Hindawi Publishing Corporation
Journal of Applied Mathematics and Decision Sciences
Volume 2007, Article ID 24053, 13 pages
doi:10.1155/2007/24053

Research Article

On Solving L_q-Penalized Regressions

Tracy Zhou Wu, Yingyi Chu, and Yan Yu

Received 1 November 2006; Accepted 18 July 2007

Recommended by Fernando Beltran

L_q-penalized regression arises in multidimensional statistical modelling where all or part of the regression coefficients are penalized to achieve both accuracy and parsimony of statistical models. There is often substantial computational difficulty except for the quadratic penalty case. The difficulty is partly due to the nonsmoothness of the objective function inherited from the use of the absolute value. We propose a new solution method for the general L_q-penalized regression problem based on space transformation and thus efficient optimization algorithms. The new method has immediate applications in statistics, notably in penalized spline smoothing problems. In particular, the LASSO problem is shown to be polynomial time solvable. Numerical studies show promise of our approach.

1. Introduction

L_q-penalized regression estimates the regression coefficients $\vec{\beta}$ by minimizing the penalized sum of squares:

$$\min_{\vec{\beta}} \|\mathbf{Y} - \mathbf{X}\vec{\beta}\|_2^2 + \sum_{i=1}^{m} \lambda_i |\beta_i|^q, \quad q > 0, \tag{1.1}$$

where $\mathbf{Y}$ and $\mathbf{X}$ are observed response vector and design matrix, respectively, $\|\cdot\|$ is the Euclidean norm, and $|\cdot|$ denotes the absolute value. λ_i in (1.1) is called smoothing parameter in statistics literature. L_q-penalized regression arises in multidimensional statistical modeling when both accuracy and complexity of the model are considered important. Compared with ordinary least squares (OLS) regression, penalized regression tends

to shrink or set some (small) coefficients to zero to achieve lower variance and higher prediction accuracy at a little cost of bias.

Some special cases of (1.1) are worth mentioning. The case of $q = 2$ is equivalent to ridge regression introduced by Hoerl and Kennard [1], where small positive quantities are added to the diagonal of $\mathbf{X}'\mathbf{X}$ when the original least squares problem is not orthogonal. Problem (1.1) is equivalent to bridge regression (Frank and Friedman [2]), where the residual sum of squares is norm constrained rather than penalized. Due to the equivalence between L_q-penalized regression and Bridge regression, setting $q = 1$ in (1.1) results in the LASSO estimation problem (Tibshirani [3]). The LASSO shrinks the OLS estimators of $\vec{\beta}$ towards zero and potentially sets smaller β_i's to zero, operating as shrinkage and variable selection operator. Subset selection fits into the penalized regression framework with $q = 0$, where the number of coefficients (variables) is bounded. There is no reason to exclude other possible values of q when considering different situations.

Except for the simplest case with $q = 2$, where the solution can be worked out explicitly, computational difficulty arises in solving L_q-penalized regression problems. The difficulty lies partly in the absolute value function involved in the penalty part and the nonsmoothness inherited from it.

Fu [4] studied the bridge regression problem and described a modified Newton-Raphson algorithm for the case $q > 1$ and a shooting algorithm for $q = 1$. Ruppert and Carroll [5] adopted the iterated reweighted ridge regression (IRRR) method to solve problems with arbitrary L_q-penalty function. Yet neither is the optimality proved, nor is it known how closely IRRR finds the real minimum. Among L_q problems, solution methods for the LASSO problem ($q = 1$) have been studied most intensively. The method given in the original paper of Tibshirani [3] is an active set approach, where bounding inequalities are introduced sequentially and the convergence is argued to be guaranteed in a finite (though potentially exponential) number of iterations. There are other authors having done work for related problems. In Osborne et al. [6, 7], the primal-dual relationship of the LASSO problem is studied, based on which a descent algorithm and a homotopy algorithm are introduced. Recent work of Efron et al. [8] recommends a modified LARS algorithm, where the parameters are incrementally added to or deleted from the active set (of nonzero parameters) one at a time. The theoretical complexities of these algorithms are unknown except for Tibshirani [3].

We investigate an alternative solution method for the general L_q-penalized regression problem (1.1) based on space transformation and efficient optimization algorithms. Our idea resembles the one briefly mentioned by Tibshirani [3] as David Gay's suggestion for $q = 1$, though there are no details given by Tibshirani [3]. To our best knowledge, there is no previous documented work addressing similar idea theoretically. We hope to fill the gap by providing more detailed methods for general L_q problems and supporting our method with solid theoretical results.

Throughout the paper, we assume that q is a nonnegative real number. We propose a general treatment for L_q case. A space transformation for problem (1.1) is introduced and the original problem is recast into a minimization of a *smooth function* under only *bound constraints*. The availability of derivatives information, combined with the simple structure of the constraints, enables a range of efficient algorithms that exploit these

properties to be used in solving the problem. In particular, the trust region method (Moré [9]; Conn et al. [10]), which makes use of the gradient and Hessian, has been developed as one of the standard algorithms for solving large-scale problems, and its adaptations for bound-constrained problems are available (e.g., Coleman and Li [11]). Furthermore, the transformed problem preserves the convexity for $q \geq 1$ and thus it can be solved to optimality by using the aforementioned algorithms. For the case of $q = 1$, the recast model is shown to be a positive semidefinite quadratic program, for which polynomial-time algorithm is available (see, e.g., Kozlov et al. [12]; Chung and Murty [13]; Ye and Tse [14]). In other words, the L_1-penalized problem is proved to be a *tractable* problem (i.e., a $\mathcal{P}$ problem).

The proposed method has implications for solving several important classes of statistical problems. First, the special case with $q = 1$ is an equivalent form of the LASSO. Therefore, theoretically, the LASSO problem is proved to be a $\mathcal{P}$ problem. Practically, the proposed method in this paper can be used to generate the solutions efficiently for any given set of smoothing parameters. Thus it allows fast data-driven selection of optimal smoothing parameters. Since our problem setup allows multiple smoothing parameters $\{\lambda_i\}_{i=1}^m$, applications of our algorithm to major smoothing problems, notably the penalized regression spline with additive main effects and interactions, are straightforward.

The rest of the paper is organized as follows. Section 2 introduces the transformation approach which recasts the original problem into a new problem that is easier to optimize. Computational complexity and algorithms are presented. Section 3 addresses the issue of choosing the smoothing parameter in the penalized regression. We apply our new approach to some statistical problems in Section 4. Section 5 concludes the paper. All the mathematical proofs are in Section 6.

2. The space transformation approach

In this section, we first discuss transformation for the general model with L_q penalty. Then we also consider the special L_1 case not only for its special properties, but also because the L_1 norm is probably the most popular nonquadratic penalty.

2.1. Transformation. Consider the model

$$\min_{\vec{\beta}} \|\mathbf{Y} - \mathbf{X}\vec{\beta}\|_2^2 + \sum_{i=1}^{m} \lambda_i |\beta_i|^q. \tag{2.1}$$

We introduce binary variables b_i to indicate whether β_i is positive or not. The original minimization (2.1) is equivalently written as

$$\begin{aligned} &\min_{\vec{\beta}} \|\mathbf{Y} - \mathbf{X}\vec{\beta}\|_2^2 + \sum_{i=1}^{m} \lambda_i (b_i \cdot \beta_i - (1 - b_i) \cdot \beta_i)^q, \\ &\text{s.t. } b_i \cdot \beta_i \geq 0, \quad (1 - b_i) \cdot \beta_i \leq 0, \quad b_i \in \{0, 1\}. \end{aligned} \tag{2.2}$$

The first two constraints in model (2.2) precisely represent the relation between b_i and β_i, that is, $\beta_i \geq 0$ if $b_i = 1$; $\beta_i \leq 0$ otherwise.

Further, we introduce variables z_i and $\bar{z}_i$, where $z_i = b_i \cdot \beta_i$ and $\bar{z}_i = (1 - b_i) \cdot \beta_i$. Then, the regression parameters $\vec{\beta}$ can be recovered by $\vec{\beta} = \vec{z} + \vec{\bar{z}}$, where $\vec{z} = [z_1, \ldots, z_m]^T$ and $\vec{\bar{z}} = [\bar{z}_1, \ldots, \bar{z}_m]^T$.

Now the original minimization problem (2.1) on the $\vec{\beta}$ space is transformed to the space of $\vec{z}$ and $\vec{\bar{z}}$:

$$\begin{aligned} &\min_{\vec{\beta}} \left|\left|\mathbf{Y} - \mathbf{X}(\vec{z} + \vec{\bar{z}})\right|\right|_2^2 + \sum_{i=1}^{m} \lambda_i (z_i - \bar{z}_i)^q, \\ &\text{s.t. } z_i \geq 0 \quad \forall i, \quad \bar{z}_i \leq 0 \quad \forall i, \quad z_i \cdot \bar{z}_i = 0 \quad \forall i. \end{aligned} \tag{2.3}$$

The equivalence is formally established as follows.

Lemma 2.1 (*z-space equivalence*). *There is a bijection between the feasible solutions of (2.3) and those of (2.1), such that the corresponding solutions in the two spaces give the same objective value.*

Both the original and transformed problems are feasible and there exist optimal solutions for each problem. More importantly, the nonlinear constraint $z_i \cdot \bar{z}_i = 0$ in (2.3) is shown (in the following lemma) to be redundant, simplifying the problem to

$$\min_{\vec{\beta}} \left|\left|\mathbf{Y} - \mathbf{X}(\vec{z} + \vec{\bar{z}})\right|\right|_2^2 + \sum_{i=1}^{m} \lambda_i (z_i - \bar{z}_i)^q, \quad \text{s.t. } z_i \geq 0 \quad \forall i, \quad \bar{z}_i \leq 0 \quad \forall i. \tag{2.4}$$

Lemma 2.2 (redundancy of bilinear constraints). *Suppose* $(\vec{z^*}, \vec{\bar{z}^*})$ *is an optimal solution of (2.4). Then* $(\vec{z^{*\prime}}, \vec{\bar{z}^{*\prime}})$ *defined as* $((\vec{z^*} + \vec{\bar{z}^*})^+, (\vec{z^*} + \vec{\bar{z}^*})^-)$, *where* u^+ *denotes* $(\max(0, u_1), \ldots, \max(0, u_d))^T$ *and* u^- *denotes* $(\min(0, u_1), \ldots, \min(0, u_d))^T$ *for any* $d \in R^d$, *is also (if it is different at all) an optimal solution of (2.4), and it satisfies the constraint* $z_i \cdot \bar{z}_i = 0$ $\forall i = 1, \ldots, m$.

Lemma 2.2 shows the redundancy of the complicating constraints in (2.3), in the sense that, to find an optimal solution of (2.3), one can solve the problem (2.4) instead. If an optimal solution from (2.4), $(\vec{z^*}, \vec{\bar{z}^*})$, satisfies the constraints, then it is also an optimal solution of (2.3). Otherwise, $(\vec{z^*}, \vec{\bar{z}^*})$ can be altered to $(\vec{z^{*\prime}}, \vec{\bar{z}^{*\prime}})$, which is again an optimal solution of (2.3).

Lemma 2.2 hold for any $q > 0$.

If we assume all λ_i to be positive, then Lemma 2.2 can be stated in a simpler way as follows.

Corollary 2.3. *Assume all* $\lambda_i > 0$. *Then any optimal solution of (2.4) is also an optimal solution of (2.3) and vice versa.*

2.2. Solving the transformed problem. The transformed problem becomes differentiable. The analytical forms of the gradient and Hessian are immediately available. Therefore, a number of multidimensional search algorithms for nonlinear programing (such as the Newton's method) can be applied. In our implementation, we employ a trust-region-based algorithm, where a local quadratic program is used to calculate the search direction.

The transformed problem is only subject to bound constraints, which can be handled efficiently, for example, using the techniques of Lin and More [15] and Coleman and Li [11]. The details of the implemented algorithm are further described in Section 2.4.

Furthermore, the transformation preserves the convexity of the original problem when $q \geq 1$.

Lemma 2.4 (convexity). *If $q \geq 1$, the problem (2.4) is convex.*

The convexity implies that any local minima of the problem are global. Therefore, for $q \geq 1$, the aforementioned algorithms are able to solve the problem to *optimality*. If $q < 1$, the original problem is already nonconvex, and thus global optimality is not guaranteed with the proposed method, while a local optima can still be obtained using the same method.

2.3. The transformed problem for L_1 case. Now we consider solving the transformed problem for $q = 1$, not only because of our special interest in L_1-related models (e.g., LASSO), but because, when $q = 1$, the above problem possesses extra nice properties. So, the original problem is

$$\min_{\vec{\beta}} \|\mathbf{Y} - \mathbf{X}\vec{\beta}\|_2^2 + \sum_{i=1}^{m} \lambda_i |\beta_i|. \tag{2.5}$$

Following the transformation outlined in Section 2.1, the equivalent problem is

$$\begin{aligned} \min_{\vec{\beta}} \|\mathbf{Y} - \mathbf{X}(\vec{z} + \vec{\bar{z}})\|_2^2 + \sum_{i=1}^{m} \lambda_i (z_i - \bar{z}_i), \\ \text{s.t. } z_i \geq 0 \quad \forall i, \quad \bar{z}_i \leq 0 \quad \forall i, \\ (z_i \cdot \bar{z}_i = 0 \ \forall i), \end{aligned} \tag{2.6}$$

where the last constraint is again redundant and we can drop it. So, to avoid confusion, in what follows, we refer problem (2.6) without the last complicating constraint.

An immediate consequence of this transformation is that the L_1-penalized problem (2.5) can be proved to be polynomial time solvable.

Theorem 2.5 (complexity). *Solving (2.5) to optimality is a $\mathcal{P}$ problem.*

It is easy to show that (2.6) is a positive semidefinite (PSD) quadratic program with a polynomial number of bound constraints, for which efficient (polynomial time) algorithms exist (see, e.g., Chung and Murty [13]). The tractability is attributed to the convexity shown in Section 2.2 of the equivalent quadratic program (2.6); in which case the local optimum equals the global optimum. Furthermore, the problem (2.6) only has bound constraints on its variables, which allows the application of highly efficient solution techniques specific to box- or bound-constrained quadratic programing (see, e.g., Moré and Toraldo [16] and Coleman and Li [17]).

2.4. Algorithms. The transformation has enabled the application of a range of algorithms in the nonlinear programing literature to solve the regression problem. The algorithm used in our study is based on the trust region method, which has evolved for over two decades as a standard technique in solving large nonlinear programs. The algorithm has been implemented as a function in Matlab (e.g., quadprog, fmincon). We provide a general description for the implemented algorithm, while technical details can be found in the original paper of Coleman and Li [11] as well as Conn et al. [10].

The algorithm starts from an interior (i.e., strictly feasible) solution (for our problem such a solution is trivially obtainable). The algorithm proceeds by generating a series of interior solutions using the trust region method. In a small neighborhood of the current solution, the objective function is approximated by *local quadratic function*, which is obtained as the second-order Taylor expansion at this point. In other words, in the current neighborhood, the local quadratic function is *trusted* as a sufficiently precise approximation of the real objective function. This local quadratic function is minimized over the neighborhood, to obtain a *descending step* (with respect to the local quadratic approximation). In fact, to reduce the complexity of solving the local quadratic program, it is minimized only on a two dimensional subspace, which is determined by the technique of Coleman and Li [11]. The algorithm ensures that the new solution point is still interior to the feasible region in the following way. If the descending step happens to cross one or more bound constraints, then the step is reflected with respect to the first bound constraint that it crosses. By this way, the next interior solution point is found. The new solution is evaluated on the real objective function. If the new objective value is indeed better than the previous one, then this step of move is confirmed and the algorithm proceeds to find the next interior solution point. Otherwise, the local quadratic function is not considered to be trustable as a good approximation. In this scenario, the algorithm backtracks to the previous solution, but the neighborhood is shrunk to recompute a descending step. The algorithm proceeds as above until the series of interior solutions converges. The convergence of the algorithm to local optima is proved in Coleman and Li [11].

If $q \geq 1$, the above algorithm finds a global optimal solution, since the problem is convex and local optima equal global optima. If $q < 1$, however, the local optima being found are not guaranteed to be a global optimal solution. In this case, we implement a simple random multistart method to try to find a global solution with more confidence, although this can not be guaranteed.

3. Selection of optimal smoothing parameters

The vector of smoothing parameters $\vec{\lambda} = \{\lambda_j\}_{j=1}^{m}$ is not predetermined. When each λ_j equals zero, there is no penalty on regression coefficients at all, and so the situation is the same as ordinary least squares (OLS). When each λ_j goes to ∞, any nonzero coefficients will result in a large increase in the objective function. Thus the minimization tends to give very few nonzero β's. There is a tradeoff between the penalty part and the sum of squared errors (sometimes called "data fidelity"). In other words, $\vec{\lambda}$ controls the amount of smoothing and the selection of optimal smoothing is crucial. With efficient algorithms

to solve the penalized regression problem (at given smoothing parameters) outlined in Section 2, data-driven procedure for choosing $\vec{\lambda}$ becomes less daunting.

There are various criteria to choose $\vec{\lambda}$. One popular choice is cross-validation (CV). The cross-validation score is defined as $\mathrm{CV}(\vec{\lambda}) = (1/n)\sum_{i=1}^{n}\{y_i - \vec{x}_i^{\,T}\hat{\vec{\beta}}^{(-i)}\}^2$, where $\hat{\vec{\beta}}^{(-i)}$ is the minimizer of (1.1) when omitting the ith data point $(y_i, \vec{x}_i)$. So CV is a measure of prediction power of the estimated model. Optimal smoothing parameters $\vec{\lambda}$ minimize $\mathrm{CV}(\vec{\lambda})$ and the process of finding that optimal $\vec{\lambda}$ usually requires computing the value of $\mathrm{CV}(\vec{\lambda})$ for each grid of $\vec{\lambda}$. However, to compute each $\mathrm{CV}(\vec{\lambda})$ directly, it is necessary to run n regressions, which is not efficient. Generalized cross-validation (GCV) approximates CV and can be expressed in a computationally expedient way when $q = 2$ via the following formula in Green and Silverman [18]:

$$\mathrm{GCV}(\vec{\lambda}) = n^{-1}\frac{\sum_{i=1}^{n}\{y_i - \vec{x}_i^{\,T}\hat{\vec{\beta}}\}^2}{\{1 - n^{-1}\operatorname{tr}A(\vec{\lambda})\}^2}, \tag{3.1}$$

where $A(\vec{\lambda})$ is the hat matrix such that $\hat{\vec{Y}} = A(\vec{\lambda})\vec{Y}$. Usually no closed-form expression of GCV is available for arbitrary q and approximation is needed for quick automatic selection of $\vec{\lambda}$ (see, e.g., Section 4 in Tibshirani [3] and Section 5 in Fu [4] for details). The extension, to which such approximation is useful, is however subject to skepticism by some numerical mathematicians. In our email inquiry about such issues, Turlach shared with us the view that GCV approximated in aforementioned papers is not reliable sometimes and one would rather choose smoothing parameters subjectively (based on visual evaluation of the fits). Similar comment can be found in Osborne et al. [6]. Our experience is that many times GCV (approximated) gives guidance of choosing an optimal smoothing parameter but it does not always work. Other selection method based on information criteria, such as AIC, may be considered as well. In fact, smoothing parameter selection is always a topic that researchers want to explore more about and it remains to be solved.

4. Numerical examples

4.1. Linear regression with diabetes data. As the first example, we try our algorithm on L_1 regression. We apply our method to the diabetes data which have been studied many times. The dataset contains $m = 10$ covariates, including age, sex, body mass index (BMI), average blood pressure (BP), six blood serum measurements, the response of interest, and a quantitative measure of disease progression one year after the covariates are measured. The goal is to build a model to predict the response y from covariates $x_1, \ldots, x_{10}$ and find important factors in disease progression.

Linear regression is fitted between the response variable and ten covariates. In order to produce a more parsimonious and interpretable model, L_1 penalty is imposed on regression coefficients and a single smoothing parameter λ is used to penalize all the regression coefficients. Data are standardized in the same way as in Efron et al. [8] prior to regression.

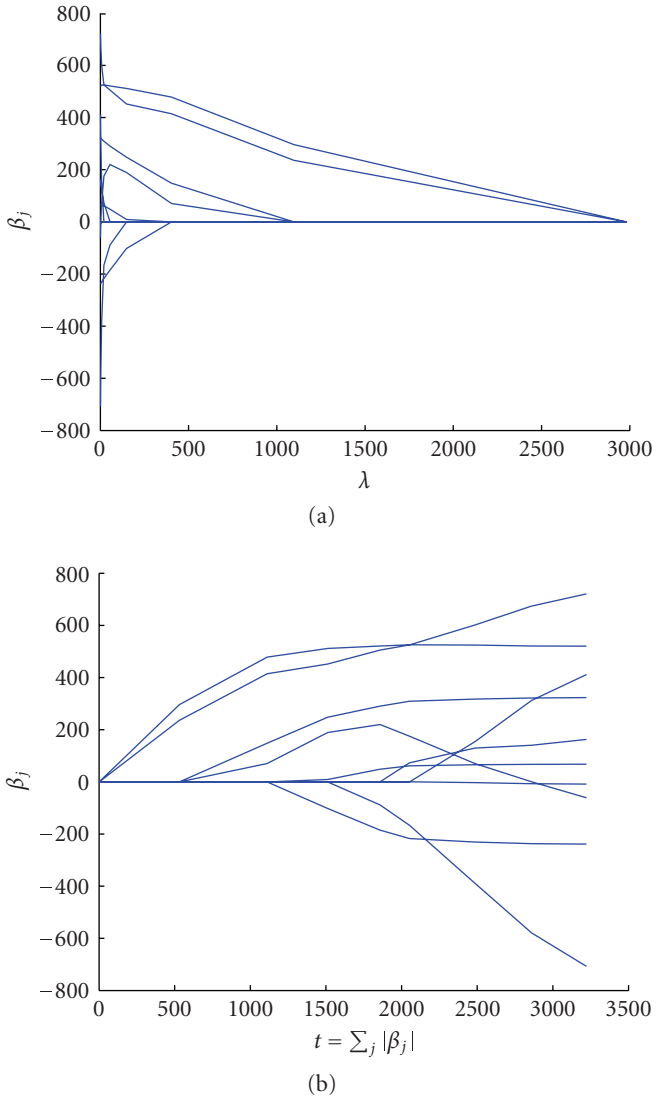

FIGURE 4.1. Variable selection by LASSO regression ($q = 1$) for the diabetes data. The y-axis for both panels represents $\widehat{\beta}_j$, the x-axis for the up panel is λ and for the lower panel is the sum $t = \sum_j |\widehat{\beta}_j|$. Both panels show the trajectories of regression coefficients as a result of changing tuning parameters.

In Figure 4.1, the upper panel shows the trajectories of $\widehat{\beta}_j$ as the penalty λ increases from 0 to ∞. The increase of λ shrinks the regression coefficients towards zero, and thus it operates as variable selection. From the magnitude of regression coefficients at any value of λ, relative influence of the covariates is clear. For comparison, we also plot the trajectories of $\widehat{\beta}_j$'s as $t = \sum_j |\beta_j|$ increases 0 to ∞, shown in the lower plot of Figure 4.1.

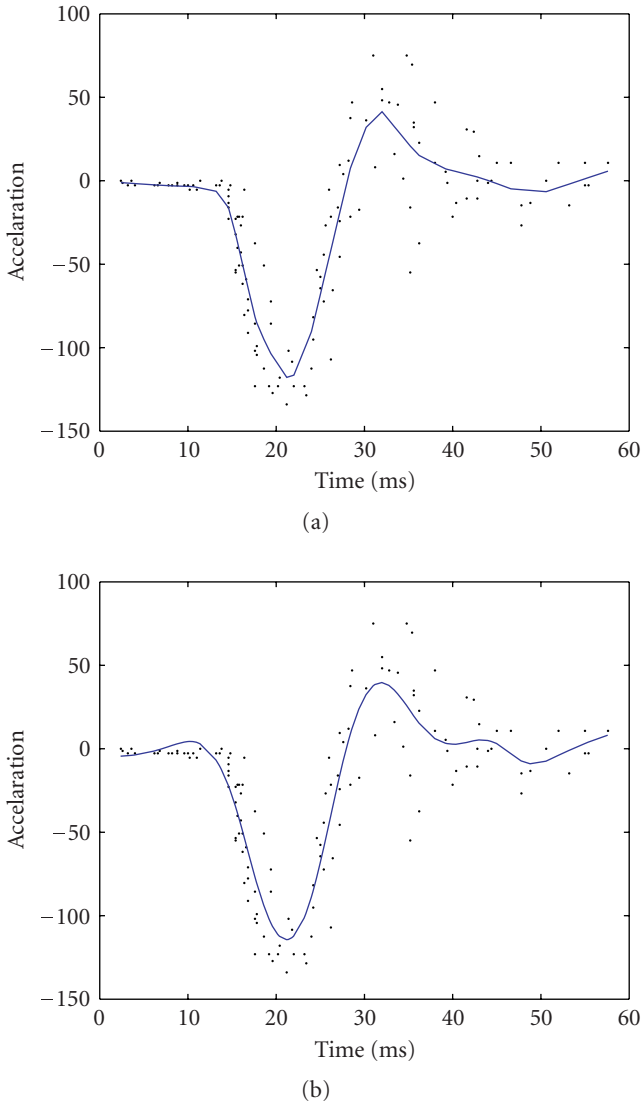

FIGURE 4.2. Penalized regression spline fits ($\hat{m}(x)$) to the motorcycle impact data. The y-axis is acceleration, and the x-axis is time (milliseconds). The two panels show the results with $q = 1$ and $q = 2$, respectively. The "kink" in this dataset around 14 milliseconds is better captured by the regression with $q = 1$.

They clearly show the variable selection and shrinkage as a result of the L_1-penalty function. The pattern shown in the figure is very similar to the one in LARS (Efron et al. [8]). According to the plot, there is occasionally some β that first goes positive and then goes negative (or inversely). This situation has also occurred in Efron et al. [8], though an intuitively more reasonable pattern of the coefficients would be from zero to either positive

or negative monotonously. Implementing our new method, we found the computation is really fast than otherwise.

4.2. Penalized regression spline with motorcycle data. In what follows, we implement our method in arbitrary case of q's on a simulated motocycle crash dataset (Silverman [19]), where y = acceleration and x = time (in milliseconds) are dependent and independent variables, respectively. There is obvious nonlinearity in the scatter plot of y and x. Ruppert and Carroll [5] modeled their relationship nonparametrically by a P-spline. The unknown mean function $m(\cdot)$ is modeled as

$$m(x) = \beta_0 + \beta_1 x + \beta_2 x^2 + \cdots + \beta_p x^p + \sum_{k=1}^{K} \beta_{p+k} (x - \kappa_k)_+^q, \tag{4.1}$$

where p is the degree of the spline, $\{\kappa_k\}_{k=1}^K$ are spline knots, and $u_+ = uI(u > 0)$ ($I(\cdot)$ is the indicator function). Model (4.1) uses the so-called truncated power function basis $\{1, x, \ldots, x^p, (x - \kappa_1)_+^p, \ldots, (x - \kappa_K)_+^p\}$. This basis is convenient for illustration purpose. Other bases, for example, B-splines, can be used.

We minimize

$$n^{-1} \sum_{i=1}^{n} \{y_i - m(x)\}^2 + \lambda \sum_{k=1}^{K} |\beta_{p+k}|^q, \tag{4.2}$$

where λ_i, $i = 0, 1, \ldots, p+1$, are set to zero, and $\lambda_i, i = p+1, \ldots, p+K$, are set equal to λ. According to Ruppert and Carroll [5], the number and location of knots are not crucial once the number of knots exceeds a minimum of $\min(n/4, 40)$, for example, and possible overfitting can be governed by the smoothing parameter λ. A 40-knot P-spline is fit to the data. The conclusions from the fit are similar to Ruppert and Carroll [5]; for example, nonquadratic penalties and linear spline best accommodate the apparent change point in the data. Figure 4.2 shows a typical regression fit for using linear splines ($p = 1$) on the full motorcycle dataset (133 data points). The two panels show the results with $q = 1$ and $q = 2$, respectively. The "kink" in this dataset around 14 milliseconds is better captured by the regression with $q = 1$. During the above model fitting process, we estimate many alternative models with different q's and p's with the algorithms developed in this paper and the computational speed is very satisfactory.

5. Conclusions

This paper mainly suggests a new alternative method for solving the generic regression model with L_q-penalty functions. For L_1 penalty case, the problem is proved to be polynomial time solvable. The approach is extended to the L_q cases, recasting the problem into a smooth optimization problem, while preserving the convexity properties of the original problem. When $q \geq 1$, efficient algorithm is available. The theory has immediate applications to solve statistical problems. We implement our method in both the LASSO problem and the penalized spline smoothing problem, and numerical results show substantial promise of the proposed method.

6. Mathematical proofs

Proof of Lemma 2.2. The function $\vec{\beta} = f(\vec{z}, \vec{\bar{z}}) = \vec{z} + \vec{\bar{z}}$ is a bijection between the feasible space of $(\vec{z}, \vec{\bar{z}})$ defined by (2.3) and the feasible space of $\vec{\beta}$ defined by (2.1). Given a value of $(\vec{z}, \vec{\bar{z}})$, the function gives a unique $\vec{\beta}$. Given a $\vec{\beta}$, there is a unique value in the z space, given by ($\vec{z} = \max(\beta, 0)$, $\vec{\bar{z}} = \min(\beta, 0)$), which satisfies the constraints (2.3).

(i) According to the definition, $(\overrightarrow{z^{*\prime}})_i$ is the positive part of $(\vec{z^*} + \vec{\bar{z}^*})_i$, and $(\overrightarrow{\bar{z}^{*\prime}})_i$ is the negative part. Therefore, at least one of them has to be zero, and thus $(\overrightarrow{z^{*\prime}}, \overrightarrow{\bar{z}^{*\prime}})$ satisfies the constraint $z_i \cdot \bar{z}_i = 0$ for all $i = 1, \ldots, m$.

(ii) Next we prove that $(\overrightarrow{z^{*\prime}}, \overrightarrow{\bar{z}^{*\prime}})$ is an optimal solution of (2.4). According to the definition of $(\overrightarrow{z^{*\prime}}, \overrightarrow{\bar{z}^{*\prime}})$, the following holds:

$$\overrightarrow{z^{*\prime}} + \overrightarrow{\bar{z}^{*\prime}} = \vec{z^*} + \vec{\bar{z}^*}. \tag{6.1}$$

As $\vec{z^*} \geq 0$ and $\vec{\bar{z}^*} \leq 0$, we have $\overrightarrow{z^{*\prime}} = (\vec{z^*} + \vec{\bar{z}^*})^+ \leq \vec{z^*}$ and $\overrightarrow{\bar{z}^{*\prime}} = (\vec{z^*} + \vec{\bar{z}^*})^- \geq \vec{\bar{z}^*}$. Consequently,

$$\mathbf{0} \leq \widetilde{\mathbf{z}^{*\prime}} - \widetilde{\bar{\mathbf{z}}^{*\prime}} \leq \widetilde{\mathbf{z}^*} - \widetilde{\bar{\mathbf{z}}^*}. \tag{6.2}$$

Since $\lambda_i \geq 0$ and $q \geq 0$, we have

$$\left|\left|\mathbf{Y} - \mathbf{X}\left(\overrightarrow{z^{*\prime}} + \overrightarrow{\bar{z}^{*\prime}}\right)\right|\right|_2^2 + \sum_{i=1}^{m} \lambda_i \left(z_i^{*\prime} - \bar{z}_i^{*\prime}\right)^q \leq \left|\left|\mathbf{Y} - \mathbf{X}\left(\vec{z^*} + \vec{\bar{z}^*}\right)\right|\right|_2^2 + \sum_{i=1}^{m} \lambda_i \left(z_i^* - \bar{z}_i^*\right)^q. \tag{6.3}$$

Because $(\vec{z^*}, \vec{\bar{z}^*})$ is an *optimal* solution (2.4), the equality of the above inequality has to hold. This proves that $(\overrightarrow{z^{*\prime}}, \overrightarrow{\bar{z}^{*\prime}})$ is also an optimal solution of (2.4). □

Proof of Corollary 2.3. (i) Suppose $(\vec{z^*}, \vec{\bar{z}^*})$ is an optimal solution of (2.4). We first show that it is also a feasible solution of (2.3), that is, $(\vec{z^*}, \vec{\bar{z}^*})$ also satisfies the constraint $z_i \cdot \bar{z}_i = 0 \ \forall i = 1, \ldots, m$. If otherwise, then the solution $(\overrightarrow{z^{*\prime}}, \overrightarrow{\bar{z}^{*\prime}}) = ((\vec{z^*} + \vec{\bar{z}^*})^+, (\vec{z^*} + \vec{\bar{z}^*})^-)$ is different from $(\vec{z^*}, \vec{\bar{z}^*})$. When $\lambda_i > 0$ $i = 1, \ldots, n$, $(\overrightarrow{z^{*\prime}}, \overrightarrow{\bar{z}^{*\prime}})$ is a strictly better solution than $(\vec{z^*}, \vec{\bar{z}^*})$ following the same reasoning as in the proof of Lemma 2.2, that is,

$$\left|\left|\mathbf{Y} - \mathbf{X}\left(\overrightarrow{z^{*\prime}} + \overrightarrow{\bar{z}^{*\prime}}\right)\right|\right|_2^2 + \sum_{i=1}^{m} \lambda_i \cdot \left(z^{*\prime}{}_i - \bar{z}^{*\prime}{}_i\right)^q < \left|\left|\mathbf{Y} - \mathbf{X}\left(\vec{z^*} + \vec{\bar{z}^*}\right)\right|\right|_2^2 + \sum_{i=1}^{m} \lambda_i \cdot \left(z_i^* - \bar{z}_i^*\right)^q. \tag{6.4}$$

This contradicts the assumption that $(\vec{z^*}, \vec{\bar{z}^*})$ is an optimal solution of (2.4). Therefore $(\vec{z^*}, \vec{\bar{z}^*})$ is a feasible solution of (2.3). Since (2.4) is a relaxation of (2.3), solution $(\vec{z^*}, \vec{\bar{z}^*})$ is an optimal solution of (2.3).

(ii) Suppose $(\overrightarrow{z^{*\prime}}, \overrightarrow{\bar{z}^{*\prime}})$ is an optimal solution of (2.3). We show that it is also an optimal solution of (2.4). Suppose otherwise. Then there is an optimal solution $(\vec{z^*}, \vec{\bar{z}^*})$ of (2.4),

such that

$$\left|\left|\mathbf{Y}-\mathbf{X}(\overrightarrow{z^*}+\overrightarrow{\bar{z}^*})\right|\right|_2^2+\sum_{i=1}^{m}\lambda\cdot(z_i^*-\bar{z}_i^*)^q<\left|\left|\mathbf{Y}-\mathbf{X}(\overrightarrow{z^{*\prime}}+\overrightarrow{\bar{z}^{*\prime}})\right|\right|_2^2+\sum_{i=1}^{m}\lambda_i\cdot(z_i^{*\prime}-\bar{z}_i^{*\prime})^q. \tag{6.5}$$

According to (i), $(\overrightarrow{z^*},\overrightarrow{\bar{z}^*})$ is also a feasible solution of (2.3). The above inequality, however, implies that $(\overrightarrow{z^*},\overrightarrow{\bar{z}^*})$ is a better solution than $(\overrightarrow{z^{*\prime}},\overrightarrow{\bar{z}^{*\prime}})$ for (2.3), contradicting the assumption that $(\overrightarrow{z^{*\prime}},\overrightarrow{\bar{z}^{*\prime}})$ is an optimal solution of (2.3). Therefore $(\overrightarrow{z^{*\prime}},\overrightarrow{\bar{z}^{*\prime}})$ is also an optimal solution of (2.4).

□

Proof of Lemma 2.4. When $\lambda_i \geq 0$ and $q \geq 1$, the function $\sum_{i=1}^{m}\lambda_i x_i^q, x_i \in \mathbb{R}$, is convex. As convexity is preserved under function composition with affine functions, the penalty term $\sum_{i=1}^{m}\lambda_i(z_i-\bar{z}_i)^q$ is still convex. The loss function is a square function, which is also convex. Therefore, the summation of them, $\|\mathbf{Y}-\mathbf{X}(\overrightarrow{z}+\overrightarrow{\bar{z}})\|_2^2+\sum_{i=1}^{m}\lambda_i(z_i-\bar{z}_i)^q$, is convex. □

Proof of Theorem 2.5. The problem (2.5) is polynomially transformed to the problem (2.6), which is equivalent to (2.5) (according to Lemma 2.2). Furthermore, it is easy to see that (2.6) is a positive semidefinite quadratic program with a polynomial number of bound constraints. Minimizing such a program to optimality is known to be a $\mathcal{P}$ problem (see, e.g., Chung and Murty [13]). Therefore, the original equivalent problem (2.5) is also a $\mathcal{P}$ problem. □

References

[1] A. E. Hoerl and R. W. Kennard, "Ridge regression: biased estimation for nonorthogonal problems," *Technometrics*, vol. 12, no. 1, pp. 55–67, 1970.

[2] I. Frank and J. Friedman, "A statistical view of some chemometrics regression tools," *Technometrics*, vol. 35, no. 2, pp. 109–135, 1993.

[3] R. Tibshirani, "Regression shrinkage and selection via the LASSO," *Journal of the Royal Statistical Society. Series B*, vol. 58, no. 1, pp. 267–288, 1996.

[4] W. J. Fu, "Penalized regressions: the bridge versus the lasso," *Journal of Computational and Graphical Statistics*, vol. 7, no. 3, pp. 397–416, 1998.

[5] D. Ruppert and R. J. Carroll, "Penalized regression splines," preprint, 1997, http://legacy.orie.cornell.edu/~davidr/papers/srsrev02.pdf.

[6] M. R. Osborne, B. Presnell, and B. A. Turlach, "On the LASSO and its dual," *Journal of Computational and Graphical Statistics*, vol. 9, no. 2, pp. 319–337, 2000.

[7] M. R. Osborne, B. Presnell, and B. A. Turlach, "A new approach to variable selection in least squares problems," *IMA Journal of Numerical Analysis*, vol. 20, no. 3, pp. 389–403, 2000.

[8] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least angle regression," *The Annals of Statistics*, vol. 32, no. 2, pp. 407–499, 2004.

[9] J. J. Moré and D. C. Sorensen, "Computing a trust region step," *SIAM Journal on Scientific and Statistical Computing*, vol. 4, no. 3, pp. 553–572, 1983.

[10] A. R. Conn, N. I. M. Gould, and P. L. Toint, *Trust-Region Methods*, MPS/SIAM Series on Optimization, SIAM, Philadelphia, Pa, USA, 2000.

[11] T. F. Coleman and Y. Li, "An interior trust region approach for nonlinear minimization subject to bounds," *SIAM Journal on Optimization*, vol. 6, no. 2, pp. 418–445, 1996.

[12] M. K. Kozlov, S. P. Tarasov, and L. G. Hačijan, "Polynomial solvability of convex quadratic programming," *Soviet Mathematics Doklady*, vol. 20, pp. 1108–1111, 1979.
[13] S. J. Chung and K. G. Murty, "Polynomially bounded ellipsoid algorithms for convex quadratic programming," in *Nonlinear Programming (Madison, Wis, USA, 1980)*, vol. 4, pp. 439–485, Academic Press, New York, NY, USA, 1981.
[14] Y. Ye and E. Tse, "An extension of Karmarkar's projective algorithm for convex quadratic programming," *Mathematical Programming*, vol. 44, no. 2, pp. 157–179, 1989.
[15] C. Lin and J. J. More, "Newton's method for large bound-constrained optimization problems," Tech. Rep. ANL/MCS-P724-0898, Mathematical and Computer Sciences Division, Argonne National Laboratories, Argonne, Ill, USA, 1998.
[16] J. J. Moré and G. Toraldo, "Algorithms for bound constrained quadratic programming problems," *Numerische Mathematik*, vol. 55, no. 4, pp. 377–400, 1989.
[17] T. F. Coleman and Y. Li, "A reflective Newton method for minimizing a quadratic function subject to bounds on some of the variables," *SIAM Journal on Optimization*, vol. 6, no. 4, pp. 1040–1058, 1996.
[18] P. J. Green and B. W. Silverman, *Nonparametric Regression and Generalized Linear Models*, vol. 58 of *Monographs on Statistics and Applied Probability*, Chapman & Hall, London, UK, 1994.
[19] B. W. Silverman, "Some aspects of the spline smoothing approach to nonparametric regression curve fitting," *Journal of the Royal Statistical Society. Series B*, vol. 47, no. 1, pp. 1–52, 1985.

Tracy Zhou Wu: JPMorgan Chase Bank, 1111 Polaris Pkwy, Columbus, OH 43240, USA;
Department of Quantitative Analysis and Operations Management, University of Cincinnati,
P.O. Box 210130, Cincinnati, OH 45221, USA
Email address: tracy.z.wu@jpmchase.com

Yingyi Chu: ABN AMRO Bank, 250 Bishopsgate, London EC2M 4AA, UK
Email address: yingyi.chu@uk.abnamro.com

Yan Yu: Department of Quantitative Analysis and Operations Management, University of Cincinnati,
P.O. Box 210130, Cincinnati, OH 45221, USA
Email address: yan.yu@uc.edu