gamma_1$, then $t = 2$ and (8) gives $\boldsymbol{\Delta}=(-2/\sqrt{3},\ 2/\sqrt{3},\ 1/\sqrt{3})$.

See the Supplementary Materials for further examples.

3 OEM FOR ORDINARY LEAST SQUARES

We now develop OEM for the OLS problem in (2) when the regression matrix $\boldsymbol{X}$ has an arbitrary form. OEM converges to the solution even if the model is misspecified. In practice, standard linear model diagnostic procedures should be enacted. Active orthogonalization is first employed to obtain $\boldsymbol{\Delta}$ in (8). If $\boldsymbol{X}_c$ is known (Healy and Westmacott 1956), then skip this step. For any initial estimator $\boldsymbol{\beta}^{(0)}$, the second step imputes $\mathbf{y}_{\text{miss}}$ in (4) with $\mathbf{y}_I = \boldsymbol{\Delta}\boldsymbol{\beta}^{(0)}$. Let $\mathbf{y}_c=(\mathbf{y}^{'},\mathbf{y}_I^{'})^{'}$. The third step solves

$$\boldsymbol{\beta}^{(1)}=\underset{\boldsymbol{\beta}}{\operatorname{argmin}}\|\mathbf{y}_c-\boldsymbol{X}_c\boldsymbol{\beta}\|^2. \quad (11)$$

Then, the second and third steps iterate for obtaining $\boldsymbol{\beta}^{(2)}, \boldsymbol{\beta}^{(3)}$, … until convergence. Define

$$\boldsymbol{A}=\boldsymbol{\Delta}'\boldsymbol{\Delta}. \quad (12)$$

For $\boldsymbol{X}_c$ in (3), let $(d_1, \ldots, d_p)$ denote the diagonal elements of $\boldsymbol{X}_c'\boldsymbol{X}_c$. For $k = 0, 1, \ldots$, let

$$\mathbf{u}=(u_1,\ldots,u_p)'=\boldsymbol{X}'\mathbf{y}+\boldsymbol{A}\boldsymbol{\beta}^{(k)}, \quad (13)$$

and (11) becomes

$$\boldsymbol{\beta}^{(k+1)}=\underset{\boldsymbol{\beta}}{\operatorname{argmin}}\sum_{j=1}^{p}(d_j\beta_j^2-2u_j\beta_j), \quad (14)$$

which is *separable* in the dimensions of $\boldsymbol{\beta}$. Thus, (14) has a simple form

$$\beta_j^{(k+1)}=u_j/d_j, \quad \text{for } j=1,\ldots,p, \quad (15)$$

which involves no matrix inversion. If $\boldsymbol{X}_c$ is constructed by active orthogonalization with $\boldsymbol{S}$ in (5), then $d_j=ds_j^2$ in (15) for $j = 1, \ldots, p$.

The above algorithm works for any $\boldsymbol{S}$ in (5). For simplicity, we fix $\boldsymbol{S} = \boldsymbol{I}_p$ in the remaining part of this section; see Remark 2 for a discussion on selection of $\boldsymbol{S}$. In practice, obtaining $\boldsymbol{\Delta}$ in (8) may be computationally expensive. Instead, one can simply compute $\boldsymbol{A} = \boldsymbol{\Delta}'\boldsymbol{\Delta}$ and the diagonal entries $d_1, \ldots, d_p$ of $\boldsymbol{X}_c'\boldsymbol{X}_c$. For $\boldsymbol{S} = \boldsymbol{I}_p$, by (8), $\boldsymbol{A} = d\boldsymbol{I}_p - \boldsymbol{X}'\boldsymbol{X}$, where d is a number no less than the largest eigenvalue γ_1 of $\boldsymbol{X}'\boldsymbol{X}$. This essentially reduces OEM to finding a reasonable d, which can be accomplished efficiently. A possible choice is $d = \operatorname{trace}(\boldsymbol{X}'\boldsymbol{X})$. Another choice is $d = \gamma_1$ to obtain the fastest convergence; see the appendix. We compute γ_1 by the Lanczos algorithm (Lanczos 1950) described in the Supplementary Materials.

Remark 3—When $p > n$, we ease the computational burden by replacting the $p \times p$ matrix $\boldsymbol{X}'\boldsymbol{X}$ with the $n \times n$ matrix $\boldsymbol{X}\boldsymbol{X}'$ in the Lanczos method as the two matrices have identical non-zero eigenvalues.

When $\boldsymbol{X}$ has full column rank, the convergence results in Wu (1983) indicates that the OEM sequence given by (15) converges to the OLS estimator for any initial point $\boldsymbol{\beta}^{(0)}$. Next, we discuss the convergence property of OEM when $\boldsymbol{X}'\boldsymbol{X}$ is singular, which covers the case of $p > n$. Let r denote the rank of $\boldsymbol{X}$. For $r < p$, the singular value decomposition (Wilkinson 1965) of $\boldsymbol{X}$ is

$$\boldsymbol{X}=\boldsymbol{U}'\begin{pmatrix}\boldsymbol{\Gamma}_0^{1/2} & \mathbf{0}\\ \mathbf{0} & \mathbf{0}\end{pmatrix}\boldsymbol{V},$$

where $\boldsymbol{U}$ is an $n \times n$ orthogonal matrix, $\boldsymbol{V}$ is a $p \times p$ orthogonal matrix, and $\boldsymbol{\Gamma}_0$ is a diagonal matrix with diagonal elements $\gamma_1 \geq \cdots \geq \gamma_r$ which are the positive eigenvalues of $\boldsymbol{X}'\boldsymbol{X}$. Define

$$\hat{\boldsymbol{\beta}}^* = (\boldsymbol{X}'\boldsymbol{X})^+\boldsymbol{X}'\mathbf{y}, \quad (16)$$

where $^+$ denotes the Moore-Penrose generalized inverse (Ben-Israel and Greville 2003).

Theorem 1—Suppose that $\boldsymbol{X}'\boldsymbol{X} + \boldsymbol{\Delta}'\boldsymbol{\Delta} = \gamma_1\boldsymbol{I}_p$. If $\boldsymbol{\beta}^{(0)}$ lies in the linear space spanned by the first r columns of $\boldsymbol{V}'$, then as $k \to \infty$, for the OEM sequence $\{\boldsymbol{\beta}^{(k)}\}$ of the ordinary least squares, $\boldsymbol{\beta}^{(k)} \to \hat{\boldsymbol{\beta}}^*$.

In active orthogonalization, the condition $\boldsymbol{X}'\boldsymbol{X} + \boldsymbol{\Delta}'\boldsymbol{\Delta} = \gamma_1\boldsymbol{I}_p$ holds if $d = \gamma_1$ and $\boldsymbol{S} = \boldsymbol{I}_p$ in (8). Using $\boldsymbol{\beta}^{(0)} = \mathbf{0}$ satisfies the condition in Theorem 1.

The Moore-Penrose generalized inverse is widely used in statistics for a degenerated system. Theorem 1 indicates that OEM converges to $\hat{\boldsymbol{\beta}}^*$ in (16) in this case. When $r < p$, the limiting vector $\hat{\boldsymbol{\beta}}^*$ given by an OEM sequence has the following properties. First, it has the minimal Euclidean norm among the least squares estimators $(\boldsymbol{X}'\boldsymbol{X})^-\boldsymbol{X}'\mathbf{y}$ (Ben-Israel and Greville 2003). Second, its model error has a simple form, $E[(\hat{\boldsymbol{\beta}}^*-\boldsymbol{\beta})'(\boldsymbol{X}'\boldsymbol{X})(\hat{\boldsymbol{\beta}}^*-\boldsymbol{\beta})] = r\sigma^2$. Third, $\boldsymbol{X}\boldsymbol{\alpha} = \mathbf{0}$ implies $\boldsymbol{\alpha}'\hat{\boldsymbol{\beta}}^* = 0$ for any vector $\boldsymbol{\alpha}$. The third property indicates that $\hat{\boldsymbol{\beta}}^*$ inherits the multicollinearity between the columns in $\boldsymbol{X}$. This property is stronger than grouping coherence for penalized least squares in the appendix.

We now discuss the computational efficiency of OEM for computing $\hat{\boldsymbol{\beta}}^*$ in (16) when $\boldsymbol{X}$ is degenerated. Recall that $\boldsymbol{X}'\boldsymbol{X}$ and $\boldsymbol{X}\boldsymbol{X}'$ have the same nonzero eigenvalues. The computation of γ_1 in the OEM iterations by the Lanczos algorithm has complexity $O(\min\{n, p\}^2 \max\{n, p\})$. Since the complexity of the OEM iterations is $O(np^2)$, the whole computational complexity of OEM for computing $\hat{\boldsymbol{\beta}}^*$ is $O(np^2)$. The singular value decomposition method computes $(\boldsymbol{X}'\boldsymbol{X})^+$ first by singular value decomposition to obtain $\hat{\boldsymbol{\beta}}^*$, and has computational complexity $O(np^2 + p^3)$. The OEM algorithm is superior to this method in terms of complexity for large p.

We conduct a simulation study to compare the speeds of OEM and the singular value decomposition-based least squares method for computing $\hat{\boldsymbol{\beta}}^*$ in (16). The MATLAB function pinv is used in the singular value decomposition method as above. Generate all entries of $\boldsymbol{X}$ and $\mathbf{y}$ independently from the standard normal distribution. A new predictor calculated as the mean of all the covariates is added to degenerate the design matrix. Table 1 compares a simple MATLAB implementation of OEM and the singular value decomposition method in computing $\hat{\boldsymbol{\beta}}^*$. Both OEM and the singular value decomposition method give the same results. The results for $p > n$ situations are similar to those of $n > p$ and are included in the

Supplementary Materials. Tables 1 and in Section C.1 in the Supplementary Materials indicate that OEM is faster than the singular value decomposition method for any combination of n and p, which is consistent with the above complexity analysis.

4 OEM FOR PENALIZED LEAST SQUARES

The OEM procedure can be easily extended to penalized least squares problems. Consider a penalized version of (1):

$$\min_{\boldsymbol{\beta}\in\Theta} \left[\|\mathbf{y}-\boldsymbol{X}\boldsymbol{\beta}\|^2+P(\boldsymbol{\beta};\lambda)\right], \quad (17)$$

where $\boldsymbol{\beta} \in \boldsymbol{\Theta}$, $\boldsymbol{\Theta}$ is a subset of $\mathbb{R}^p$, P is a penalty function, and λ is the vector of tuning parameters. To apply the penalty P equally to all the variables, the regression matrix $\boldsymbol{X}$ is standardized so that

$$\sum_{i=1}^{n} x_{ij}^2=1, \text{ for } j=1,\ldots,p. \quad (18)$$

Popular choices for P include ridge regression (Hoerl and Kennard 1970), the nonnegative garrote (Breiman 1995), the lasso (Tibshirani 1996), SCAD (Fan and Li 2001), and the MCP (Zhang 2010).

Suppose that $\boldsymbol{\Theta}$ and P in (17) are *decomposable* as $\Theta=\prod_{j=1}^{p}\Theta_j$ and $P(\boldsymbol{\beta};\lambda)=\sum_{j=1}^{p}P_j(\beta_j;\lambda)$. For the problem in (17), the first step of OEM is active orthogonalization, which computes $\boldsymbol{\Delta}$ in (8). For any initial estimator $\boldsymbol{\beta}^{(0)}$, the second step imputes $\mathbf{y}_{\text{miss}}$ in (4) by $\mathbf{y}_{\text{I}} = \boldsymbol{\Delta}\boldsymbol{\beta}^{(0)}$. Let $\mathbf{y}_{\text{c}}=(\mathbf{y}', \mathbf{y}_{\text{I}}')'$. The third step solves

$$\boldsymbol{\beta}^{(1)}=\underset{\boldsymbol{\beta}\in\Theta}{\operatorname{argmin}} \left[\|\mathbf{y}_{\text{c}}-\boldsymbol{X}_{\text{c}}\boldsymbol{\beta}\|^2+P(\boldsymbol{\beta};\lambda)\right].$$

The second and third steps iterate to compute $\boldsymbol{\beta}^{(k)}$ for $k = 1, 2, \ldots$ until convergence. Similar to (14), we have an iterative formula

$$\beta_j^{(k+1)}=\arg\min_{\beta_j\in\Theta_j} [d_j\beta_j^2-2u_j\beta_j+P_j(\beta_j;\lambda)], \text{ for } j=1,\ldots,p, \quad (19)$$

with $\mathbf{u} = (u_1, \ldots, u_p)'$ defined in (13). This shortcut as applied to the Lasso penalty (Tibshirani 1996) is as follows:

With $\boldsymbol{\Theta}_j = \mathbb{R}$,

$$P_j(\beta_j;\lambda)=2\lambda|\beta_j|, \quad (20)$$

and (19) becomes

$$\beta_j^{(k+1)}=\text{sign}(u_j)\left(\frac{|u_j|-\lambda}{d_j}\right)_+. \quad (21)$$

Here, for $a \in \mathbb{R}$, $(a)_+$ denotes $\max\{a, 0\}$. The formulas for further penalties are included in the Supplementary Materials.

OEM for (17) is also an EM algorithm. Let the observed data $\mathbf{y}$ follow the model in (1). Assume that the complete data $\mathbf{y}_c=(\mathbf{y}', \mathbf{y}'_{\text{miss}})'$ in (4) follows a regression model $\mathbf{y}_c = \boldsymbol{X}_c\boldsymbol{\beta} + \boldsymbol{\varepsilon}_c$, where $\boldsymbol{\varepsilon}_c$ is from $N(\mathbf{0}, \boldsymbol{I}_m)$. Let $\hat{\boldsymbol{\beta}}$ be a solution to (17) given by $\hat{\boldsymbol{\beta}} = \text{argmax}_{\boldsymbol{\beta}\in\boldsymbol{\Theta}} L(\boldsymbol{\beta} \mid \mathbf{y})$, and the penalized likelihood function $L(\boldsymbol{\beta} \mid \mathbf{y})$ is

$$(2\pi)^{-n/2}\exp\left(-\frac{1}{2}\|\mathbf{y}-\boldsymbol{X}\boldsymbol{\beta}\|^2\right)\exp\left[-\frac{1}{2}P(\boldsymbol{\beta};\lambda)\right].$$

Given $\boldsymbol{\beta}^{(k)}$, the second step of OEM for (17) is the E-step,

$$\begin{aligned}E[\log\{L(\boldsymbol{\beta}|\mathbf{y}_c)\}|\mathbf{y},\boldsymbol{\beta}^{(k)}]&=-C\{\|\mathbf{y}-\boldsymbol{X}\boldsymbol{\beta}\|^2+E(\|\mathbf{y}_{\text{miss}}-\boldsymbol{X}\boldsymbol{\beta}\|^2|\boldsymbol{\beta}^{(k)})+P(\boldsymbol{\beta};\lambda)\}\\&=-C\{n+\|\mathbf{y}-\boldsymbol{X}\boldsymbol{\beta}\|^2+\|\boldsymbol{\Delta}\boldsymbol{\beta}^{(k)}-\boldsymbol{\Delta}\boldsymbol{\beta}\|^2+P(\boldsymbol{\beta};\lambda)\}\end{aligned}$$

for some constant $C > 0$. Define

$$Q(\boldsymbol{\beta}|\boldsymbol{\beta}^{(k)})=\|\mathbf{y}-\boldsymbol{X}\boldsymbol{\beta}\|^2+\|\boldsymbol{\Delta}\boldsymbol{\beta}^{(k)}-\boldsymbol{\Delta}\boldsymbol{\beta}\|^2+P(\boldsymbol{\beta};\lambda). \quad (22)$$

The third step of OEM is the M-step,

$$\boldsymbol{\beta}^{(k+1)}=\underset{\boldsymbol{\beta}\in\Theta}{\text{argmin}}Q(\boldsymbol{\beta}|\boldsymbol{\beta}^{(k)}), \quad (23)$$

which is equivalent to (19) when $\boldsymbol{\Theta}$ and P in (17) are decomposable.

Example 3—For the model in (1), let the complete matrix $\boldsymbol{X}_c$ be an orthogonal design from Xu (2009) with 4096 runs in 30 factors. Let $\boldsymbol{X}$ in (1) be the submatrix of $\boldsymbol{X}_c$ consisting of the first 3000 rows and let $\mathbf{y}$ be generated from (1) with $\sigma = 1$ and

$$\beta_j = (-1)^j \exp[-2(j-1)/20] \text{ for } j = 1, \ldots, p. \quad (24)$$

Here, let $p = 30$, $n = 3000$, and the response values for the last 1096 rows of $\boldsymbol{X}_c$ be missing. OEM is used to solve the SCAD problem with an initial value $\boldsymbol{\beta}^{(0)} = \mathbf{0}$ and a stopping criterion when relative changes in all coefficients are less than 10^{-6}. For $\lambda = 1$ and $a = 3.7$ in the SCAD penalty, defined in the Supplementary Materials, Figure 1 plots values of the objective function in (17) with the SCAD penalty of the OEM sequence against iteration numbers, where the convergence occurs at iteration 13, and the objective function significantly reduces after two iterations.

5 CONVERGENCE OF THE OEM ALGORITHM

We now establish convergence properties of OEM with the general penalty in (17). We also give results to compare the convergence rates of OEM for OLS, the elastic-net, and the lasso. These convergence rate results show that for the same data set, OEM converges faster for penalized least squares than ordinary least squares. This provides a new theoretical comparison between these methods. The objective functions of existing EM convergence results like those in Wu (1983), Green (1990) and McLachlan and Krishnan (2008) are typically continuously differentiable. This condition does not hold for the objective function in (17) with the lasso and other penalties, and these existing results do not directly apply here.

For the model in (1), denote the objective function in (17) by

$$l(\boldsymbol{\beta}) = \|\mathbf{y} - \boldsymbol{X}\boldsymbol{\beta}\|^2 + P(\boldsymbol{\beta}; \lambda), \quad (25)$$

which is defined on a subset $\boldsymbol{\Theta}$ of $\mathbb{R}^p$. For penalties like the bridge, it is infeasible to perform the M-step in (23) directly. For this situation, following the generalized EM algorithm in Dempster, Laird, and Rubin (1977), we define the following *generalized OEM* algorithm

$$\boldsymbol{\beta}^{(k)} \to \boldsymbol{\beta}^{(k+1)} \in \mathcal{M}(\boldsymbol{\beta}^{(k)}), \quad (26)$$

where $\boldsymbol{\beta} \to \mathcal{M}(\boldsymbol{\beta}) \subset \boldsymbol{\Theta}$ is a point-to-set map such that

$$Q(\boldsymbol{\phi}|\boldsymbol{\beta}) \leq Q(\boldsymbol{\beta}|\boldsymbol{\beta}), \quad \text{for all } \boldsymbol{\phi} \in \mathcal{M}(\boldsymbol{\beta}).$$

Here, Q is given in (22). The OEM sequence defined by (23) is a special case of (26). For example, the generalized OEM algorithm can be used for the bridge penalty (Frank and Friedman 1993), where $\Theta_j = \mathbb{R}$ and

$$P_j(\beta_j;\lambda)=\lambda|\beta_j|^a \quad (27)$$

for some $a \in (0, 1)$ in (17). Since the solution to (19) with the bridge penalty has no closed form, one may use one-dimensional search to compute $\beta_j^{(k+1)}$ that satisfies (26).

We make the following three assumptions throughout this section.

Assumption 1. The parameter space Θ is a closed convex subset of $\mathbb{R}^p$.

Assumption 2. For a fixed λ, the penalty $P(\beta; \lambda)$ in (17) is continuous with respect to $\beta \in \Theta$.

Assumption 3. The sequence $\{\beta^{(k)}\}$ defined in (26) lies in a compact subset of $\mathbb{R}^p$.

Assumption 1 covers the case in which the sequence $\{\beta^{(k)}\}$ may fall on the boundary of Θ (Nettleton 1999), like the nonnegative garrote (Breiman 1995) and the nonnegative lasso (Efron et al. 2004). The generalized OEM algorithms for the penalties discussed in Section 4 all satisfy Assumptions 1 and 2. By the monotonicity property of the generalized EM algorithm, Assumption 3 holds if we have the lower compact condition that $\{\beta \in \Theta : l(\beta) \leq C\}$ is compact for any constant C. This condition is generally required in the convergence analysis of MM algorithms (Lange 1999), which are extensions of EM algorithms. It is obvious that the lower compact condition holds if $l(\beta) \to \infty$ as $\|\beta\| \to \infty$. Therefore, for SCAD and MCP, the lower compact condition holds if $\boldsymbol{X}$ has full column rank; for other penalties in Section 4, it holds for any $\boldsymbol{X}$ and $\mathbf{y}$. The generalized OEM algorithm for the bridge penalty in (27) also satisfies the above assumptions.

Remark 4—When p is larger than n, the lower compact condition does not hold for SCAD or MCP. For this case, Assumption 3 also holds if there exists some k such that $\beta^{(k)}$ lies in the set $\{\beta \in \Theta : l(\beta) < \inf \mathscr{L}\}$, where $\mathscr{L}$ is the set of limit points of all $\{l(\varphi_n)\}$ with $\varphi_n \in \Theta$ and $\|\varphi_n\| \to \infty$ as $n \to \infty$.

The objective functions in (17) with the lasso and other penalties are not continuously differentiable. A more general definition of stationary points is needed. We call $\beta \in \Theta$ a stationary point of l if

$$\liminf_{t \to 0_+} \frac{l((1-t)\boldsymbol{\beta}+t\boldsymbol{\phi})-l(\boldsymbol{\beta})}{t} \geq 0 \quad \text{for all } \boldsymbol{\phi} \in \Theta.$$

Let S denote the set of stationary points of l. By Assumption 2, $\mathcal{M}$ is a closed point-to-set map (Zangwill 1969; Wu 1983). Analogous to Theorem 1 in Wu (1983) on the global convergence of the EM algorithm, we have the following result.

Theorem 2—Let $\{\boldsymbol{\beta}^{(k)}\}$ be a generalized OEM sequence generated by (26). Suppose that

$$l(\boldsymbol{\beta}^{(k+1)}) < l(\boldsymbol{\beta}^{(k)}) \quad \text{for all } \boldsymbol{\beta}^{(k)} \in \Theta \backslash S. \tag{28}$$

Then all limit points of $\{\boldsymbol{\beta}^{(k)}\}$ are elements of S and $l(\boldsymbol{\beta}^{(k)})$ converges monotonically to $l^* = l(\boldsymbol{\beta}^*)$ for some $\boldsymbol{\beta}^* \in S$.

Theorem 3—If $\boldsymbol{\beta}^*$ is a local minimum of $Q(\boldsymbol{\beta} \mid \boldsymbol{\beta}^*)$, then $\boldsymbol{\beta}^* \in S$.

This theorem follows from the fact that $l(\boldsymbol{\beta}) - Q(\boldsymbol{\beta} \mid \boldsymbol{\beta}^*)$ is differentiable and

$$\frac{\partial [l(\boldsymbol{\beta}) - Q(\boldsymbol{\beta} | \boldsymbol{\beta}^*)]}{\partial \boldsymbol{\beta}} \Big|_{\boldsymbol{\beta} = \boldsymbol{\beta}^*} = 0.$$

Remark 5—By Theorem 3, if $\boldsymbol{\beta}^{(k)} \notin S$, then $\boldsymbol{\beta}^{(k)}$ cannot be a local minimum of $Q(\boldsymbol{\beta} \mid \boldsymbol{\beta}^{(k)})$. Thus, there exists at least one point $\boldsymbol{\beta}^{(k+1)} \in \mathcal{M}(\boldsymbol{\beta}^{(k)})$ such that $Q(\boldsymbol{\beta}^{(k+1)} \mid \boldsymbol{\beta}^{(k)}) < Q(\boldsymbol{\beta}^{(k)} \mid \boldsymbol{\beta}^{(k)})$ and therefore satisfies the condition in (28). As a special case, an OEM sequence generated by (23) satisfies (28) in Theorem 2.

Next, we derive convergence results of a generalized OEM sequence $\{\boldsymbol{\beta}^{(k)}\}$ in (26), which, by Theorem 3, hold automatically for an OEM sequence. If the penalty function $P(\boldsymbol{\beta}; \lambda)$ is convex and $l(\boldsymbol{\beta})$ has a unique minimum, Theorem 4 shows that $\{\boldsymbol{\beta}^{(k)}\}$ converges to the global minimum.

Theorem 4—For $\{\boldsymbol{\beta}^{(k)}\}$ defined in Theorem 2, suppose that $l(\boldsymbol{\beta})$ in (25) is a convex function on $\boldsymbol{\Theta}$ with a unique minimum $\boldsymbol{\beta}^*$ and that (28) holds for $\{\boldsymbol{\beta}^{(k)}\}$. Then $\boldsymbol{\beta}^{(k)} \to \boldsymbol{\beta}^*$ as $k \to \infty$.

Theorem 5 discusses the convergence of an OEM sequence $\{\boldsymbol{\beta}^{(k)}\}$ for more general penalties. For $a \in \mathbb{R}$, define $S(a) = \{\boldsymbol{\varphi} \in S : l(\boldsymbol{\varphi}) = a\}$. From Theorem 2, all limit points of an OEM sequence are in $S(l^*)$, where l^* is the limit of $l(\boldsymbol{\beta}^{(k)})$ in Theorem 2. Theorem 5 states that the limit point is unique under certain conditions.

Theorem 5—Let $\{\boldsymbol{\beta}^{(k)}\}$ be a generalized OEM sequence generated by (26) with $\boldsymbol{\Delta}'\boldsymbol{\Delta} > 0$. If (28) holds, then all limit points of $\{\boldsymbol{\beta}^{(k)}\}$ are in a connected and compact subset of $S(l^*)$. In particular, if the set $S(l^*)$ is discrete in that its only connected components are singletons, then $\boldsymbol{\beta}^{(k)}$ converges to some $\boldsymbol{\beta}^*$ in $S(l^*)$ as $k \to \infty$.

6 NUMERICAL ILLUSTRATIONS FOR SOLVING PENALIZED LEAST SQUARES

Many algorithms for solving the penalized least squares problem in (17) are available, including the LARS algorithm introduced in Efron, Hastie, Johnstone, and Tibshirani (2004) and the coordinate descent (CD) algorithm (Tseng 2001; Friedman, Hastie, Hofling and Tibshirani 2007; Wu and Lange 2008; Tseng and Yun 2009). The corresponding R packages include lars (Hastie and Efron 2011), and glmnet (Friedman, Hastie, and Tibshirani 2011). For nonconvex penalties in (17) like SCAD and MCP, existing algorithms for solving this optimization problem include local quadratic approximation (Fan and Li 2001; Hunter and Li 2005), local linear approximation (Zou and Li 2008), the CD algorithm (Breheny and Huang 2010; Mazumder, Friedman, and Hastie 2011) and the minimization by iterative soft thresholding algorithm (Schifano, Strawderman, and Wells 2010), among others. We demonstrate that in big data settings when $n \gg p$, our OEM implementation vastly outperforms coordinate descent, as implemented in glmnet, which is considered one of the fastest algorithms for penalized regression problems. OEM is less efficient when $p \gg n$. To mitigate this, computation can be vastly reduced via screening methods. Here we compare OEM with the CD, LARS, and glmnet algorithms for penalized least squares.

6.1 COMPARISONS WITH OTHER ALGORITHMS

We now compare the computational efficiency of OEM for penalized least squares problems with the coordinate descent (CD) algorithm. OEM is implemented in R with main code in C ++. For fitting the lasso, we compare OEM and the R package glmnet, which uses Fortran for the main computation. For MCP, we compare OEM with a C implementation of the CD algorithm in the R package ncvreg and with a slightly different CD implementation in Fortran in the R package sparsenet.

We consider the situation when the sample size n is larger than the number of variables p. For the penalized problems, solutions for 100 tuning parameter values are computed for all methods. Three different covariance matrix structures for the predictor variables are compared. The first is the case where all the variables are independently generated from standard normal distribution, the second and third cases involve design matrices with a correlations structure

$$\mathrm{Cor}(X_i, X_j) = \rho^{|i-j|} \text{ for } i, j = 1, \ldots, p, \quad (29)$$

where $\rho = 0, 0.2, 0.8$. The response is generated as a linear combination of the design matrix and the true model is $\mathbf{y} = \mathbf{X}\beta + \boldsymbol{\varepsilon}$, where $\boldsymbol{\varepsilon}$ follows the normal distribution $N(\mathbf{0}, \boldsymbol{I}_n)$, β is independent of $\boldsymbol{\varepsilon}$ and follows the normal distribution $N(\mathbf{0}, \boldsymbol{I}_p)$.

To compare the performance of OEM, CD implemented in the ncvreg package (Breheny, et al 2011), and sparsenet (Mazumder, et al 2011) algorithms for nonconvex penalties, data are generated 10 times and the average runtime are shown in Table 3. The results indicate that OEM has advantages when the sample size is significantly larger than the number of

variables especially for the independent design. All algorithms tend to require more fitting time when the correlations among the covariates increase. The results are similar for the SCAD penalty for OEM and CD and are not presented, but sparsenet lacks an implementation of SCAD. Table 2 in the Supplementary Materials shows that, for computing SCAD, the CD algorithm is faster and the computational gap gets wider when the ratio of p/n increases. This phenomenon also occurs for the lasso.

A close scrutiny of the two algorithms reveals that they take similar number of iterations but the computation of OEM required a one time computation of matrix multiplication $\boldsymbol{X}'\boldsymbol{X}$ and the complexity of this process is $O(np^2)$, which dominates the algorithm especially when p is very large. This is the main drawback of the OEM algorithm.

An advantageous and unique property of the OEM algorithm is its ability to provide solutions for full tuning parameter paths for multiple penalties simultaneously with minimal relative added computation, especially as sample size increases. If in a particular application a comparison between many different penalties is required, OEM provides a way of computing them all at once. While many methods such as coordinate descent have been shown to be efficient in computing solutions to full penalization paths, this result is different in that OEM is efficient in the computation of solutions for full penalization paths for multiple penalties. To the best of our knowledge, no other algorithm possesses this property. A demonstration of this provided in Table 4. We compare the speed of OEM in computing a path of 100 tuning parameter values for the lasso penalty with that of OEM for paths of 100 tuning parameter values for each of the lasso, SCAD, and MCP penalties all at once.

6.2 LARGE-SCALE PERFORMANCE

In many large scale applications, the regression matrix $\boldsymbol{X}$ cannot be fit into memory. Due to its structure, the OEM algorithm extends naturally to such problems. The key computational components of the OEM algorithm, $\boldsymbol{A}$ and $\boldsymbol{X}'\mathbf{y}$, can be computed in a block row-wise fashion, which removes the need to store the entirety of $\boldsymbol{X}$ in memory at once. Specifically, under this scheme, $\boldsymbol{A}$ is computed iteratively by partitioning $\boldsymbol{X}$ into manageable submatrices $\boldsymbol{X}_c$: $c = 1, \ldots, C$ and computing $\boldsymbol{X}'\boldsymbol{X} = \sum_{c=1}^{C} \boldsymbol{X}_c'\boldsymbol{X}_c$. $\boldsymbol{X}'\mathbf{y}$ can be computed similarly. This allows OEM to scale naturally to settings where the number of observations is arbitrarily large. Methods based on CD do not naturally scale in this sense, as the entire coordinate gradient must be formed at each iteration.

In order to demonstrate the utility of OEM for extremely large-scale applications, a simulation with large sample sizes was conducted. For the dense matrix simulation, the data were simulated like in Table 2 with $\rho = 0$. As many applications such as text analysis result in regression matrices with mostly zero values, a sparse setting was investigated. In this setting the regression matrix $\boldsymbol{X}$ is much larger than in the dense simulation, but a majority of the values are 0. $\boldsymbol{A}$ was computed in blocks of size 10^6 for the dense simulation and of size 10^7 for the sparse simulation. Results are shown in Table 5.

6.3 DATA ANALYSIS

Consider a dataset from US Census Bureau County and City Data Book 2007, United States Department of Commerce (2007). The response is population change in percentage. The 36 covariates are described in the Supplementary Materials. These variables are in percentage of population of the individual counties.

There are 2573 (counties) observations without missing observations. The linear regression model in (1) is used to fit the data. The solution paths for the lasso, SCAD and MCP fitted to the data set are given in the Supplementary Materials. The number of non-zero coefficients, cross validation residual sum of squares, AIC and BIC are presented in Table 6, where the tuning parameter λ is chose by BIC.

The significant variables reveal that the population change is highly related to the living standards of the counties. Table 6 compares the fitted models from different penalized least squares problems. Note that MCP has the most sparse model with little sacrifice of CV error, AIC and BIC scores. LASSO has the model with smallest CV error but including nearly all the candidate predictors. In the example, the penalized models favor complex models with many nonzero coefficients and this reveals the fact that there are many factors that have profound influence on population change of counties in the US. In addition, the last two columns of Table 6 also give the runtime of fitting the 10-fold cross-validation to the data, where OEM is implemented in R with main code in C++, LASSO with CD from glmnet, and SCAD and MCP from ncvreg.

7 DISCUSSION

We have proposed a new algorithm called OEM for solving ordinary and penalized least squares problems for general data structures. We have showed that in order to *actively orthogonalize* a regression matrix $\boldsymbol{X}$, one need not explicitly add new rows This can be avoided by a procedure that only requires the largest eigenvalue of $\boldsymbol{X}'\boldsymbol{X}$. For big data problems, we have demonstrated that OEM outperforms the competing methods and shows great potential in many modern large-scale applications. In addition to its numerical performance, OEM has several desirable theoretical properties, including convergence to the Moore-Penrose generalized inverse-based least squares estimator for singular regression matrices and convergence to a point having grouping coherence for the lasso, SCAD or MCP. For applications such as micro-array, one might be interested in extending the result to the small n and large p case, where OEM is generally slower than state-of-the-art algorithms. This suggests a new interface between optimization and statistics for penalized methods.

The algorithm can be sped up by using various methods from the EM literature (McLachlan and Krishnan 2008). A detailed discussion of this topic is included in the Supplementary Materials.

Supplementary Material

Refer to Web version on PubMed Central for supplementary material.

Acknowledgments

We thank the editor, associate editor and referees for their constructive comments, which have substantially improved the paper. Xiongs research was supported by the National Natural Science Foundation of China (Grant No. 11271355, 11471172).

References

Ben-Israel, A.; Greville, TNE. Generalized Inverses, Theory and Applications. 2. New York: Springer; 2003.

Breiman L. Better Subset Regression Using the Nonnegative Garrote. Technometrics. 1995; 37:373–384.

Breheny P, Huang J. Coordinate Descent Algorithms for Nonconvex Penalized Regression, With Applications to Biological Feature Selection. The Annals of Applied Statistics. 2011; 5:232–253. [PubMed: 22081779]

Bühlmann, P.; van de Geer, S. Statistics for High-Dimensional Data: Methods, Theory and Applications. Berlin: Springer; 2011.

Dempster AP, Laird NM, Rubin DB. Maximum Likelihood from Incomplete Data via the EM Algorithm. Journal of the Royal Statistical Society, Ser B. 1977; 39:1–38.

Efron B, Hastie T, Johnstone I, Tibshirani R. Least Angle Regression. The Annals of Statistics. 2004; 32:407–451.

Fan J, Li R. Variable Selection via Nonconcave Penalized Likelihood and Its Oracle Properties. Journal of the American Statistical Association. 2001; 96:1348–1360.

Fan J, Lv J. Sure Independence Screening for Ultrahigh Dimensional Feature Space (with discussion). Journal of the Royal Statistical Society, Ser B. 2008; 70:849–911.

Fan J, Lv J. Properties of Non-concave Penalized Likelihood with NP-dimensionality. Information Theory, IEEE Transactions. 2011; 57:5467–5484.

Fan J, Peng H. Non-concave Penalized Likelihood With Diverging Number of Parameters. The Annals of Statistics. 2004; 32:928–961.

Frank LE, Friedman J. A Statistical View of Some Chemometrics Regression Tools. Technometrics. 1993; 35:109–135.

Friedman J, Hastie T, Hofling H, Tibshirani R. Pathwise Coordinate Optimization. The Annals of Applied Statistics. 2007; 1:302–332.

Friedman J, Hastie T, Tibshirani R. Regularization Paths for Generalized Linear Models via Coordinate Descent. Journal of Statistical Software. 2009; 33:1–22. [PubMed: 20808728]

Friedman, J.; Hastie, T.; Tibshirani, R. “Glmnet,” R package. 2011.

Green PJ. On Use of the EM Algorithm for Penalized Likelihood Estimation. Journal of the Royal Statistical Society, Ser B. 1990; 52:443–452.

Hastie, T.; Efron, B. “Lars,” R package. 2011.

Healy MJR, Westmacott MH. Missing Values in Experiments Analysed on Automatic Computers. Journal of the Royal Statistical Society, Ser C. 1956; 5:203–206.

Hoerl AE, Kennard RW. Ridge Regression: Biased Estimation for Nonorthogonal Problems. Technometrics. 1970; 12:55–67.

Hunter DR, Li R. Variable Selection Using MM Algorithms. The Annals of Statistics. 2005; 33:1617–1642. [PubMed: 19458786]

Huo X, Chen J. Complexity of Penalized Likelihood Estimation. Journal of Statistical Computation and Simulation. 2010; 80:747–759.

Huo X, Ni XL. When Do Stepwise Algorithms Meet Subset Selection Criteria? The Annals of Statistics. 2007; 35:870–887.

Lanczos C. An iteration method for the solution of the eigenvalue problem of linear differential and integral operators. Journal of Research of the National Bureau of Standards. 1950; 45:255–282.

Lange, K. Numerical Analysis for Statisticians. New York: Springer; 1999.

Mazumder R, Friedman J, Hastie T. SparseNet: Coordinate Descent with Non-Convex Penalties. Journal of the American Statistical Association. 2011; 106:1125–1138. [PubMed: 25580042]

Meinshausen N, Yu B. Lasso-Type Recovery of Sparse Representations for High-Dimensional Data. The Annals of Statistics. 2009; 37:246–270.

McLachlan, G.; Krishnan, T. The EM Algotithm and Extensions. 2. New York: Wiley; 2008.

Nettleton D. Convergence Properties of the EM Algorithm in Constrained Parameter Spaces. Canadian Journal of Statistics. 1999; 27:639–648.

Owen AB. A Robust Hybrid of Lasso and Ridge Regression. Technical Report. 2006

Schifano ED, Strawderman R, Wells MT. Majorization-Minimization Algorithms for Nonsmoothly Penalized Objective Functions. Electronic Journal of Statistics. 2010; 23:1258–1299.

Tibshirani R. Regression Shrinkage and Selection via the Lasso. Journal of the Royal Statistical Society, Ser B. 1996; 58:267–288.

Tseng P. Convergence of a Block Coordnate Descent Method for Nondifferentialble Minimization. Journal of Optimization Theory and Applications. 2001; 109:475–494.

Tseng P, Yun S. A Coordinate Gradient Descent Method for Nonsmooth Separable Minimization. Mathematical Programming B. 2009; 117:387–423.

United States Department of Commerce. City and County Data Book: 2007. 2007. https://www.census.gov/statab/www/ccdb.html

Wang H, Li R, Tsai C-L. Tuning Parameter Selectors for the Smoothly Clipped Absolute Deviation Method. Biometrika. 2007; 94:553–568. [PubMed: 19343105]

Wilkinson, JH. The Algebraic Eigenvalue Problem. New York: Oxford University Press; 1965.

Wu CFJ. On the Convergence Properties of the EM Algorithm. The Annals of Statistics. 1983; 11:95–103.

Wu T, Lange K. Coordinate Descent Algorithm for Lasso Penalized Regression. The Annals of Applied Statistics. 2008; 2:224–244.

Xu H. Algorithmic Construction of Efficient Fractional Factorial Designs With Large Run Sizes. Technometrics. 2009; 51:262–277.

Zangwill, WI. Nonlinear Programming: A Unified Approach. Englewood Cliffs, New Jersey: Prentice Hall; 1969.

Zhang C-H. Nearly Unbiased Variable Selection under Minimax Concave Penalty. The Annals of Statistics. 2010; 38:894–942.

Zou H, Hastie T. Regularization and Variable Selection via the Elastic Net. Journal of the Royal Statistical Society, Ser B. 2005; 67:301–320.

Zou H, Li R. One-step Sparse Estimates in Nonconcave Penalized Likelihood Models. The Annals of Statistics. 2008; 36:1509–1533. [PubMed: 19823597]

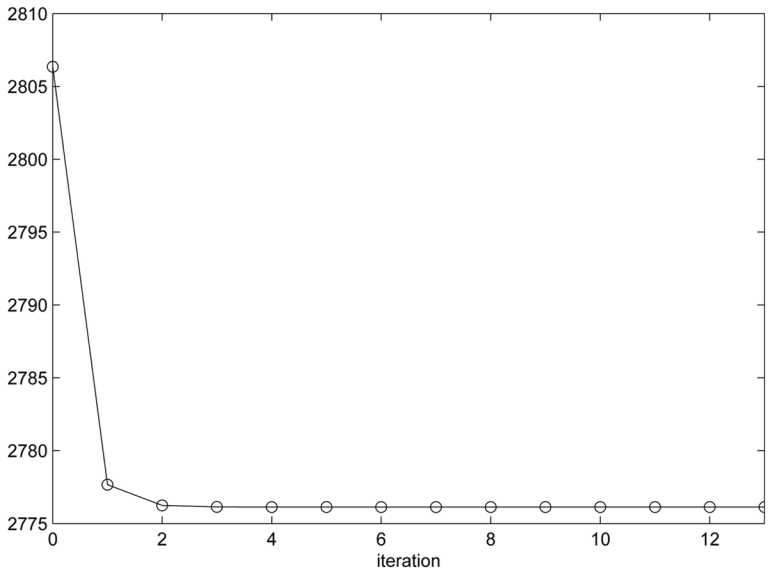

Figure 1.
Values of the objective function of an OEM sequence for the SCAD against iterations for Example 3.

Table 1

Average runtime (seconds) comparison between OEM and the SVD least squares method for $n > p$

n	p	OEM	SVD
	10	0.0433	0.0956
	50	0.2439	0.4098
50, 000	200	1.4156	4.9765
	1000	5.4165	45.3270
	5000	72.0630	442.3300

Table 2

Average runtime (seconds) comparison between OEM and glmnet for LASSO when $n \gg p$

p	n	OEM			glmnet		
		$\rho = 0$	$\rho = 0.2$	$\rho = 0.8$	$\rho = 0$	$\rho = 0.2$	$\rho = 0.8$
200	1×10^4	0.06	0.06	0.09	0.25	0.25	0.25
	1×10^5	0.46	0.46	0.49	2.81	2.81	2.80
	1×10^6	4.49	4.50	4.57	28.21	28.33	28.52
	1×10^7	45.86	45.77	46.34	325.25	325.57	330.37
1, 000	5×10^3	1.52	1.58	2.99	1.95	2.05	2.21
	1×10^4	1.61	1.60	2.72	3.92	4.01	4.57
	1×10^5	9.63	9.63	10.48	38.23	39.55	46.17
	1×10^6	92.97	92.75	93.38	382.21	386.71	461.81

Table 3

Average runtime (seconds) comparison between OEM and CD implemented in ncvreg and sparsenet for MCP when n is larger than p

p	n	OEM			CD			sparsenet		
		$\rho = 0$	$\rho = 0.2$	$\rho = 0.8$	$\rho = 0$	$\rho = 0.2$	$\rho = 0.8$	$\rho = 0$	$\rho = 0.2$	$\rho = 0.8$
20	400	0.0015	0.0014	0.0024	0.0219	0.0225	0.0245	0.0054	0.0052	0.0054
	1000	0.0013	0.0014	0.0025	0.0539	0.0497	0.0539	0.0084	0.0085	0.0092
	2000	0.0013	0.0014	0.0025	0.0671	0.0929	0.1139	0.0143	0.0144	0.0165
50	1000	0.0033	0.0034	0.0082	0.1522	0.1548	0.1840	0.0178	0.0175	0.0201
	2500	0.0035	0.0037	0.0073	0.3397	0.3267	0.4330	0.0381	0.0382	0.0467
	5000	0.0043	0.0046	0.0083	0.6245	0.6328	0.9002	0.0748	0.0731	0.0888
100	2000	0.0099	0.0108	0.0229	0.8130	0.8037	0.9536	0.0629	0.0622	0.067
	5000	0.0124	0.0126	0.0232	1.7770	1.9665	2.4915	0.1437	0.1424	0.1594
	10000	0.0180	0.0181	0.0303	3.6344	3.8181	4.7047	0.2867	0.2754	0.3136

Table 4

Average runtime (seconds) comparison between OEM lasso and OEM lasso, SCAD, MCP

p	*n*	OEM Lasso	OEM Lasso, SCAD, MCP
200	1×10^4	0.0994	0.1552
	1×10^5	0.5646	0.6094
	1×10^6	5.6788	5.6909
	5×10^6	28.1284	28.2814

Table 5

Average runtime comparison (seconds) of OEM for various large-scale penalized and unpenalized least squares problems. All penalized models are fit for 100 values of the tuning parameter. Matrix Sparsity = 0 corresponds to a matrix with all nonzero values.

Matrix Sparsity	p	n	OLS	Lasso	MCP	Lasso, SCAD, MCP
0	100	1×10^8	140.140	140.141	140.140	140.143
		1×10^9	1, 394.577	1, 394.578	1, 394.578	1, 394.580
		1×10^{10}	14, 267.520	14, 267.520	14, 267.520	14, 267.520
0	1, 000	1×10^7	946.168	946.266	946.265	946.463
		5×10^7	4, 625.954	4, 626.046	4, 626.045	4, 626.232
		1×10^8	9, 231.134	9, 231.223	9, 231.223	9, 231.405
0.999	10, 000	1×10^8	676.078	708.447	703.409	776.048
		5×10^8	3, 009.860	3, 047.521	3, 034.636	3, 098.680
		1×10^9	6, 119.103	6, 150.690	6, 144.003	6, 203.088
0.999	25, 000	1×10^8	2, 817.139	10, 584.220	3, 322.415	11, 924.730
		5×10^8	21, 630.780	22, 081.390	21, 866.760	22, 614.350
		1×10^9	40, 605.580	40, 952.200	40, 812.850	41, 382.630
0.995	10, 000	1×10^8	4, 221.646	4, 230.995	4, 321.926	4, 350.699
		5×10^8	20, 737.900	20, 732.570	20, 772.450	20, 789.470
		1×10^9	43, 202.950	43, 350.960	43, 313.470	43, 406.450

Table 6

Lasso, SCAD and MCP results for the U.S. Census Bureau data

Penalty	Final Model				Runtime (s)	
	Size	CV error	AIC	BIC	OEM	CD
LASSO	32	46.93	3.81	3.87	2.097	0.273
SCAD	28	47.12	3.81	3.87	1.783	3.454
MCP	23	47.17	3.82	3.88	1.433	3.032