Statistics and Its Interface Volume 6 (2013) 511–518

Coordinate great circle descent algorithm with application to single-index models

Peng Zeng* and Yichao Wu†,‡

Coordinate descent algorithm has been widely used to solve high dimensional optimization problems with a non-differentiable objective function recently. To provide theoretical justification, Tseng (2001) showed that it leads to a stationary point when the non-differentiable part of the objective function is separable. Motivated by the single index model, we consider optimization problems with a unit-norm constraint in this article. Because of this unit-norm constraint, the coordinate descent algorithm cannot be applied. In addition, non-separability of the non-differentiable part of the objective function makes the result of Tseng (2001) not directly applicable. In this paper, we propose a novel coordinate great circle descent algorithm to solve this family of optimization problems. The validity of the algorithm is justified both theoretically and via simulation studies. We also use the Boston housing data to illustrate this algorithm by applying it to fit single-index models.

AMS 2000 subject classifications: Primary 62H12, 62F10; secondary 62P05.

Keywords and phrases: Constrained optimization, Coordinate descent algorithm, Penalization, Single-index model, Unit-norm constraint.

1. INTRODUCTION

Due to the recent advance of data acquisition and storage, statisticians have been challenged by data with a large number of variables. The demand to analyze such data has motivated the fast-growing area of variable selection. Many methods have been developed for variable selection in the regularization framework. Examples include the LASSO (Tibshirani, 1996), the SCAD (Fan and Li, 2001), and the adaptive lasso (Zhang and Lu, 2007; Zou, 2006) among many others. A great deal of theoretical study and algorithmic development have been devoted to these methods. See Fan and Lv (2010) for a selective overview.

Many of the aforementioned methods can be formulated in the regularization framework

$$\min_{\boldsymbol{\beta}} Q(\boldsymbol{\beta}) = \ell(\boldsymbol{\beta}) + \sum_{j=1}^{p} p(|\beta_j|;\lambda_j), \tag{1}$$

where $\ell(\cdot)$ denotes the loss function such as the negative log-likelihood and $p(\cdot;\lambda)$ denotes some penalty function with tuning parameter $\lambda \geq 0$. There exist many algorithms for (1) depending on the choices of loss and penalty functions. In general, these algorithms work great for problems with a moderate number of parameters. Yet computational efficiency becomes an issue when the dimensionality is high. To improve efficiency, Fu (1998) proposed the coordinate descent algorithm which cycles through different components of the parameter vector to be optimized and updates one component each time. It was demonstrated to be very efficient since each updating involves only a marginal univariate optimization problem. This algorithm was later investigated by Daubechies et al. (2004), Friedman et al. (2007), Wu and Lange (2008), and others.

Towards a theoretical understanding, Tseng (2001) showed that the (block) coordinate descent algorithm can guarantee to find a stationary point when the non-differentiable part of the objective function is separable. Yet this separability assumption usually cannot be satisfied by constrained optimization problems, particularly those with equality constraints since these equality constraints intrinsically introduce certain relationships among parameters and thus make the objective function nonseparable. A typical example is the single-index model (Härdle and Stoker, 1989), which assumes that $E(Y|\boldsymbol{X}) = m(\boldsymbol{X}^T\boldsymbol{\theta})$ for some unknown link function $m(\cdot)$ and index direction $\boldsymbol{\theta}$. The single index is widely used to analyze financial and economic data. Examples are Powell et al. (1989), Ichimura (1993), Xia et al. (2002) and many others. For the purpose of identifiability, it is commonly assumed that the index direction has a unit norm, namely $\boldsymbol{\theta}^T\boldsymbol{\theta} = 1$. Consequently the domain of $\boldsymbol{\theta}$ is the unit sphere, and the components of $\boldsymbol{\theta}$ cannot change freely. Due to this unit-norm constraint, the coordinate descent algorithm is not directly applicable to the estimation of $\boldsymbol{\theta}$ in the single-index model. Motivated by the single-index model, we propose a coordinate great circle descent algorithm, which targets at optimization problems with a unit-norm constraint. Under some mild conditions, we show that

*Partially supported by NSF grant DMS-1107029.

†Partially supported by NSF grant DMS-1055210 and NIH/NCI grant R01 CA-149569.

‡Corresponding author.

the coordinate great circle descent algorithm converges to a stationary point.

The rest of the article is organized as follows. Section 2 presents the coordinate great circle descent algorithm. Its convergence analysis is provided in Section 3. Some implementation issues are discussed in Section 4 for the special case of a quadratic loss. Section 5 demonstrates our new algorithms using some simulation examples and one real data example. We conclude with Section 6.

2. COORDINATE GREAT CIRCLE DESCENT ALGORITHM

Consider the following optimization problem

$$\min_{\boldsymbol{\beta}} \quad Q(\boldsymbol{\beta}) = \ell(\boldsymbol{\beta}) + \sum_{j=1}^{p} p(|\beta_j|;\lambda_j) \qquad \text{subject to} \quad \boldsymbol{\beta}^T\boldsymbol{\beta} = 1, \tag{2}$$

where $\ell(\cdot)$ is a smooth loss function and $p(\cdot;\lambda)$ is a penalty function defined on $[0,\infty)$ depending on a nonnegative tuning parameter λ. The coordinate descent algorithm does not apply directly to (2) due to the unit-norm constraint. In the following, we propose an extension of the coordinate descent algorithm to solve (2) and prove that the new algorithm converges to a stationary point under some mild conditions.

Notice that the coordinate descent algorithm is essentially a special case of line search. At each iteration, it updates the solution by minimizing the objective function along a carefully chosen direction. Here, we want to adopt this general idea, and search along curves on which all points satisfy the unit-norm constraint. A good choice of such curves are great circles on the unit sphere. Denote $\boldsymbol{e}_j$ to be the $p \times 1$ vector with one in its jth component and zero otherwise. When $\boldsymbol{\beta} \neq \pm\boldsymbol{e}_j$, the great circle passing $\boldsymbol{\beta}$ and $\boldsymbol{e}_j$ is given by

$$\mathcal{C}(\boldsymbol{\beta},\boldsymbol{e}_j) = \{c\boldsymbol{e}_j \pm \sqrt{1-c^2}\boldsymbol{\beta}_{(-j)} : -1 \le c \le 1\},$$

where $\boldsymbol{\beta}_{(-j)} = (\boldsymbol{\beta} - \beta_j\boldsymbol{e}_j)/\sqrt{\sum_{k\neq j}\beta_k^2}$, which is obtained by setting the jth component of $\boldsymbol{\beta}$ as zero and then rescaling it to unit length. It is easy to verify that all points in $\mathcal{C}(\boldsymbol{\beta},\boldsymbol{e}_j)$ satisfy the unit-norm constraint.

Suppose that the current solution is $\tilde{\boldsymbol{\beta}} = (\tilde{\beta}_1,\ldots,\tilde{\beta}_p)^T$. We minimize the objective function over the great circle $\mathcal{C}(\tilde{\boldsymbol{\beta}},\boldsymbol{e}_j)$ and update the current solution by the corresponding optimizer. This step is repeated by cycling through $j = 1,2,\ldots,p$ until two consecutive solutions are close enough. Note that while searching over the great circle, the sub-optimization problem is univariate. It is in general very easy to solve these univariate sub-optimization problems even though an explicit solution may not be available. Note that $\mathcal{C}(\tilde{\boldsymbol{\beta}},\boldsymbol{e}_j)$ is well defined only when $\tilde{\boldsymbol{\beta}} \neq \pm\boldsymbol{e}_j$. If $\tilde{\boldsymbol{\beta}} = \pm\boldsymbol{e}_j$, the updating at j may be skipped and we continue to the next one.

To start the coordinate great circle descent algorithm, we need to choose an initial solution. There are many ways to choose an appropriate initial value for $\boldsymbol{\beta}$. For example, we may randomly choose one point on the unit sphere, or rescale the minimizer of the corresponding unconstrained problem to unit length.

3. CONVERGENCE ANALYSIS

Next we study properties of the proposed coordinate great circle descent algorithm in terms of its convergence. A point, say $\boldsymbol{\beta}^*$, is called a coordinate minimum point of $Q(\cdot)$ if it satisfies

$$Q(\boldsymbol{\beta}^*) \le Q(\boldsymbol{\beta}) \quad \text{for any } \boldsymbol{\beta} \in \mathcal{C}(\boldsymbol{\beta}^*,\boldsymbol{e}_j) \quad \text{if } \boldsymbol{\beta}^* \neq \pm\boldsymbol{e}_j \quad \text{for } j = 1,\ldots,p. \tag{3}$$

The following theorem asserts that the great circle coordinate descent algorithm converges to a coordinate minimum point under mild conditions.

Theorem 1. *If $Q(\boldsymbol{\beta})$ has at most one minimum over $\mathcal{C}(\boldsymbol{\beta},\boldsymbol{e}_j)$, namely the great circle passing $\boldsymbol{\beta}$ and $\boldsymbol{e}_j$, for any $j = 1,2,\ldots,p$ and any $\boldsymbol{\beta} \neq \pm\boldsymbol{e}_j$, then the great circle coordinate descent algorithm converges to a coordinate minimum point of $Q(\boldsymbol{\beta})$.*

Proof of Theorem 1. The proof is a straight forward extension of the proof for the convergence of the coordinate descent algorithm (Luenberger and Ye, 2008, p. 253). Note that the algorithm map of the great circle descent algorithm is the following composition of $2p$ maps

$$SC_pSC_{p-1}\cdots SC_1,$$

where $C_j(\boldsymbol{\beta}) = \mathcal{C}(\boldsymbol{\beta},\boldsymbol{e}_j)$ and S denotes the great circle search algorithm with the convention that $SC_j(\boldsymbol{e}_j) = \boldsymbol{e}_j$. It is obvious that map C_j is continuous and S is closed since the search is over the great circle which is closed and bounded. Consequently the above algorithm map is closed. Then the convergence of the coordinate great circle descent algorithms follows directly as a search over the great circle along any coordinate direction either decreases the objective function or it cannot change the current solution due to the uniqueness assumption. This completes the proof. □

Because the domain is the unit sphere instead of a Euclidean space, we need to modify the calculus of Euclidean space to make it appropriate for this particular manifold. Define the first-order spherical directional derivative of $f(\boldsymbol{\beta})$ at a point $\boldsymbol{\beta}$ on the unit sphere along a direction $\boldsymbol{\alpha}$ (of length one) by

$$f_s'(\boldsymbol{\beta};\boldsymbol{\alpha}) = \lim_{\delta\downarrow 0}\frac{1}{\delta}\Big\{f\Big(\frac{\boldsymbol{\beta}+\delta\boldsymbol{\alpha}}{\|\boldsymbol{\beta}+\delta\boldsymbol{\alpha}\|}\Big) - f(\boldsymbol{\beta})\Big\},$$

where the subscript s means "spherical" and $\|\cdot\|$ is the usual Euclidean norm of a vector. It is a direct generalization of the directional derivative in a Euclidean space with a restriction that points remain on the unit sphere when approaching $\boldsymbol{\beta}$. If $f(\boldsymbol{\beta})$ is differentiable, then the spherical directional derivative can be calculated by

$$f'_s(\boldsymbol{\beta};\boldsymbol{\alpha}) = \boldsymbol{\alpha}^T(\boldsymbol{I}-\boldsymbol{\beta}\boldsymbol{\beta}^T)\frac{\partial f(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}},$$

where $\partial f(\boldsymbol{\beta})/\partial\boldsymbol{\beta}$ is the gradient of f at $\boldsymbol{\beta}$ and $\boldsymbol{I}$ is the identity matrix. This spherical directional derivative can be understood as follows. The gradient of f can be decomposed as the sum of two components: one is orthogonal to $\boldsymbol{\beta}$ and the other parallel to $\boldsymbol{\beta}$. The former is exactly the projection of the gradient onto the linear space tangent to the unit sphere at $\boldsymbol{\beta}$, which is the rate of change of f with the unit-norm constraint. The latter characterizes the tendency of moving away from the unit sphere and hence does not contribute to $f'_s(\boldsymbol{\beta};\boldsymbol{\alpha})$.

Note that in terms of directional derivative, a coordinate minimum point $\boldsymbol{\beta}^*$ satisfies

(4)
$$Q'_s(\boldsymbol{\beta}^*;\boldsymbol{e}_j)\ge 0 \text{ and } Q'_s(\boldsymbol{\beta}^*;-\boldsymbol{e}_j)\ge 0 \quad \text{for } j=1,2,\dots,p$$

whenever these spherical directional derivatives exist. Next we prove that as long as the loss function part of the objective function is smooth enough, the directional derivative along any direction is nonnegative at the coordinate minimum point.

We call $\boldsymbol{\beta}^*$ a stationary point of $Q(\cdot)$ if it satisfies

$$Q'_s(\boldsymbol{\beta}^*,\boldsymbol{\alpha})\ge 0 \tag{5}$$

for any $p\times 1$ vector $\boldsymbol{\alpha}$ satisfying $\|\boldsymbol{\alpha}\|=1$.

Theorem 2. *If $\ell(\cdot)$ is differentiable and $p(\cdot;\lambda)$ is differentiable on $(0,\infty)$, and $p(\cdot;\lambda)$ is right-differentiable at 0, every coordinate minimum point of the coordinate great circle descent algorithm for (2) is a stationary point.*

Proof. We work on the loss function part (ℓ) and penalty part function (p) separately. For the loss function,

$$\ell'_s(\boldsymbol{\beta};\boldsymbol{\alpha}) = \boldsymbol{\alpha}^T(\boldsymbol{I}_p-\boldsymbol{\beta}\boldsymbol{\beta}^T)\frac{\partial \ell(\boldsymbol{\beta})}{\partial \boldsymbol{\beta}}, \tag{6}$$

where $\partial\ell(\boldsymbol{\beta})/\partial\boldsymbol{\beta}$ is the gradient of ℓ.

For the penalty function part, we have if $\beta_k\ne 0$

$$\lim_{\delta\to 0^+}\delta^{-1}\left\{p\left(\frac{|\beta_k+\delta\alpha_k|}{\|\ \boldsymbol{\beta}+\delta\boldsymbol{\alpha}\ \|};\lambda_k\right)-p(|\beta_k|;\lambda_k)\right\}$$
$$= \operatorname{sign}(\beta_k)(\alpha_k-\boldsymbol{\beta}^T\boldsymbol{\alpha}\beta_k)p'(|\beta_k|;\lambda_k),$$

where $\operatorname{sign}(\cdot)$ is the sign function and $p'(\cdot;\lambda_k)$ is the derivative of $p(\cdot;\lambda_k)$, and if $\beta_k=0$,

$$\lim_{\delta\to 0^+}\delta^{-1}\left\{p\left(\frac{|\beta_k+\delta\alpha_k|}{\|\ \boldsymbol{\beta}+\delta\boldsymbol{\alpha}\ \|};\lambda_k\right)-p(|\beta_k|;\lambda_k)\right\}$$
$$= |\alpha_k|p'(0^+;\lambda_k),$$

where $p'(0^+;\lambda_k)$ is the right-derivative of p at 0. Consequently we have

$$\lim_{\delta\to 0^+}\delta^{-1}\sum_{k=1}^{p}\left\{p\left(\frac{|\beta_k+\delta\alpha_k|}{\|\ \boldsymbol{\beta}+\delta\boldsymbol{\alpha}\ \|};\lambda_k\right)-p(|\beta_k|;\lambda_k)\right\}$$
$$= \boldsymbol{\alpha}^T(\boldsymbol{I}-\boldsymbol{\beta}\boldsymbol{\beta}^T)\boldsymbol{d}+\sum_{k:\beta_k=0}|\alpha_k|p'(0^+;\lambda_k),$$

where $\boldsymbol{d}$ is a $p\times 1$ vector whose kth element is $I_{(\beta_k\ne 0)}\operatorname{sign}(\beta_k)p'(|\beta_k|;\lambda_k)$. Here I_A denotes the indicator function such that $I_A=1$ if A is true and 0 otherwise.

As the coordinate great circle descent algorithm converges to a coordinate minimum point $\boldsymbol{\beta}$, we have

$$Q'_s(\boldsymbol{\beta};\boldsymbol{e}_j)\ge 0 \text{ and } Q'_s(\boldsymbol{\beta};-\boldsymbol{e}_j)\ge 0 \tag{7}$$

for every $j=1,2,\dots,p$. It further implies that

$$\begin{aligned} D_j^+ &= \boldsymbol{e}_j^T(\boldsymbol{I}-\boldsymbol{\beta}\boldsymbol{\beta}^T)\frac{\partial}{\partial\boldsymbol{\beta}}\ell(\boldsymbol{\beta})+\boldsymbol{e}_j^T(\boldsymbol{I}-\boldsymbol{\beta}\boldsymbol{\beta}^T)\boldsymbol{d}\\ &\quad+\sum_{k:\beta_k=0}I_{(k=j)}p'(0^+;\lambda_k)\\ &\ge 0\end{aligned}$$

and

$$\begin{aligned} D_j^- &= -\boldsymbol{e}_j^T(\boldsymbol{I}-\boldsymbol{\beta}\boldsymbol{\beta}^T)\frac{\partial}{\partial\boldsymbol{\beta}}\ell(\boldsymbol{\beta})-\boldsymbol{e}_j^T(\boldsymbol{I}-\boldsymbol{\beta}\boldsymbol{\beta}^T)\boldsymbol{d}\\ &\quad+\sum_{k:\beta_k=0}I_{(k=j)}p'(0^+;\lambda_k)\\ &\ge 0\end{aligned}$$

for $j=1,2,\dots,p$. Consequently we have

$$\begin{aligned} Q'_s(\boldsymbol{\beta};\alpha) &= \boldsymbol{\alpha}^T(\boldsymbol{I}-\boldsymbol{\beta}\boldsymbol{\beta}^T)\frac{\partial}{\partial\boldsymbol{\beta}}\ell(\boldsymbol{\beta})+\boldsymbol{\alpha}^T(\boldsymbol{I}-\boldsymbol{\beta}\boldsymbol{\beta}^T)\boldsymbol{d}\\ &\quad+\sum_{k:\beta_k=0}|\alpha_k|p'(0^+;\lambda_k)\\ &=(\sum_{j:\alpha_j>0}\alpha_jD_j^+)+(\sum_{j:\alpha_j<0}(-\alpha_j)D_j^-)\ge 0\end{aligned}$$

for any $\boldsymbol{\alpha}$ satisfying $\|\boldsymbol{\alpha}\|=1$. This implies that $\boldsymbol{\beta}$ is a stationary point. □

4. IMPLEMENTATION ISSUES FOR QUADRATIC LOSS FUNCTION

In general, at each step of the coordinate great circle descent algorithm, the search over the coordinate great circle does not have a closed-form solution. However if the loss function is quadratic, a closed-form solution is possible. In this section, we consider an important special case of (2) with a quadratic loss function and a weighted lasso penalty function. In this case, the optimization problem (2) can be rewritten as

$$(8) \qquad \min_{\boldsymbol{\beta}} \quad Q(\boldsymbol{\beta}) = \frac{1}{2}\boldsymbol{\beta}^T \boldsymbol{S}\boldsymbol{\beta} - \boldsymbol{r}^T\boldsymbol{\beta} + \sum_{k=1}^{p} \lambda_k |\beta_k|$$
$$\text{subject to} \quad \boldsymbol{\beta}^T\boldsymbol{\beta} = 1,$$

where $\boldsymbol{S}$ is a $p \times p$ nonnegative definite matrix, $\boldsymbol{r}$ is a $p \times 1$ vector, $\boldsymbol{\beta}$ is a $p \times 1$ vector of parameters, and $\lambda_k \geq 0$ are tuning parameters.

Suppose that $\boldsymbol{\beta}^*$ is a vector on the unit sphere, and we derive the algorithm to find the vector $\boldsymbol{\beta}$ that minimizes $Q(\boldsymbol{\beta})$ along the great circle $\mathcal{C}(\boldsymbol{\beta}^*, \boldsymbol{e}_j)$. Notice that any vector in $\mathcal{C}(\boldsymbol{\beta}^*, \boldsymbol{e}_j)$ can be expressed as

$$\boldsymbol{\beta} = c\boldsymbol{e}_j + \sqrt{1-c^2}\boldsymbol{\beta}_{(-j)} \text{ or } \boldsymbol{\beta} = c\boldsymbol{e}_j - \sqrt{1-c^2}\boldsymbol{\beta}_{(-j)}$$

for $c \in [-1, 1]$. Hence the objective function can be written as

$$\begin{aligned} Q_j(c) = &\frac{1}{2}c^2(\boldsymbol{e}_j^T\boldsymbol{S}\boldsymbol{e}_j - \boldsymbol{\beta}_{(-j)}^T\boldsymbol{S}\boldsymbol{\beta}_{(-j)}) \\ &\pm c\sqrt{1-c^2}\boldsymbol{e}_j^T\boldsymbol{S}\boldsymbol{\beta}_{(-j)} + c(\lambda \mathrm{sign}(c) - \boldsymbol{e}_j^T\boldsymbol{r}) \\ &+ \sqrt{1-c^2}(\sum_{k\neq j}\lambda_k|\beta_{(-j),k}| \\ &\mp \boldsymbol{\beta}_{(-j)}^T\boldsymbol{r}) + \frac{1}{2}\boldsymbol{\beta}_{(-j)}^T\boldsymbol{S}\boldsymbol{\beta}_{(-j)}, \end{aligned}$$

where $\beta_{(-j),k}$ denotes the kth element of $\boldsymbol{\beta}_{(-j)}$. The function $Q_j(c)$ is differentiable on $(-1, 0) \cup (0, 1)$ and the only non-differentiable point is $c = 0$. Therefore, the minimum of $Q_j(c)$ is achieved either at the end points $c = \pm 1$, at the non-differentiable point $c = 0$, or at a stationary point satisfying $Q_j'(c) = 0$, where $Q_j'(c)$ is the first order derivative when $c \in (-1, 0)$ or $c \in (0, 1)$. Note that $Q_j'(c)$ has different forms for $c \in (-1, 0)$ and $c \in (0, 1)$.

Let us discuss how to find the stationary point of $Q_j(c)$ satisfying $Q_j'(c) = 0$. For ease of presentation, we consider a generic form of $Q_j(c)$. Denote

$$g(x) = ax^2 + bx\sqrt{1-x^2} + cx + d\sqrt{1-x^2}$$

for $x \in (-1, 0)$ or $x \in (0, 1)$. It is clear that $Q_j(c)$ exactly has the form of $g(c)$. The first-order derivative of $g(x)$ is

$$g'(x) = 2ax + 2b\sqrt{1-x^2} - \frac{dx+b}{\sqrt{1-x^2}} + c.$$

Set $g'(x) = 0$ and we obtain

$$(9) \qquad (2ax + c)\sqrt{1-x^2} = 2bx^2 + dx - b.$$

Squaring the above equation on both side yields a fourth-order polynomial

$$\begin{aligned} (10) \quad &4(b^2 + a^2)x^4 + 4(bd + ac)x^3 + (d^2 - 4b^2 - 4a^2 + c^2)x^2 \\ &-(4ac + 2bd)x + (b^2 - c^2) = 0. \end{aligned}$$

There exists an explicit formula for the roots of a fourth-order polynomial. Equation (10) has up to four real roots, but some roots of (10) may not be the roots of (9), because (10) also implies $(2ax + c)\sqrt{1-x^2} = -(2bx^2 + dx - b)$. We need to check whether a root of (10) is indeed a root of (9).

The stationary point of $g(x)$ can also be a maximizer or a saddle point instead of a minimizer. Hence we need to check the second-order derivative to make sure it is a minimizer. Some calculation yields the second-order derivative of $g(x)$,

$$g''(x) = 2a - \frac{2bx}{\sqrt{1-x^2}} - \frac{bx+d}{(1-x^2)^{3/2}}.$$

A sufficient condition for the stationary point being a minimizer is $g''(x) > 0$. In practice, we can simply compare the values of the objective function at $c = -1, 0, 1$ and all stationary points.

Recall that Theorem 1 and Theorem 2 only guarantee that the proposed coordinate great circle descent algorithm converges to a stationary point. It is unclear whether this stationary point is a local minimizer. Next we use a simulation example to check this issue.

Example 1. Let $p = 10$. We first randomly generate a 100×10 matrix $\mathbb{X}$, whose entries are independently simulated from uniform$(-2, 2)$, and a 10×1 vector $\boldsymbol{b}$, whose components are independently sampled from uniform$(-2, 2)$. Then calculate $\boldsymbol{y} = \mathbb{X}\boldsymbol{\beta} + 0.5\boldsymbol{\varepsilon}$, where $\boldsymbol{\varepsilon}$ is a 100×1 vector whose components are independently sampled from $\mathrm{N}(0, 1)$. Then we calculate $\boldsymbol{S} = \mathbb{X}^T\boldsymbol{J}\mathbb{X}$ and $\boldsymbol{r} = \mathbb{X}^T\boldsymbol{J}\boldsymbol{y}$, where $\boldsymbol{J} = \boldsymbol{I} - n^{-1}\boldsymbol{1}\boldsymbol{1}^T$, where $\boldsymbol{1}$ denotes a vector of ones. We also randomly generate λ from uniform$(0, 200)$. The above procedure is repeated for 1,000 times and we obtain $\{(\boldsymbol{S}_i, \boldsymbol{r}_i, \lambda_i), i = 1, \ldots, 1{,}000\}$.

For each triplet $(\boldsymbol{S}_i, \boldsymbol{r}_i, \lambda_i)$, we apply the proposed coordinate great circle descent algorithm to find $\hat{\boldsymbol{\beta}}_i$ that minimizes

$$Q(\boldsymbol{\beta}) = \frac{1}{2}\boldsymbol{\beta}^T\boldsymbol{S}_i\boldsymbol{\beta} - \boldsymbol{r}_i^T\boldsymbol{\beta} + \lambda\sum_{k=1}^{p}|\beta_k|$$

subject to $\boldsymbol{\beta}^T\boldsymbol{\beta} = 1$. The initial value is set to be $(1, 0, \ldots, 0)^T$. We use the following procedure to check if $\hat{\boldsymbol{\beta}}_i$ is a local minimizer. Denote $\boldsymbol{\beta}^* = (\beta_1^*, \ldots, \beta_p^*)^T$, where β_k^* is obtained by rounding the kth component of $\hat{\boldsymbol{\beta}}_i$ to the nearest 0.01. We consider all the possible points in the set of

$$\begin{aligned} \mathcal{A} = \{&(\beta_1^* + 0.01 * n_1, \ldots, \beta_p^* + 0.01 * n_p)^T, \\ &n_1, \ldots, n_p = -3, -2, -1, 0, 1, 2, 3\}. \end{aligned}$$

For each point in $\mathcal{A}$, we first scale it to unit length and then calculate the corresponding objective function value and compare it with $Q(\hat{\boldsymbol{\beta}}_i)$. If $Q(\hat{\boldsymbol{\beta}}_i)$ is less than the objective function value for all points in $\mathcal{A}$, it is verified that $\hat{\boldsymbol{\beta}}_i$ is indeed a local minimizer.

In all 1,000 repetitions, $\hat{\boldsymbol{\beta}}_i$ is indeed a local minimizer. The quartiles of the number iteration cycles to reach the local minimizer is 2 (5%), 8 (25%), 9 (50%), and 9 (75%), 11 (95%). The algorithm converges in a small number of iterations on average.

The above simulation example provides some encouraging numerical evidence that the coordinate great circle descent algorithm can possibly identify a local minimizer although Theorem 1 and Theorem 2 only guarantee that the algorithm stops at a stationary point. In practice, to make sure the algorithm indeed stops at a local minimum, we can also check the second-order derivative. For the objective function $Q(\boldsymbol{\beta})$ in (8), simple calculation yields the spherical directional derivative as

$$Q'_s(\boldsymbol{\beta};\boldsymbol{d}) = \boldsymbol{d}^T(\boldsymbol{I}-\boldsymbol{\beta}\boldsymbol{\beta}^T)(\boldsymbol{S}\boldsymbol{\beta}-\boldsymbol{r}+\boldsymbol{s}),$$

where $\boldsymbol{s}=(s_1,\ldots,s_p)^T$ and

$$s_k = \begin{cases}\operatorname{sign}(\beta_k)\,\lambda_k, & \beta_k \neq 0,\\ \operatorname{sign}(d_k)\,\lambda_k, & \beta_k = 0.\end{cases}$$

The second-order spherical directional derivative is the spherical directional derivative of the first-order spherical directional derivative, namely

$$Q''_s(\boldsymbol{\beta};\boldsymbol{d}) = \boldsymbol{d}^T\boldsymbol{M}\boldsymbol{d},$$

where $\boldsymbol{M}=(\boldsymbol{N}+\boldsymbol{N}^T)/2$ with

$$\boldsymbol{N} = (\boldsymbol{I}-\boldsymbol{\beta}\boldsymbol{\beta}^T)[\boldsymbol{S}(\boldsymbol{I}-\boldsymbol{\beta}\boldsymbol{\beta}^T) - \boldsymbol{\beta}^T(\boldsymbol{S}\boldsymbol{\beta}-\boldsymbol{r}+\boldsymbol{s})\boldsymbol{I} \\ - (\boldsymbol{S}\boldsymbol{\beta}-\boldsymbol{r}+\boldsymbol{s})\boldsymbol{\beta}^T].$$

A sufficient condition to guarantee that $\boldsymbol{\beta}$ is a local minimizer is that $Q''_s(\boldsymbol{\beta};\boldsymbol{d})>0$ for any $\boldsymbol{d}$. Because $\boldsymbol{M}$ and $\boldsymbol{N}$ still depend on $\boldsymbol{d}$ via $\boldsymbol{s}$, it is difficult to verify this condition.

Define $\boldsymbol{s}'$ as $\boldsymbol{s}'=(s'_1,\ldots,s'_k)^T$, where

$$s'_k = \begin{cases}\operatorname{sign}(\beta_k)\,\lambda_k, & \beta_k \neq 0,\\ 0, & \beta_k = 0.\end{cases}$$

Similarly we can define $\boldsymbol{M}'$ and $\boldsymbol{N}'$ as $\boldsymbol{M}$ and $\boldsymbol{N}$ with $\boldsymbol{s}$ replaced by $\boldsymbol{s}'$. Simple calculation yields that $\boldsymbol{M}'=\boldsymbol{M}$ and $\boldsymbol{N}'=\boldsymbol{N}$. Additionally $\boldsymbol{M}'$ and $\boldsymbol{N}'$ do not depend on $\boldsymbol{d}$. Therefore, a sufficient condition to guarantee that $\boldsymbol{\beta}$ is a local minimizer is that $\boldsymbol{M}'$ is positive definite. Although it may be difficult to know when $\boldsymbol{M}'$ is positive definite for general $\boldsymbol{S}$, $\boldsymbol{r}$ and λ, we can always verify whether $\boldsymbol{M}'$ is positive definite when $\boldsymbol{S}$, $\boldsymbol{r}$, and λ are given.

5. APPLICATION TO SINGLE INDEX MODEL

As aforementioned in the introduction, a single-index model assumes that $E(Y|\boldsymbol{X}) = m(\boldsymbol{X}^T\boldsymbol{\theta})$ for unknown link function $m(\cdot)$ and index vector $\boldsymbol{\theta}$ with a unit norm. Many methods have been proposed for fitting single-index models, for example Härdle and Stoker (1989) and Xia (2006). Recently, Zeng et al. (2012) proposed a method, called sim-lasso, for simultaneous estimation and variable selection by combining local linear smoothing with a lasso-type penalty function. Based on a dataset $\{(\boldsymbol{x}_i,y_i), i=1,2,\ldots,n\}$, the estimate $\boldsymbol{\theta}$ is obtained from the following minimization problem

$$(11)\quad \min_{a_j,b_j,\boldsymbol{\theta}} \quad \sum_{j=1}^n\sum_{i=1}^n (y_i - a_j - b_j\boldsymbol{\theta}^T(\boldsymbol{x}_i-\boldsymbol{x}_j))^2 w_{ij} \\ +\lambda\sum_{j=1}^n |b_j| \sum_{k=1}^p |\theta_k| \\ \text{subject to} \quad \boldsymbol{\theta}^T\boldsymbol{\theta}=1,$$

where $w_{ij} = K_h(\boldsymbol{\theta}^T(\boldsymbol{x}_i-\boldsymbol{x}_j))/\sum_\ell K_h(\boldsymbol{\theta}^T(\boldsymbol{x}_\ell-\boldsymbol{x}_j))$ and $K_h(\cdot)$ is a kernel function with bandwidth h. The original algorithm proposed in Zeng et al. (2012) is to alternately update $\{a_j,b_j,j=1,\ldots,n\}$ and $\boldsymbol{\theta}$ until convergence. Because of the scaling invariance property, the unit-norm constraint is ignored when updating $\boldsymbol{\theta}$ from $\{a_j,b_j,j=1,\ldots,n\}$, and $\boldsymbol{\theta}$ is then scaled to unit length. Although it works, a potential pitfall is that when λ is large, the solution of $\boldsymbol{\theta}$ is 0 and in this case it is not possible to scale $\boldsymbol{\theta}$ to unit length.

Notice that when updating $\boldsymbol{\theta}$ for given $\{a_j,b_j,j=1,\ldots,n\}$, the problem (11) simplifies to a special case of (8). Therefore, we may apply the proposed coordinate great circle descent algorithm to update $\boldsymbol{\theta}$. With this adjustment, sim-lasso is free of pitfalls.

In implementation, we need the value of bandwidth h and tuning parameter λ in (11). The bandwidth h is selected by the rule-of-thumb $h_0 = s[4/(2p+1)n^{-1}]^{1/(p+4)}$ (Silverman, 1986), where s is ideally the standard deviation of $\boldsymbol{X}^T\boldsymbol{\theta}$. Because $\boldsymbol{\theta}$ is unknown, we approximate s by the median of the standard deviations of the components of $\boldsymbol{X}$. The tuning parameter λ is chosen using cross-validation based on the sum of squared prediction errors. The initial value of sim-lasso is selected using OPG-lasso. See Zeng et al. (2012) for detailed discussions.

We use a simulation example and a real data example to demonstrate the performance of sim-lasso coupled with the coordinate great circle descent algorithm.

5.1 Simulation examples

In this simulation example, we assume that the true underlying model is

$$Y = 4\sqrt{|\boldsymbol{\beta}^T\boldsymbol{X}+1|} + \boldsymbol{\beta}^T X + \varepsilon,$$

where $\boldsymbol{\beta}=(1,-1,2,-0.5,0,\ldots)^T$, the components of $\boldsymbol{X}$ are independently sampled from uniform$(-2,\,2)$, and ε is independent of $\boldsymbol{X}$ and is sampled from $N(0,1)$. It is clear that $\boldsymbol{\theta}=\boldsymbol{\beta}/\|\boldsymbol{\beta}\|$. The sample size is $n=100$ and the dimension of X varies from $p=10$ to $p=120$.

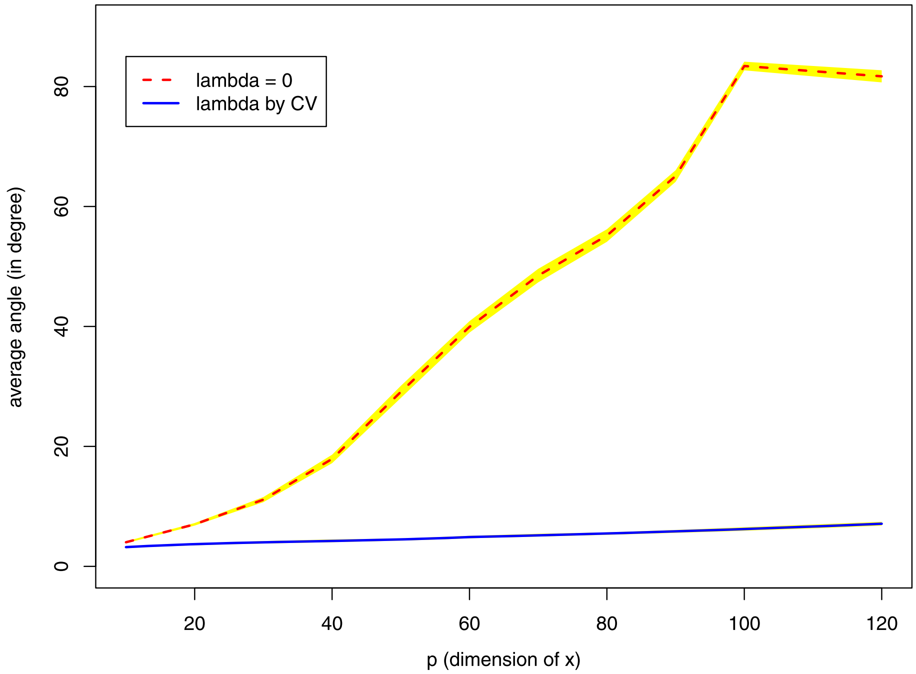

Figure 1. The average angle between estimates and truth.

To demonstrate the advantage of using the penalization approach, we compare the performance of sim-lasso for two possible choices of λ, namely, $\lambda = 0$ and λ chosen from 10-fold cross-validation. Note that $\lambda = 0$ means that there is no penalization added. For a fair comparison, the bandwidth and the initial value are the same for these two scenarios, where the bandwidth is chosen by rule-of-thumb and the initial value is by OPG-lasso. The performance of an estimate of $\boldsymbol{\theta}$ is measured by the angle (in degree) between this estimate and the true $\boldsymbol{\theta}$. Here a zero angle indicates a perfect estimate while 90 means that two directions are orthogonal.

Figure 1 shows how the average angle between estimates and true $\boldsymbol{\theta}$ changes as the dimension of $\boldsymbol{X}$ increases, based on 200 replicates. It is observed that the performance for $\lambda = 0$ deteriorates quickly as p increases. When $p \geq 100$, the performance levels off because it is close to the worst possibility of 90. As a contrast, the performance corresponding to the optimal λ deteriorate very slowly as p increases. It performs very well even when $p > n$.

5.2 A real data example

Consider the Boston housing data collected by Harrison and Rubinfeld (1978). The objective of this study is to understand which factors influence the median value of owner-occupied homes. The dataset contains 506 observations. The response is the logarithm of the median value of owner-occupied homes in \$1,000's (y). There are 13 predictors, including crime rate (per capita by town; x_1), proportion of residential land zoned for lots over 25,000 square feet (x_2), proportion of non-retail business acres per town (x_3), Charles River dummy variable (=1 if tract bounds river and 0 otherwise; x_4), nitric oxides concentration (parts per 10 million; x_5), average number of rooms per dwelling (x_6), proportion of owner-occupied units built prior to 1940 (x_7), weighted distances to five Boston employment centers (x_8), index of accessibility to radial highways (x_9), full-value property-tax rate per \$10,000 (x_{10}), pupil-teacher ratio by town (x_{11}), proportion of blacks by town (transform to $1,000(B_k - 0.63)^2$; x_{12}) lower status of the population (x_{13}). The predictors are standardized to have zero mean and one standard deviation before analysis.

The bandwidth is $h = 0.6197$ using the rule-of-thumb. The solution path of the estimated index θ is displayed in Figure 2, where the vertical lines indicate the positions of optimal λ for estimation and for variable selection, respectively, as explained next. A 20-fold cross validation selection $\lambda = 0.0025$ for the optimal tuning parameter for estimation. Usually, this tuning parameter leads to overselection of the variables and a larger tuning parameter is preferred for variable selection purpose. An empirical rule is to use the largest λ within one standard deviation of the minimum cross-validation score; see Hastie et al. (2001) and Zeng et al. (2012). Based on this rule, the optimal tuning parameter for variable selection is $\lambda = 0.01$. The estimated value for $\boldsymbol{\theta}$ is $\hat{\boldsymbol{\theta}} = (-0.251, 0.000, 0.000, 0.075, -0.134, 0.237, 0.000, \text{-}0.193, 0.116, -0.114, -0.207, 0.159, -0.852)^T$. The scatter plot of y against $\hat{\theta}^T x$ is displayed in Figure 2. The predictor x_2, x_3, x_7 does not explain extra variability of house values after controlling other predictors. The low

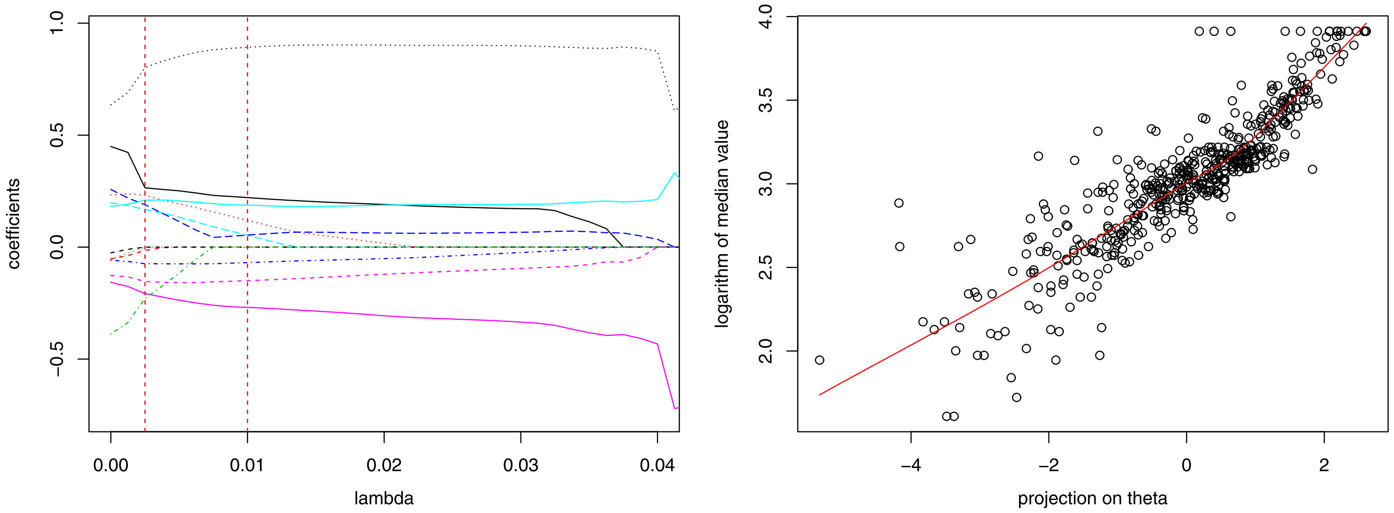

Figure 2. Solution path for the estimated θ and scatter plot of y against $\hat{\boldsymbol{\theta}}^T x$.

status of the population (x_{13}) is the leading factor that is associated with low house values. The average number of rooms per dwelling is the leading factor that is associated with high house values.

6. DISCUSSIONS

Another potential application of (8) is principal component analysis. The proposed coordinate great circle descent algorithm can be readily used to estimate a sparse first principal component direction. However, it is not clear how to extend it to estimate the remaining principal component directions, because the principal component directions are usually assumed to be orthogonal to each other. The orthogonality is essentially linear equality constraints, which cannot be accommodated by the current algorithm. Another possible application is the shape-constrained problem as in Cule et al. (2010). The major difficulty is that our theoretical results are not directly applicable to such cases. This will be a potential future research project.

ACKNOWLEDGEMENTS

We thank the reviewers and editors for helpful and constructive comments. This research was partially supported by NSF grant DMS-1107029 (Zeng), NIH/NCI grant R01 CA-149569 (Wu) and NSF grant DMS-1055210 (Wu).

Received 18 June 2013

REFERENCES

Cule, M. L., Samworth, R. J., and Stewart, M. I. (2010). Maximum likelihood estimation of a multi-dimensional log-concave density. *J. Roy. Statist. Soc., Ser. B. (with discussion)* **72**, 545–600. MR2758237

Daubechies, I., Defrise, M., and De Mol, C. (2004). An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Comm. Pure Appl. Math.* **57**, 1413–1457. MR2077704

Fan, J. and Li, R. (2001). Variable selection via nonconcave penalized likelihood and it oracle properties. *Journal of the American Statistical Association* **96**, 1348–1360. MR1946581

Fan, J. and Lv, J. (2010). A selective overview of variable selection in high dimensional feature space. *Statistica Sinica* **20**, 101–148. MR2640659

Friedman, J., Hastie, T., Höfling, H., and Tibshirani, R. (2007). Pathwise coordinate optimization. *Annals of Applied Statistics* **1**, 302–332. MR2415737

Fu, W. J. (1998). Penalized regressions: The bridge versus the lasso. *Journal of Computational and Graphical Statistics* **7**, 397–416. MR1646710

Härdle, W. and Stoker, T. M. (1989). Investigating smooth multiple regression by the method of average derivatives. *Journal of the American Statistical Association* **84**, 986–995. MR1134488

Harrison, D. and Rubinfeld, D. L. (1978). Hedonic housing prices and the demand for clean air. *Journal of Environmental Economics and Management* **5**, 81–102.

Hastie, T., Tibshirani, R., and Friedman, J. (2001). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction.* New York: Springer. MR1851606

Ichimura, H. (1993). Semiparametric least squares sls! and weighted sls estimation of single-index models. *Journal of Econometrics* **58**, 71–120. MR1230981

Luenberger, D. G. and Ye, Y. (2008). *Linear and Nonlinear Programming.* Springer. MR2423726

Powell, J. L., Stock, J. H., and Stoker, T. M. (1989). Semiparametric estimation of index coefficients. *Econometrica* **57**, 1403–1430. MR1035117

Silverman, B. W. (1986). *Density Estimation for Statistics and Data Analysis.* London: Chapman & Hall Ltd. MR0848134

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Series B* **58**, 267–288. MR1379242

Tseng, P. (2001). Convergence of a block coordinate descent method for nondifferentiable minimization. *Journal of Optimization Theory And Applications* **109**, 475–494. MR1835069

Wu, T. T. and Lange, K. (2008). Coordinate descent algorithm for lasso penalized regression. *The Annals of Applied Statistics 2*, 224–244. MR2415601

Xia, Y. (2006). Asymptotic distributions for two estimators of the single-index model. *Econometric Theory* **22**, 1112–1137. MR2328530

Xia, Y., Tong, H., and Li, W. K. (2002). Single-index volatility models and estimation. *Statistica Sinica* **12**, 785–799. MR1929964

Zeng, P., He, T. and Zhu, Y. (2012). A lasso-type approach for estimation and variable selection in single-index models. *Journal of Computational and Graphical Statistics* **21**, 92–109. MR2913358

Zhang, H. H. and Lu, W. (2007). Adaptive lasso for cox's proportional hazards model. *Biometrika* **94**, 691–703. MR2410017

Zou, H. (2006). The adaptive lasso and its oracle properties. *Journal of the American Statistical Association* **101**, 1418–1429. MR2279469

Peng Zeng
Department of Mathematics and Statistics
Auburn University
Auburn, AL 36849
USA
E-mail address: zengpen@auburn.edu

Yichao Wu
Department of Statistics
North Carolina State University
Raleigh, NC 27695
USA
E-mail address: wu@stat.ncsu.edu