Gradient LASSO for feature selection

Yongdai Kim YDKIM@STATS.SNU.AC.KR
Department of Statistics, Seoul National University, Seoul 151-742, Korea

Jinseog Kim JSKIM@STATS.SNU.AC.KR
Statistical Research Center for Complex Systems, Seoul National University, Seoul 151-742, Korea

Abstract

LASSO (Least Absolute Shrinkage and Selection Operator) is a useful tool to achieve the shrinkage and variable selection simultaneously. Since LASSO uses the L_1 penalty, the optimization should rely on the quadratic program (QP) or general non-linear program which is known to be computational intensive. In this paper, we propose a gradient descent algorithm for LASSO. Even though the final result is slightly less accurate, the proposed algorithm is computationally simpler than QP or non-linear program, and so can be applied to large size problems. We provide the convergence rate of the algorithm, and illustrate it with simulated models as well as real data sets.

1. Introduction

Tibshirani (1996) introduced an interesting method for shrinkage and variable selection, called "Least Absolute Shrinkage and Selection Operator (LASSO)". It achieves better prediction accuracy by shrinkage as the ridge regression, but at the same time, it gives a sparse solution, which means that some coefficients are exactly 0. Hence, LASSO is thought to achieve the shrinkage and variable selection simultaneously.

Knight and Fu (2000) proved some asymptotic results for LASSO type estimator. Chen et al. (1999) and Bakin (1999) applied the idea of LASSO to wavelet and developed a method called "basis pursuit". Gunn and Kandola (2002) applied LASSO to the kernel machine, and Zhang et al. (2003) applied to smoothing spline models.

Appearing in *Proceedings of the 21st International Conference on Machine Learning*, Banff, Canada, 2004.

One problem in LASSO is that the objective function is not differentiable, and hence special optimization techniques are necessary. Tibshirani (1996) used the quadratic program (QP) for least square regressions and the iteratively reweighted least square procedure with QP for generalized linear models. Osborne et al. (2000) proposed a faster QP algorithm for LASSO, which was implemented by Lokhorst et al. (1999) as `lasso2` package in R system. Recently, Efron et al. (2004) developed an algorithm closely related to Osborne et al. (2000)'s algorithm. The algorithms based on QP, however, may not be easily applicable to large data sets when the dimension of inputs is very large. One such example is the likelihood basis pursuit where the dimension of inputs is proportional to the sample size (see Section 6 for details). Moreover, the algorithms may not converge to the optimal solution when the loss function is other than the squares loss. Besides QP, Grandvalet and Canu (1999) implemented a fixed point algorithm using the equivalence between the adaptive ridge regression and LASSO, and Perkins et al. (2003) developed a stagewise gradient descent algorithm called *grafting*. These algorithms, however, may not lead global convergence.

In this paper, we propose a gradient descent algorithm for LASSO. The proposed algorithm is computationally simpler than QP or non-linear program, and so can be applicable to large size problems. We prove that the proposed algorithm converges to the optimum under regularity conditions. Also, we provide the convergence rate of the algorithm.

The proposed algorithm is based on the L_1 boosting algorithm proposed by Mason et al. (2000), which is a regularized boosting algorithm. While the boosting algorithms such as the AdaBoost (Freund & Schapire, 1997) and LogitBoost (Friedman, 2001) find the optimal linear combination of given weak learners, the L_1 boosting algorithm finds the optimal convex combination of weak learners. We utilize the fact that the optimization problem of the LASSO can be considered

as an extension of the L_1 boosting to develop the proposed algorithm.

The paper is organized as follows. In section 2, we explain the LASSO. In section 3, the proposed algorithm is presented, and its theoretical properties are studied in section 4. Results of the empirical studies are given in section 5, a real dataset is analyzed in section 6, and discussions follow in section 7.

2. LASSO

We first present the general LASSO setting and give the three examples. Let $(y_1, \mathbf{x}_1), \ldots, (y_n, \mathbf{x}_n)$ be n output/input pairs where $y_i \in \mathcal{Y}$ and $\mathbf{x}_i \in \mathcal{X}$. Here, $\mathcal{Y}$ and $\mathcal{X}$ are the domains of the output and input, respectively. We assume that $\mathcal{X} = \mathcal{X}_1 \otimes \mathcal{X}_2 \otimes \cdots \otimes \mathcal{X}_d$ and $\mathcal{X}_l$ are subsets of $R^{p_l}, l = 1, \ldots, d$. Also, we write $\mathbf{x}_i = (\mathbf{x}_{i1}, \ldots, \mathbf{x}_{id})$ where $\mathbf{x}_{il} \in \mathcal{X}_l$. Finally, we let $\boldsymbol{\beta} = (\boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_d)$ be the corresponding regression coefficients, where $\boldsymbol{\beta}_l \in R^{p_l}$.

For a given loss function l, the objective of LASSO is to find $\boldsymbol{\beta}$ which minimizes the (empirical) risk

$$R(\beta_0, \boldsymbol{\beta}) = \sum_{i=1}^{n} l\left(y_i, \beta_0 + \sum_{l=1}^{d} \mathbf{x}_{il}\boldsymbol{\beta}_l^{'}\right)$$

subject to $|\boldsymbol{\beta}_l|_1 \leq \lambda_l$ for $l = 1, \ldots, d$. Here $\lambda_l \geq 0$ and $|\boldsymbol{\beta}_l|_1 = \sum_{k=1}^{p_l} |\beta_{lk}|$.

Example 1. *Multivariate linear regression*

Let $(y_1, \mathbf{z}_1), \ldots, (y_n, \mathbf{z}_n)$ be n output/input pairs. A multivariate linear regression model is given by

$$y_i = \beta_0 + \beta_1 z_{i1} + \cdots + \beta_k z_{ik} + \epsilon_i$$

where ϵ_i are assumed to be a mean zero random quantities. The LASSO estimate $\hat{\beta}_0, \ldots, \hat{\beta}_k$ is the minimizer of

$$\sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{k} \beta_j z_{ij}\right)^2$$

subject to $\sum_{j=1}^{k} |\beta_j| \leq \lambda$. This problem can be embedded into the general setting of the LASSO given in the beginning of section 2 with $d = 1, p_1 = k$, $\mathbf{x}_{i1} = (z_{i1}, \ldots, z_{ik}), \boldsymbol{\beta}_1 = (\beta_1, \ldots, \beta_k)$ and $l(y, a) = (y - a)^2$.

Example 2. *Multivariate logistic regression*

A multivariate logistic regression model for two class classification problems is given by

$$\text{logit}\, \Pr(y_i = 1 | \mathbf{z}_i) = f(\mathbf{z}_i) \tag{1}$$

where

$$f(\mathbf{z}_i) = \beta_0 + \beta_1 z_{i1} + \cdots + \beta_k z_{ik}.$$

Here, $y_i \in \{0, 1\}$ and $\text{logit}(x) = \log(x/(1-x))$. The LASSO estimate $\hat{\beta}_0, \ldots, \hat{\beta}_k$ is the minimizer of the negative log likelihood

$$\sum_{i=1}^{n} \left[-y_i f(\mathbf{z}_i) + \log\left(1 + e^{f(\mathbf{z}_i)}\right)\right]$$

subject to $\sum_{j=1}^{k} |\beta_j| \leq \lambda$. The general LASSO setting of the multivariate logistic regression is the same as that of the multivariate linear regression except $l(y, a) = -ya + \log(1 + e^a)$.

Example 3. *Likelihood Basis pursuit*

The likelihood basis pursuit model with second order interaction terms for classification is basically the same as (1) except $f(\mathbf{z})$ is modelled via the functional ANOVA (analysis of variance) decomposition

$$f(\mathbf{z}) = \beta_0 + \sum_{j=1}^{p} f_j(z_j) + \sum_{j<k} f_{jk}(z_j, z_k). \tag{2}$$

Then, we assume that f_j and f_{jk} are elements of the reproducing kernel Hilbert space (RKHS) $\mathcal{H}_j$ and $\mathcal{H}_{jk}$ with the reproducing kernels K_j and K_{jk}, respectively. Motivated by Kimeldorf and Wahba (1972), we assume that

$$f_j(z_j) = \sum_{r=1}^{n} c_{rj} K_j(z_{rj}, z_j)$$

and

$$f_{jk}(z_j, z_k) = \sum_{r=1}^{n} c_{rjk} K_{jk}((z_{rj}, z_{rk}), (z_j, z_k)).$$

This model is considered by many authors including Gunn and Kandola (2002) and Zhang et al. (2003). Zhang et al. (2003) proposed to estimate c_{rj} and c_{rjk} by minimizing

$$\sum_{i=1}^{n} l(y_i, f(\mathbf{z}_i))$$

subject to $\sum_{r=1}^{n} \sum_{j=1}^{p} |c_{rj}| \leq \lambda_1$ and $\sum_{r=1}^{n} \sum_{j<k} |c_{rjk}| \leq \lambda_2$. This problem can be embedded into the general setting of the LASSO with $d = 2, (p_1, p_2) = (np, np(p-1)/2)$ and $\mathbf{x}_{i1} = (K_j(z_{rj}, z_{ij}), r = 1, \ldots, n, j = 1, \ldots, p), \mathbf{x}_{i2} = (K_{jk}((z_{rj}, z_{rk}), (z_{ij}, z_{ik})), r = 1, \ldots, n, j < k)$, $\boldsymbol{\beta}_1 = (c_{rj}, r = 1, \ldots, n, j = 1, \ldots, p)$, and $\boldsymbol{\beta}_2 = (c_{rjk}, r = 1, \ldots, n, j < k)$.

3. Algorithm

We first assume that the intercept term $\beta_0 = 0$. Let $\mathbf{w}_l = \boldsymbol{\beta}_l/\lambda_l$ and let $\tilde{\mathbf{x}}_{il} = \lambda_l \mathbf{x}_{il}$. Then, the equivalent problem is to find $\mathbf{w} = (\mathbf{w}_1, \ldots, \mathbf{w}_d)$ which minimizes

$$R(\mathbf{w}) = \sum_{i=1}^{n} l\left(y_i, \sum_{l=1}^{d} \tilde{\mathbf{x}}_{il}\mathbf{w}_l^{'}\right)$$

subject $|\mathbf{w}_l|_1 \leq 1$ for $l = 1, \ldots, d$.

Let $\mathbf{z}_{lk} = (\tilde{x}_{1lk}, \ldots, \tilde{x}_{nlk}) \in R^n$ for $k = 1, \ldots, p_l$ and $l = 1, \ldots, d$ and let $\mathbf{z}_l$ be the $n \times p_l$ matrix of $(\mathbf{z}_{l1}^{'}, \ldots, \mathbf{z}_{lp_l}^{'})$. For a given vector $f \in R^n$, define $C(f) = \sum_{i=1}^{n} l(y_i, f_i)$ and let $\nabla C(f) = (\partial C(f)/\partial f_1, \ldots, \partial C(f)/\partial f_n)$ and $\phi(f, \mathbf{z}_{lk}) = \min\{\langle \nabla C(f), \mathbf{z}_{lk}\rangle, \langle \nabla C(f), -\mathbf{z}_{lk}\rangle\}$. Here, $\langle \mathbf{a}, \mathbf{b}\rangle = \sum_{i=1}^{n} a_i b_i$ for $\mathbf{a}, \mathbf{b} \in R^n$.

Let $\mathcal{F}_l = \{\mathbf{z}_{l1}, \ldots, \mathbf{z}_{lp_l}\} \cup \{-\mathbf{z}_{l1}, \ldots, -\mathbf{z}_{lp_l}\}$ and let $co(\mathcal{F}_l)$ is the convex hull (the smallest convex set which contains $\mathcal{F}_l$). Then, the objective of LASSO is to find $\hat{f}$ where

$$\hat{f} = \text{argmin}_{f \in \mathcal{S}} C(f)$$

where $\mathcal{S} = co(\mathcal{F}_1) \oplus \cdots \oplus co(\mathcal{F}_d)$. The basic idea of the gradient LASSO is to find $\hat{f}$ sequentially as follows. Suppose $\hat{f} = \sum_{l=1}^{d} \hat{f}_l$ where $\hat{f}_l \in co(\mathcal{F}_l)$, F_l are the current estimates of $\hat{f}_l$ and $F = \sum_{l=1}^{d} F_l$. For each l, the gradient LASSO finds a direction $f_l \in \mathcal{F}_l$ such that $C(F + \alpha(f_l - F_l))$ decreases most rapidly for some $\alpha \in [0, 1]$ and updates F to $F + \alpha(f_l - F_l)$. Note that $F + \alpha(f_l - F_l)$ is still in $\mathcal{S}$. Now, the Taylor expansion implies

$$C(F + \alpha(f_l - F_l)) \approx C(F) + \alpha\langle \nabla C(F), f_l - F_l\rangle.$$

Hence, the desired direction f_l is the one which minimizes $\langle \nabla C(F), f_l\rangle$. By summing up the above arguments, we propose the gradient descent algorithm for LASSO as follows.

Gradient descent algorithm for LASSO

1. Initialization: Let $\mathbf{w}_l = 0$ for $l = 1, \ldots, d$ and $m = 0$.
2. Repeat until converges
   (a) $F_{ml} = \mathbf{z}_l \mathbf{w}_l^{'}$ and $F_m = \sum_{l=1}^{d} F_{ml}$.
   (b) Find (l, k) which minimizes $\phi(F_m, \mathbf{z}_{lk})$.
   (c) If $\phi(F_m, \mathbf{z}_{lk}) = 0$, then stop the algorithm
   (d) Else
      i. Let $\gamma = -\text{sign}\langle \nabla C(F_m), \mathbf{z}_{lk}\rangle$.
      ii. Let

$$\hat{\alpha} = \text{argmin}_{\alpha \in [0,1]} C(F_m + \alpha(\gamma \mathbf{z}_{lk} - F_{ml}))$$

      iii. Let $w_{lj} = (1 - \hat{\alpha})w_{lj}$ for $j \neq k$ and $w_{lk} = (1 - \hat{\alpha})w_{lk} + \gamma\hat{\alpha}$.
      iv. $m = m + 1$.

When β_0 is to be estimated, we put the constraint on β_0 as $|\beta_0| \leq \lambda_0$ and let $\mathbf{x}_{i0} = 1$. Then, we apply the above algorithm with the augmented input $\mathbf{x}_i^* = (\mathbf{x}_{i0}, \mathbf{x}_i)$. If λ_0 is sufficiently large that the LASSO estimate $\hat{\beta}_0$ satisfies $|\hat{\beta}_0| \leq \lambda_0$, we get the desired estimate. For the choice of λ_0, we recommend $\lambda_0 = \eta\hat{\beta}_0^*$ where $\eta > 1$ and $\hat{\beta}_0^*$ is the minimizer of $\sum_{i=1}^{n} l(y_i, \beta_0)$. Our experience confirms that $\eta \in [2, 3]$ works well.

Remark. When $d = 1$, the proposed algorithm is the same as the L_1 boosting algorithm (Mason et al., 2000) with $\mathcal{F}_1$ as the set of weak learners. The original idea of boosting proposed by Freund and Schapire (1997) is to combine weak learners to make a strong committee. Later, Mason et al. (2000), Friedman et al. (2000), and Friedman (2001) proved that for a given set of weak learners $\mathcal{F}$, the boosting algorithm essentially finds the optimal function $F \in lin(\mathcal{F})$ which minimizes the cost functional $C(F) = \sum_{i=1}^{n} l(y_i, F(\mathbf{x}_i))$ sequentially where $lin(\mathcal{F})$ is the set of all linear combinations of weak learners in $\mathcal{F}$. However, $lin(\mathcal{F})$ is too large that the overfitting emerges in particular for noisy data. To avoid this problem, Mason et al. (2000) proposed the L_1 boosting algorithm, which finds the optimal convex combination of weak learners instead of the optimal linear combination. Recently, Lugosi and Vayatis (2004) justified L_1 boosting theoretically by proving the Bayes risk consistency. Note that even if we are to find the optimal function F, the cost functional C depends on F only through $F(\mathbf{x}_1), \ldots, F(\mathbf{x}_n)$. Hence, we can say that the L_1 boosting algorithm is to find the optimal convex combination of vectors $\{\mathbf{z}_f, f \in \mathcal{F}\}$ where $\mathbf{z}_f = (f(\mathbf{x}_1), \ldots, f(\mathbf{x}_n))$, which is the exactly same as the objective of the gradient LASSO with $d = 1$. There are two important implications in this similarity. First, using the above algorithm, we can extend the L_1 boosting algorithm with more than one side condition. Second, the theoretical results of the gradient descent algorithm for LASSO in the next section can be applied to the L_1 boosting directly. In particular, the convergence rate is new.

4. Convergence Analysis

We assume that C is convex and satisfies the Lipschitz with Lipschitz constant L on $\mathcal{S}$. That is, for any two

vector $\mathbf{g}$ and $\mathbf{h}$ in $\mathcal{S}$,

$$\|\nabla C(\mathbf{g}) - \nabla C(\mathbf{h})\| \leq L\|\mathbf{g} - \mathbf{h}\|$$

where $\|\mathbf{g} - \mathbf{h}\|^2 = \sum_{i=1}^n (g_i - h_i)^2$. We assume that $\mathcal{S}$ is a bounded set. Let $M = \sup_{\mathbf{g},\mathbf{h}\in\mathcal{S}} L\|\mathbf{g} - \mathbf{h}\|^2$. Let $C^* = \inf_{F\in\mathcal{S}} C(F)$ and let $\Delta C(F) = C(F) - C^*$. The following theorem is the main result of this section.

Theorem 1

$$\Delta C(F_m) \leq \frac{2M}{m+2}.$$

We start with the two lemmas.

Lemma 1 *For any $F \in \mathcal{S}, \mathbf{v} \in co(\mathcal{F}_l)$ and $\alpha \in [0,1]$,*

$$C(F+\alpha(\mathbf{v}-F_l)) \leq C(F)+\alpha\langle\nabla C(F), \mathbf{v}-F_l\rangle + \frac{M\alpha^2}{2}.$$

Proof. Define $\Phi : R \to R$ by $\Phi(\alpha) = C(F+\alpha(\mathbf{v}-F_l))$. Let $\Phi^{'}(\alpha) = d\Phi(\alpha)/d\alpha$. Then, we have

$$\Phi^{'}(\alpha) = \langle\nabla C(F + \alpha(\mathbf{v} - F_l)), \mathbf{v} - F_l\rangle.$$

Hence,

$$\begin{aligned} & |\Phi^{'}(\alpha) - \Phi^{'}(0)| \\ = \; & |\langle\nabla C(F + \alpha(\mathbf{v} - F_l)) - \nabla C(F), \mathbf{v} - F_l\rangle| \\ \leq \; & \|\nabla C(F + \alpha(\mathbf{v} - F_l)) - \nabla C(F)\|\|\mathbf{v} - F_l\| \\ \leq \; & L\alpha\|\mathbf{v} - F_l\|^2 \end{aligned}$$

for $\alpha \in [0,1]$. The first inequality is by Cauchy-Schwarz inequality. Thus,

$$\begin{aligned} \Phi^{'}(\alpha) \; &\leq \; \Phi^{'}(0) + L\alpha\|\mathbf{v} - F_l\|^2 \\ &= \; \langle\nabla C(F), \mathbf{v} - F_l\rangle + L\alpha\|\mathbf{v} - F_l\|. \end{aligned}$$

Hence,

$$\begin{aligned} & \Phi(\alpha) - \Phi(0) \\ = \; & \int_0^\alpha \Phi^{'}(s)ds \\ \leq \; & \int_0^\alpha \left(\langle\nabla C(F), \mathbf{v} - F_l\rangle + Ls\|\mathbf{v} - F_l\|^2\right) ds \\ = \; & \alpha\langle\nabla C(F), \mathbf{v} - F_l\rangle + \frac{L}{2}\|\mathbf{v} - F_l\|^2\alpha^2. \end{aligned}$$

Since $L\|\mathbf{v} - F_l\|^2 \leq M$, the proof is done. □

Lemma 2 *For any given $F \in \mathcal{S}$, let $\mathbf{v}$ be a vector in $\mathcal{F}_l$ which minimizes $\langle\nabla C(F), \mathbf{v} - F_l\rangle$ and let*

$$\hat{\alpha} = \text{argmin}_{\alpha\in[0,1]} C(F + \alpha(\mathbf{v} - F_l)).$$

Then,

$$\begin{aligned} & \Delta C(F + \hat{\alpha}(\mathbf{v} - F_l)) \\ \leq \; & \begin{cases} \Delta C(F) - \frac{\Delta C(F)^2}{2M} & \text{if } \Delta C(F) \leq M \\ \frac{M}{2} & \text{otherwise} \end{cases} \end{aligned}$$

Proof. For given $F \in \mathcal{S}$, define the Bregman divergence $d(\mathbf{g})$ on $co(\mathcal{F}_l)$ by

$$d(\mathbf{g}) = C(\mathbf{g}) - C(F) - \langle\nabla C(F), \mathbf{g} - F_l\rangle.$$

Note that $\inf_{\mathbf{g}\in co(\mathcal{F}_l)} d(\mathbf{g}) \geq 0$ for all $F \in \mathcal{S}$ since C is convex. Hence, we have

$$\langle\nabla C(F), \mathbf{v} - F_l\rangle \leq C(\mathbf{v}) - C(F) \leq -\Delta C(F)$$

since $C(\mathbf{v}) \geq C^*$. Hence, Lemma 1 implies that

$$\Delta C(F+\alpha(\mathbf{v}-F_l)) \leq \Delta C(F) - \alpha\Delta C(F) + \frac{M}{2}\alpha^2. \quad (3)$$

By taking the maximum on the right hand side of (3) with respect to α on $[0,1]$, we will get the desired result. □

Proof of Theorem 1. We will use the mathematical induction. First, consider the case of $m = 1$. Then,

$$\Delta C(F_1) \leq M/2 = 2M/4 \leq 2M/3 = 2M/(m+2),$$

and so Theorem 1 is true for $m = 1$.

Next, suppose $\Delta C(F_m) \leq 2M/(m+2)$ for $m \geq 2$. Note that $2M/(m+2) \leq M$ for $m \geq 2$. Then, we have

$$\Delta C(F_{m+1}) \leq \Delta C(F_m) - \frac{\Delta C(F_m)^2}{2M}.$$

Since $x - x^2/2M$ is increasing on $[0, M]$, we have

$$\begin{aligned} \Delta C(F_{m+1}) \; &\leq \; \frac{2M}{m+2} - \frac{(2M)^2}{2M(m+2)^2} \\ &= \; \frac{2M}{m+2} - \frac{2M}{(m+2)^2} \leq \frac{2M}{m+3}, \end{aligned}$$

which completes the proof. □

Remark. The convergence rate of the algorithm to find the optimal convex combination of elements in the set $\mathcal{S}$ has been studied by many authors including

Table 1. Linear regression model

λ	1	2	3	4	5
GRADIENT ALGORITHM					
DEVIANCE	560.69	471.05	419.66	388.90	371.71
(S.D)	115.74	97.64	89.60	86.36	86.02
NONZERO'S	1.96	3.50	5.22	6.64	7.94
(S.D)	0.95	1.18	1.28	1.40	1.25
QP PROGRAM					
DEVIANCE	555.80	464.82	411.67	380.15	362.43
(S.D)	113.95	96.37	89.31	86.29	85.44
NONZERO'S	1.94	3.52	5.30	7.08	8.52
(S.D)	0.91	1.16	1.27	1.43	1.33

Table 2. Logistic regression model

λ	1	2	3	4	5
GRADIENT ALGORITHM					
DEVIANCE	23.52	18.59	15.81	14.09	12.92
(S.D)	1.52	2.14	2.60	2.99	3.34
NONZERO'S	2.06	3.42	4.70	5.94	6.58
(S.D)	0.68	1.25	1.34	1.60	1.64
QP PROGRAM					
DEVIANCE	23.49	18.52	15.63	13.78	12.51
(S.D)	1.53	2.11	2.55	2.94	3.30
NONZERO'S	2.04	3.44	4.80	5.94	6.96
(S.D)	0.67	1.26	1.32	1.70	1.65

Jones (1992), Barron (1993) and Zhang et al. (2003). In particular, Zhang et al. (2003) considered the sequential greedy approximation algorithm which updates the current convex combination F_m to the new one F_{m+1} by $(1-\hat{\alpha})F_m + \hat{\alpha}\hat{f}$ where

$$(\hat{\alpha}, \hat{f}) = \text{argmin}_{f \in \mathcal{S}, \alpha \in [0,1]} C((1-\alpha)F_m + \alpha f).$$

He proved that $\Delta C(F_m) = O(m^{-1})$. Since the sequential greedy optimization algorithm is the fastest sequential algorithm, Theorem 1 implies that the convergence rate of the gradient descent algorithm for LASSO is almost optimal.

5. Simulation

In this section, we compare the performance of the gradient descent algorithm with the standard QP algorithm by simulation. We first consider the multivariate linear regression model given as

$$y = \beta_0 + \beta_1 z_1 + \cdots + \beta_{10} z_{10} + \epsilon.$$

The inputs z_i are generated independently from the standard normal distribution. Also, ϵ follows the standard normal distribution. We set the true value of $\boldsymbol{\beta}$ at $(2, -0.5, 1, 0, 2, 0, 0, 0, 0, 0)$ and let $\beta_0 = 0$. We generate 50 data sets with 50 samples from the above model. Table 1 compares the proposed algorithm with the QP. The results are calculated based on the 50 data sets of the sample size 50. For the QP program, we use the algorithm of Osborne et al. (2000) implemented in *R* system.

The deviance (the empirical risk) of the gradient algorithm is slightly larger than that of the QP. This is because the gradient algorithm is asymptotic in the sense that it converges to the optimum when the number of iteration goes to infinite. We stop the algorithm when the decrease of the deviance is small. We can reduce the deviance of the gradient algorithm more by iterating the algorithm more. Note that the number of nonzero coefficients are almost identical.

Table 2 summarizes the results of the multivariate logistic regression. We let $f(\mathbf{z}) = \beta_0 + \beta_1 z_1 + \cdots + \beta_{10} z_{10}$ with $\boldsymbol{\beta} = (2, -1, 0, 4, 0, 0, 0, 0)$ and $\beta_0 = 0$. The results are similar to those for the multivariate linear regression model. To sum up, the gradient descent algorithm for LASSO proposed in the paper finds the almost optimum with less computation than the QP does.

6. Analysis of Pima Indians Diabetes dataset using the Likelihood Basis Pursuit

In this section, we analyze the Pima Idians Diabetes data using the likelihood pursuit. The data set, which has been taken from the UCI Repository Of Machine Learning Databases at http://www.ics.uci.edu/∼mlearn/MLRepository.html, has 768 observations on 8 input variables and two class output variable. The 8 input variables and one output variable are

1. "preg" : Number of times pregnant
2. "glu" : Plasma glucose concentration (glucose tolerance test)
3. "pres" : Diastolic blood pressure (mm Hg)
4. "tri" : Triceps skin fold thickness(mm)
5. "insu" : 2-Hour serum insulin (mu U/ml)
6. "mass" : Body mass index (weight in kg/(height in m)^2)
7. "pedi" : Diabetes pedigree function
8. "age" : Age (years)
9. "diabetes" : Class output variable (500 "−" and 268 "+" tests for diabetes)

The model starts with the main and second order interaction terms as in (2). We let $\lambda_1 = \lambda_2 = \lambda$ and set $\lambda = 6$ which is selected by the 5-fold cross-validation. The Gaussian kernel with the scale parameter reciprocally proportional to the dimension of inputs is used. Figure 1 shows the change of the deviance on the number of the iteration, which shows that the gradient algorithm for LASSO converges fairly fast. Figure 2 presents the L_1 norm of the selected components. Here, L_1 norm of a given component $f_j(z_j)$ is defined by $\sum_{i=1}^{n} |\sum_{l=1}^{n} c_{lj} K_j(z_{lj}, z_{ij})| /n$. L_1 norm of the second order interaction term is defined similarly. The all main effect terms except that for "tri" are selected, and for the interaction term, "tri*ins" is selected. Figures 3 and 4 draw the functional relation of the 4 most important main effect terms and one interaction term, respectively. All the functions are rather picky, which is mainly due to the shape of the kernel (Gaussian kernel). There is no special reason for choosing the Gaussian kernel. Different choice of kernels such as polynomial or spline kernels will result in smoother curves.

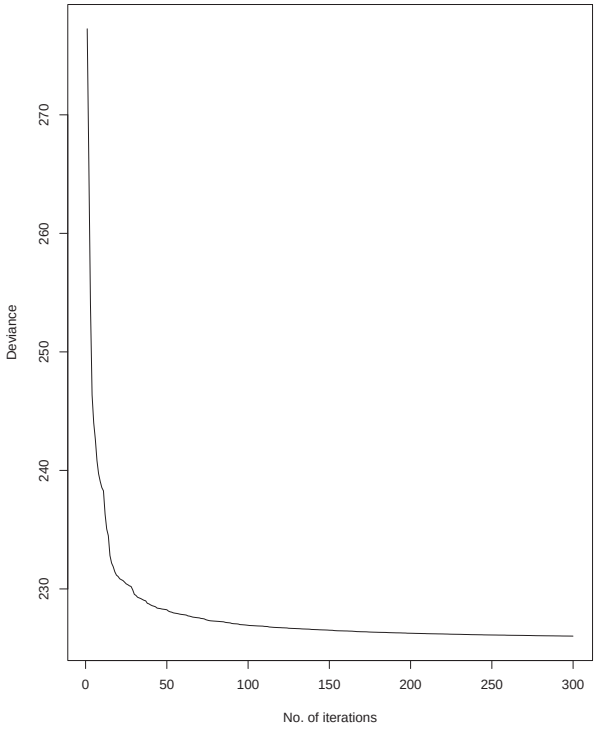

Figure 1. Deviance

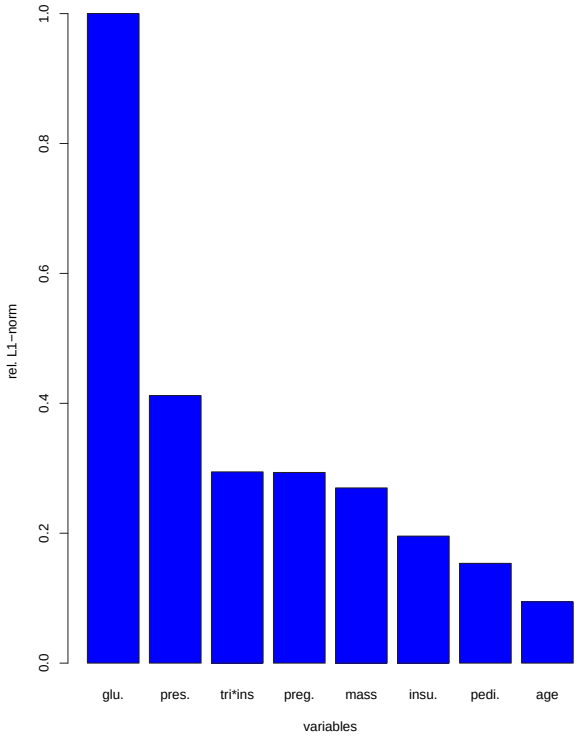

Figure 2. The L1-norms of the components

7. Discussion

In this paper, we proposed the gradient descent method for LASSO, which is computationally simpler and faster than the standard QP program even though it is less accurate than QP or nonlinear program. We showed theoretically as well as empirically that the proposed algorithm converges fairly fast and gives reliable results

One interesting feature of the proposed algorithm is that the convergence rate is independent on the dimension of input. The convergence rate is important since less iteration gives more sparse solutions. Hence, the proposed algorithm is well suited with problems with large dimensional inputs such as the likelihood basis pursuit. In the analysis of the Pima Indians Diabetes dataset, the algorithm converges to the near optimum only after 50 iterations. This is surprising since the number of components is 36. Also, the final model, which contains only 8 components out of 36, is very sparse.

One problem of the proposed algorithm is that the convergence speed is rather slow at the near optimum.

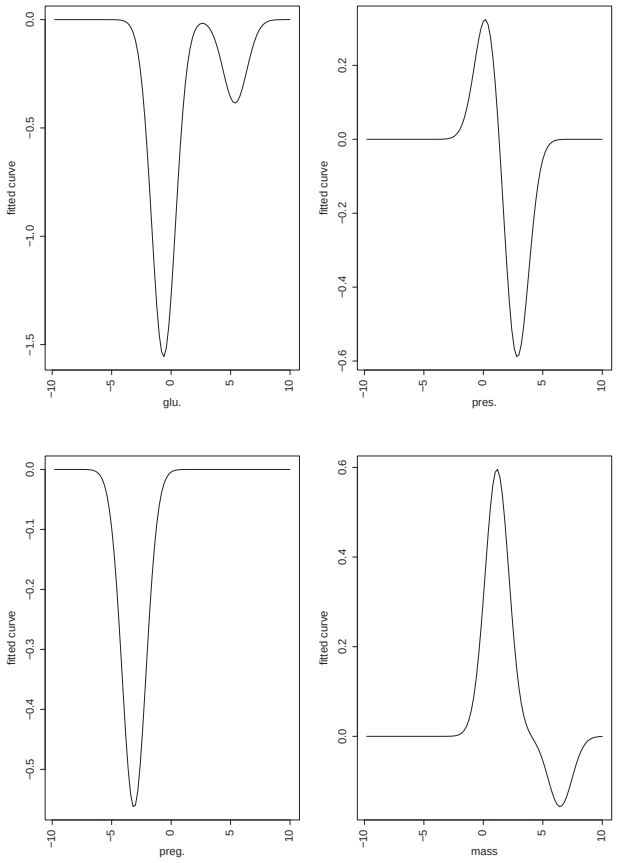

Figure 3. The 4 most important main effects

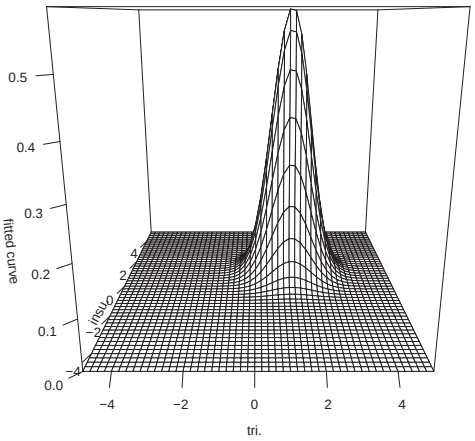

Figure 4. The interaction term "tri*ins"

Figure 1 represents the typical behavior of the deviance. That is, the deviance is dropped very fast in the early stage, and then it decreases slowly. This is why the deviances of the proposed algorithm are slightly larger than those of the QP in our simulation results. Accelerating the convergence speed at the near optimum is worth pursuing.

Acknowledgments

We would like to thank the Area Chair and the unknown reviewers whose fruitful comments led to significant improvements in presentation. This research was supported in part by US Air Force Research Grant F62562-03-P-0658 and in part by KOSEF through Statistical Research Center for Complex Systems at Seoul National University, Korea.

References

Bakin, S. (1999). *Adaptive regression and model selection in data mining problem.* Doctoral dissertation, Australian National University, Australia.

Barron, A. R. (1993). A universal approximation bounds for superpositions of a sigmoidal function. *IEEE Trans. Inform. Theory*, *39*, 930–945.

Chen, S. S., Donoho, D. L., & Saunders, M. A. (1999). Atomic decomposition by basis pursuit. *SIAM Journal on Scientific Computing*, *20*, 33–61.

Efron, B., Hastie, T., Johnstone, I., & Tibshirani, R. (2004). Least angle regression. *Annals of Statistics*.

Freund, Y., & Schapire, R. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, *55*, 119–139.

Friedman, J. (2001). Greedy function approximation : a gradient boosting machine. *Annals of Statistics*, *29*, 1189–1232.

Friedman, J., Hastie, T., & Tibshirani, R. (2000). Additive logistic regression: A statistical view of boosting. *Annals of Statistics*, *28*, 337–374.

Grandvalet, Y., & Canu, S. (1999). Outcomes of the equivalence of adaptive ridge with least absolute shrinkage. In M. Kearns, S. Solla and D. Cohn (Eds.), *Advances in neural information processing systems*, vol. 11, 445–451. MIT press.

Gunn, S. R., & Kandola, J. S. (2002). Structural modelling with sparse kernels. *Machine Learning*, *48*, 115–136.

Jones, L. K. (1992). A simple lemma on greedy approximation in Hilbert space and convergence rates for projection pursuit regression and neural network training. *Annals of Statistics*, *20*, 608–613.

Kimeldorf, G., & Wahba, G. (1972). Some results on Tchebychffian spline functions. *J. Math. Anal. Applic.*, *33*, 82–95.

Knight, K., & Fu, W. J. (2000). Asymptotics for lasso-type estimators. *Annals of Statistics*, *28*, 1356–1378.

Lokhorst, J., Turlach, B. A., & Venables, W. N. (1999). Lasso2*: An s-plus library to solve regression problems while imposing an l1 constraint on the parameters.

Lugosi, G., & Vayatis, N. (2004). On the bayes risk consistency of regularized boosting methods. *Annals of Statistics*.

Mason, L., Baxter, L., Bartlett, P., & Frean, M. (2000). Functional gradient techniques for combining hypotheses. In A. J. Smola, P. L. Bartlett, B. Scholkopf and D. Schuurmans (Eds.), *Advances in large margin classifiers*. Cambridge: MIT press.

Osborne, M. R., Presnell, B., & Turlach, B. A. (2000). A new approach to variable selection in least squares problems. *IMA Journal of Numerical Analysis*, *20*, 389–403.

Perkins, S., Lacker, K., & Theiler, J. (2003). Grafting: Fast, incremetal feature selection by gradient descent in function space. *Journal of Machine Learnign Research*, *3*, 1333–1356.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *J.R. Statist. Soc. (B)*, *58*, 267–288.

Zhang, H., Wahba, G., abd M. Voelker, Y. L., Ferris, M., Klein, R., & Klein, B. (2003). *Variable selection and model building via likelihood basis pursuit* (Technical Report 1059r). Department of Statistics, University of Wisconsin, Madison, WI.