Introduction to the LASSO

A Convex Optimization Approach for High-dimensional Problems

Niharika Gauraha

The term 'high-dimensional' refers to the case where the number of unknown parameters to be estimated, p, is of much larger order than the number of observations, n, that is $p \gg n$. Since traditional statistical methods assume many observations and a few unknown variables, they can not cope up with the situations when $p \gg n$. In this article, we study a statistical method, called the 'Least Absolute Shrinkage and Selection Operator' (LASSO), that has got much attention in solving high-dimensional problems. In particular, we consider the LASSO for high-dimensional linear regression models. We aim to provide an introduction of the LASSO method as a constrained quadratic programming problem, and we discuss the convex optimization based approach to solve the LASSO problem. We also illustrate applications of LASSO method using a simulated and a real data examples.

Niharika Gauraha is a PhD student at Indian Statistical Institute, Bangalore. Her research interests include statistical pattern recognition and machine learning. Currently she is working as a researcher at the Department of Pharmaceutical Biosciences, Uppsala University.

Keywords
LASSO, high-dimensional statistics, regularized regression, least squares regression, variable selection.

1. Introduction and Motivation

In order to build an intuition about high dimensional problems, and the limitations and difficulties associated with it, we start with the simplest case where observations are noiseless. We consider a linear model as:

$$\mathbf{Y} = \mathbf{X}\beta^0, \tag{1}$$

where $\mathbf{Y} \in \mathbb{R}^n$ is a response vector, $\mathbf{X} \in \mathbb{R}^{n \times p}$ is a design matrix and $\beta^0 \in \mathbb{R}^p$ is a vector of unknown true regression coefficients. For $p > n$, the problem (1) is an underdetermined linear system and there is no unique solution; in fact, there is an infinite set of solutions. Thus, it is impossible to identify the correct solution from the infinite solution set without some additional information or constraints. So, to simplify things we assume a set of

The sparsity assumption for an unknown vector **v** means that it has a relatively small number of non-zero elements.

constraints and then we pick one out of the many solutions that satisfies those constraints. For example, sparsity assumption for the true β^0 is a constraint that it is supposed to find a solution to the linear equation (1), that has the fewest number of non-zero entries in β^0. This problem can be mathematically described as ℓ_0-norm constrained optimization problem as:

$$\underset{\beta \in \mathbb{R}^p}{minimize} \ \|\beta\|_0 \text{ such that } \mathbf{Y} = \mathbf{X}\beta. \tag{2}$$

The problem (2) is equivalent to the best subset selection. When p is large, an exhaustive search of the best subset is computationally infeasible, because it requires considering all $\binom{p}{s}$ models (where $s \leq n$). Since the optimization problem (2) is non-convex and combinatorial in nature, we consider the nearest convex problem, which is ℓ_1-norm constrained convex optimization problem given as:

$$\underset{\beta \in \mathbb{R}^p}{minimize} \ \|\beta\|_1 \text{ such that } \mathbf{Y} = \mathbf{X}\beta. \tag{3}$$

The optimization problem (3) is known as the Basis Pursuit Linear Program (BPLP), see chapter 4 of [1] for further information. A number of efficient algorithms have been developed for solving such convex optimization problems (we will discuss in more detail later). In the following, we consider a simple numerical example to illustrate about high dimensional problem and to show that under sparsity assumption it is possible to solve the underdetermined system of linear equations.

$$\begin{bmatrix} 2 \\ 2 \end{bmatrix} = \begin{bmatrix} 2 & 1 & 0 \\ 2 & 0 & 1 \end{bmatrix} \begin{bmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \end{bmatrix} \tag{4}$$

The problem (4) is an under determined equation system with two equations and three variables $\beta = (\beta_1, \beta_2, \beta_3)^T$. Let us assume that it has a sparse solution. In other words, the number of non-zero components of β is less than or equal to two. In order to find the candidate solutions, we have to solve the equation system (4) by setting the subset of its components to zero. Here are some

candidate solutions to (4):

$$[\beta_1 = 1,\ \beta_2 = 0,\ \beta_3 = 0], \quad \|\beta\|_0 = 1, \quad \|\beta\|_1 = 1$$
$$[\beta_1 = 0,\ \beta_2 = 2,\ \beta_3 = 2], \quad \|\beta\|_0 = 2, \quad \|\beta\|_1 = 4.$$

The sparsest solution is $[\beta_1 = 1,\ \beta_2 = 0,\ \beta_3 = 0]$, and the solution of the basis pursuit is the same as the sparsest solution (but it may not be the case always).

In order to build an idea about how shrinking of coefficients helps in finding a stable solution, we perform a small simulation study for the linear regression model. We consider the usual linear regression model:

$$\mathbf{Y} = \mathbf{X}\beta^0 + \epsilon, \tag{5}$$

with response vector $\mathbf{Y}_{n\times 1}$, design matrix $\mathbf{X}_{n\times p}$, true underlying coefficient vector $\beta^0_{p\times 1}$ and error vector $\epsilon_{n\times 1}$. To simulate data $(\mathbf{Y}, \mathbf{X})$ for the linear regression model (5), we consider the following setup.

Data Simulation Setup

- $p = 20$, $n = 50$ and the elements of the design matrix, X_{ij}, are generated IID from $\mathcal{N}(0, 1)$ once and then kept fixed.
- $\epsilon_{n\times 1} \sim N_n(0, I_n)$.
- $\beta^0 = \{\underbrace{1, ..., 1}_{5}, \underbrace{.01, ..., .01}_{5}, \underbrace{0, ..., 0}_{10}\}$.

The ordinary least squares (OLS) estimator $\hat{\beta}_{OLS}$ can be computed using the following equation:

$$\hat{\beta}_{OLS} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}.$$

We recall that the expected prediction error of the OLS estimator $\hat{\beta}_{OLS}$, is given by (see section 3.2 of [2] for the detailed derivation) :

$$\mathbb{E}\left[(\mathbf{Y} - \mathbf{X}\hat{\beta}_{OLS})^2\right] = \sigma^2 + \frac{\sigma^2}{n}p.$$

Table 1. Simulation results of the OLS and srinkage estimators.

Method	Bias	Variance	Prediction Error
OLS	0.004	0.4	1.430
OLS-.01	0.004	0.368	1.404
OLS-.05	0.016	0.284	1.327
OLS-.1	**0.051**	**0.218**	**1.289**
OLS-.2	0.192	0.155	1.356
OLS-.3	0.426	0.137	1.565

Where the first term σ^2 is the irreducible error and the second term $\frac{\sigma^2}{n}p$ corresponds to the variance of the OLS estimate $\hat{f} = \mathbf{X}\hat{\beta}_{OLS}$. We note that each component $(\hat{\beta}_{OLS})_j$ contributes equal variance $\frac{\sigma^2}{n}$, regardless of whether the true coefficient is large or small (or zero). In our simulation example, to compute the expected prediction error, bias and variance of the OLS fit $\hat{f}(\mathbf{X}) = \mathbf{X}\hat{\beta}_{OLS}$, we repeat the following steps 100 times.

Simulation Steps

1. Generate an error vector as $\epsilon_{n\times 1} \sim N_n(0, I_n)$ and then compute a response vector as $\mathbf{Y} = \mathbf{X}\beta^0 + \epsilon$.
2. Compute the OLS fit $\mathbf{X}\hat{\beta}_{OLS} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}$.
3. Generate an error vector as $\epsilon_{n\times 1} \sim N(0, I_n)$ and then compute a new response vector as $\mathbf{Y}_{test} = \mathbf{X}\beta^0 + \epsilon$.
4. Compute the prediction error as $PE = \frac{1}{n}\|\mathbf{Y}_{test} - \mathbf{X}\hat{\beta}_{OLS}\|_2^2$.

The expected prediction error (EPE) is computed by taking the average over prediction error of 100 simulations. We also compute the bias and variance of the OLS fit over 100 simulations. These results are reported in the first row of the *Table* 1. Now, we try to remove the variability associated with the small coefficients by shrinking them towards zero. In our previous example, we shrink or soft-threshold (to be defined later) each coefficient of the OLS estimator by an amount λ, and we denote the new estimator as $\hat{\beta}_{OLS} - \lambda$. We use the different amount of shrinkage as $\lambda = .05, .1, .2, .3, .4$ and observe the changes in bias, variance, and prediction error (over 100 simulations), see rows 2, 3, 4, 5 and 6 of *Table* 1. From *Table* 1, it is clear that shrinkage reduces the

variance of the fit but increases its bias. We notice that when we shrink all coefficients by $\lambda = .1$, we get the minimum prediction error. Thus, the right amount of shrinkage can provide a more stable solution (at the risk of introducing little bias). We refer to section 2.9 of [2] and section 2.1.3 of [3] for more details on model selection and bias-variance trade-off.

Next, we define the Least Absolute Shrinkage and Selection Operator (LASSO), which is based on the following key concepts:
(i) ℓ_1 regularization approximates ℓ_0 regularization (best subset selection).
(ii) Shrinkage (if done properly) helps to improve prediction performance.

The LASSO, introduced by [4] is a penalized least squares technique which puts ℓ_1 constraint on the estimated regression coefficients. The LASSO estimator, $\hat{\beta}$, for the linear regression model (5) is given as follows:

The least absolute shrinkage and selection operator was introduced by Robert Tibshirani in 1996 based on Leo Breiman's non-negative garrote.

$$\hat{\beta} := \hat{\beta}(\lambda) = \underset{\beta \in \mathbb{R}^p}{\arg\min} \left\{ \frac{1}{n} \|\mathbf{Y} - \mathbf{X}\beta\|_2^2 + \lambda \, \|\beta\|_1 \right\}, \qquad (6)$$

where $\lambda \geq 0$ is the regularization parameter that controls the amount of shrinkage. Due to the geometry of the ℓ_1-norm penalty the LASSO shrinks some of the regression coefficients to exactly zero (to elaborate later). Thus it serves as a variable selection method also.

We have organized the rest of the article in the following manner. In section 2, we briefly review convexity and convex optimization theory, least squares regression, and variable selection problem. In section 3, we explain the LASSO method, we derive its closed form solution for single variable and orthonormal design case. We discuss iterative method for computation of the LASSO solution, in section 4. Section 5 is concerned with the applications of the LASSO. Section 5 gives computational details. We shall provide some concluding remarks in section 6.

2. Notations and Background

In this section, we state the notations and assumptions, and recall some preliminary results. We consider the usual linear regression set up as given in (5). We assume that the components of the noise vector $\epsilon \in \mathbb{R}^n$ are independent and identically distributed (IID) $N(0, \sigma^2)$. We use subscripts to denote the columns of $\mathbf{X}$, i.e., $\mathbf{X}_j$ denotes the jth column. We also assume that the design matrix $\mathbf{X}$ is fixed, the data is centered, and the predictors are standardized, so that we have:

$$\sum_{i=1}^{n} \mathbf{Y}_i = 0, \quad \sum_{i=1}^{n} (\mathbf{X}_j)_i = 0 \text{ and } \frac{1}{n}\mathbf{X}_j^T \mathbf{X}_j = 1 \text{ for all } j = 1, ..., p.$$

The ℓ_0-norm, ℓ_1-norm, and ℓ_2-norm are defined as:

$$\|\beta\|_0 = \sum_{j=1}^{p} I(\beta_j \neq 0) \tag{7}$$

$$\|\beta\|_1 = \sum_{j=1}^{p} |\beta_j| \tag{8}$$

$$\|\beta\|_2^2 = \sum_{j=1}^{p} \beta_j^2. \tag{9}$$

The soft thresholding is also called wavelet shrinkage, as values for both positive and negative coefficients are being 'shrinked' towards zero.

The soft-thresholding operator is defined as follows, and it is illustrated in *Figure* 1, where the straight line (in blue) is soft-thresholded by the quantity $\lambda = 1$ (in red).

$$\mathbb{S}_\lambda(x) = \begin{cases} x + \lambda & \text{if } x < -\lambda \\ 0 & \text{if } |x| \leq \lambda \\ x - \lambda & \text{if } x > \lambda \end{cases} \tag{10}$$

2.1 *Background on Convexity*

In this section, we review some background and some useful theorems concerning convexity and convex optimization theory. For more details on convex optimization theory we refer to the chapters 1 and 2 of [5] and chapters 2, 3, 4 and 5 of [6].

Definition 1 (Affine Functions) *An affine function is a function composed of a sum of a constant and a linear function, given as*

$$f(x) = Ax + b,$$

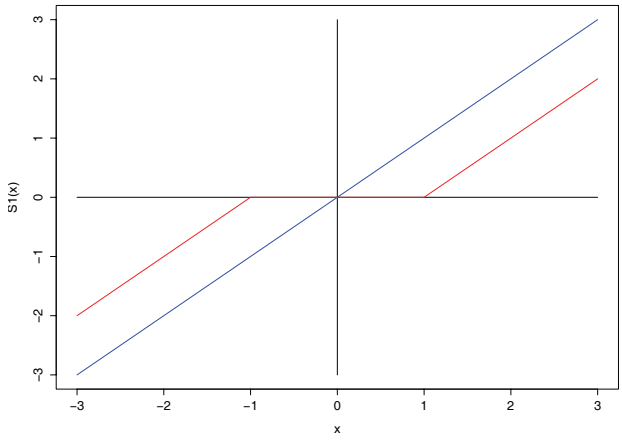


Figure 1. An illustration of the soft-thresholding function.

for some matrix A and vector b of appropriate dimensions. It can be also viewed as a linear transformation followed by a translation.

Definition 2 (Convex Sets) *A set C is said to be convex, if it contains the line segments between any two of its points, that is:*

$$\lambda x + (1-\lambda)y \in C, \quad \text{for all } x, y \in C, \text{ and for all } \lambda \in [0,1]. \tag{11}$$

For the following definitions we assume an objective function, $f(x)$, defined as $f : \mathbb{R}^p \to \mathbb{R}$, and its domain is denoted by $D(f)$.

Definition 3 (Convex Functions) *A function f is convex if its domain, $D(f)$, is a convex set and the following holds:*

$$f(\lambda x + (1-\lambda)y) \leq \lambda f(x) + (1-\lambda) f(y), \text{ for all } x, y \in D(f), \text{ for all } \lambda \in [0,1]. \tag{12}$$

If the above definition (12) holds with strict inequality for $x \neq y$ and for all $\lambda \in (0,1)$, then f is strictly convex.

Definition 4 (Sub-level Sets) *The α-sublevel set of a function f, is the set of all points x such that $f(x) \leq \alpha$, where $\alpha \in \mathbb{R}$.*

Definition 5 (First Order Condition for Convexity) *If a function f is differentiable, then f is convex if and only if, $D(f)$ is a convex set and for all $x, y \in D(f)$ the following holds:*

$$f(y) \geq f(x) + \nabla f(x)^T (y - x). \tag{13}$$

The term $f(x) + \nabla f(x)^T (y - x)$ is the first-order approximation of the function f at the point x. The first order condition implies that f is convex if and only if the tangent line is a global underestimator of the function f. Similarly, if the above condition (13) holds with strict inequality, then f is strictly convex.

Definition 6 (Sub-gradients) *A sub-gradient of a convex function f (f may not be differentiable) at x is any $g \in \mathbb{R}^p$ such that the following holds:*

$$f(y) \geq f(x) + g^T (y - x), \text{ for all } y. \tag{14}$$

For example: The absolute function $f(x) = |x|$ is not differentiable at $x = 0$. For $x > 0$, sub-gradient $g = +1$, for $x < 0$, sub-gradient $g = -1$ and at $x = 0$, sub-gradient g is any element of $[-1, 1]$.

Definition 7 (Sub-differentials) *Set of all sub-gradients of a convex function f at x is called the sub-differential of f at x, and it is denoted as:*

$$\delta f(x) = \{ g \in \mathbb{R}^p : \ g \text{ is a sub-gradient of } f \text{ at } x \}$$

For example: sub-differential of $f = |x|$ at x, is $\delta f(x) = sign(x)$, where the sign function is defined as follows.

$$sign(x) = \begin{cases} -1 & \text{if } x < 0 \\ [-1, 1] & \text{if } x = 0 \\ 1 & \text{if } x > 0 \end{cases} \tag{15}$$

The absolute function $f = |x|$ and its sub-differential $\delta f(x) = sign(x)$ is illustrated in *Figure* 2.

We note that for a convex and differentiable function f, $\delta f(x) = \nabla f(x)$.

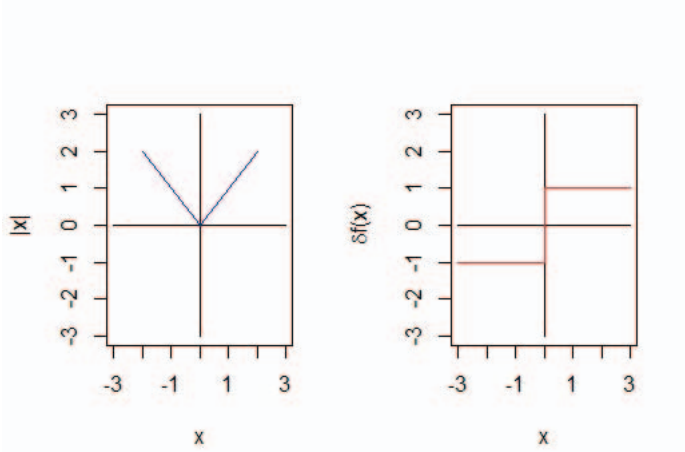


Figure 2. The absolute function and its subdifferential.

Definition 8 (Convex Optimization Problems) *A convex optimization problem is an optimization problem of the form:*

$$\begin{aligned} &\text{minimize } f(x) \\ &\text{subject to } g_i(x) \leq 0, \quad i = 1, ..., m \\ &\qquad\qquad h_i(x) = 0, \quad i = 1, ..., r \end{aligned} \tag{16}$$

where $x \in \mathbb{R}^p$ is the optimization variable, f and g_i are convex functions for all $i = 1, ..., m$ and h_i are affine functions for all $i = 1, ..., r$. If there are no constraints, $m = r = 0$, it is called unconstrained convex optimization problem.

If the objective function, f, and constraint functions are all affine, then it is called a linear programming (LP) problem. If the objective function, f, is convex and quadratic, and the constraint functions are affine, then it is called a quadratic programming(QP) problem.

Definition 9 (Lagrange Duality) *Consider a convex optimization problem as given in the Definition 8. Lagrangian of an optimization problem is defined as augmented objective with a weighted sum of constraints.*

$$L(x, u, v) = f(x) + \sum_{i=1}^{m} u_i g_i(x) + \sum_{i=1}^{r} v_i h_i(x), \tag{17}$$

where $u_i \geq 0$ for all $i = 1, ..., m$ and the vectors ($u \in \mathbb{R}^m, v \in \mathbb{R}^r$) are called dual variables.

We notice that, for every feasible $x \in D(f)$, and for every dual feasible (u, v) the following holds:

$$f(x) \geq L(x, u, v).$$

The Lagrange dual function is defined as the minimum value of the Lagrangian over $x \in \mathbb{R}^p$:

$$l(u, v) = \underset{x \in \mathbb{R}^p}{minimize}\, L(x, u, v). \tag{18}$$

Let $x^\star$ be a primal optimal solution and $f(x^\star)$ be the primal optimal value, then minimizing $L(x, u, v)$ over all x gives a lower bound for $f(x^\star)$:

$$f(x^\star) \geq \underset{x \in D(f)}{minimize}\, L(x, u, v) \geq \underset{x \in \mathbb{R}^p}{minimize}\, L(x, u, v) = l(u, v)$$

We get the best lower bound when the dual function $l(u, v)$ is maximized over (u, v), which is also known as the dual problem:

$$\begin{aligned} &\underset{u \in \mathbb{R}^m, v \in \mathbb{R}^r}{maximize}\ \ l(u, v) \\ &\text{subject to } u_i \geq 0, \text{ for all } i = 1, ..., m \end{aligned}$$

The dual problem is always convex even if the corresponding primal problem is not convex, for example, $l(u, v)$ is convex, since it is affine in (u, v).

Definition 10 (Weak and Strong Duality) *If $x^\star$ is a primal optimal solution and $(u^\star, v^\star)$ is the a dual optimal solution then weak duality always holds, that is:*

$$f(x^\star) \geq l(u^\star, v^\star).$$

Strong duality holds when the primal and dual optimal values coincide:

$$f(x^\star) = l(u^\star, v^\star).$$

Definition 11 (Slater's Condition) *For a convex primal problem, if there exists an $x \in \mathbb{R}^p$ such that $g_1(x) < 0, ..., g_m(x) < 0$ and $h_1(x) = 0, ..., h_r(x) = 0$ then strong duality holds.*

Definition 12 (KKT Condition) *Suppose $f(x)$ is a primal optimization problem, $L(x, u, v)$ is the corresponding Lagrangian and $l(u, v)$ is the dual problem as defined previously. Let $x^\star$ be a primal optimal solution and $(u^\star, v^\star)$ be a dual optimal solution then the Karush Kuhn Tucker (KKT) conditions are defined as follows.*

- *Stationarity Condition: Sub-differential of $L(x, u, v)$ at $(x^\star, u^\star, v^\star)$ must contain* 0

$$0 \in \delta f(x^\star) + \sum_{i=1}^{m} u_i^\star \ \delta g_i(x^\star) + \sum_{i=1}^{r} v_i^\star \ \delta h_i(x^\star)$$

- *Complementary Slackness*

$$u_i^\star . g_i(x^\star) = 0, \quad \text{for all } i = 1, ..., m$$

- *Primal Feasibility Condition*

$$g_i(x^\star) \leq 0, \quad \text{for all } i = 1, ..., m$$
$$\text{and } h_i(x^\star) = 0, \quad \text{for all } i = 1, ..., r$$

- *Dual Feasibility Condition*

$$u_i^\star \geq 0, \quad \text{for all } i = 1, ..., m$$

KKT conditions are necessary to find an optimum solution but are not necessarily sufficient. However, for convex optimization problems that satisfy Slater's condition, KKT conditions are also sufficient for finding an optimal solution, for proof we refer to section 5.5.3 of [5].

For convex optimization problems that satisfy Slater's condition, KKT conditions are also sufficient for finding an optimal solution.

2.2 *Optimization Techniques*

In this section, we discuss a few iterative methods for solving convex optimization problems. First we consider a simple problem, an unconstrained optimization problem as:

$$\text{minimize } f(x), \tag{19}$$

where f is convex and differentiable. Let us assume that $x^\star$ is an optimal point such that $minf(x) = f(x^\star)$. Since f is differentiable and convex, and $x^\star$ is optimal, then from KKT stationarity condition the following must hold:

$$\nabla f(x^\star) = 0. \tag{20}$$

Thus, by solving (20), we also get the solution for the unconstrained optimization problem (19). (20) is a set of p equations in p variables and mostly it can be solved by iterative algorithms.

An iterative algorithm produces a minimizing sequence x^t, $t = 1, ...$ and it is terminated when a predetermined convergence criterion is met. An iterative algorithm is called descent method when $f(x^{k+1}) < f(x^k)$ and the step function x^{t+1} is computed as:

$$x^{t+1} = x^t + s^t \Delta x^t, \tag{21}$$

where Δx is called the step direction and s^t is called the step size. In the following, we briefly study the gradient descent, sub-gradient and coordinate descent methods for solving convex optimization problems. For details on step size and convergence analysis for the gradient methods we refer to chapter 9 of [5].

2.2.1 *Gradient Descent Methods*

When the step direction in (21), is in the opposite direction of the gradient of the objective function, the descent algorithm is called the gradient descent algorithm. Thus, by substituting gradient direction $\Delta x = -\nabla f(x)$ in (21), we get the step function of the gradient descent as:

$$x^{t+1} = x^t - s^t \nabla f(x). \tag{22}$$

2.2.2 *Sub-gradient Method*

The iterative method is called sub-gradient method when the step function is defined as:

$$x^{t+1} = x^t - s^t q(x^t), \tag{23}$$

where $q(x^t)$ is a sub-gradient of f at x^t. It is also defined for the f that is convex but not necessarily differentiable.

2.2.3 *Coordinate Descent Method*

Now, we consider an objective function, f, that is convex but not necessarily differentiable and it can be split into differentiable and non-differentiable components. For example, $f = fd + fc$, where fd is convex and differentiable, and fc is convex but non-differentiable.

When the objective function, f, is differentiable or the non-differentiable component of f, is separable such that $fc(x) = \sum_{i=1}^{p} fc_i(x_i)$, where each fc_i is convex, then the objective function f, can be minimized coordinate-wise. See [7] for how such coordinate-wise minimization converges to the global minimum. For example, the step function for the coordinate-wise sub-gradient method is defined as:

$$\begin{aligned} x_1^{t+1} &= x_1^t - s^t q(x_1^t, ..., x_p^t) \\ x_2^{t+1} &= x_2^t - s^t q(x_1^{t+1}, x_2^t .., x_p^t) \\ &... \\ x_p^{t+1} &= x_p^t - s^t q(x_1^{t+1}, ..., x_{p-1}^{t+1}, x_p^t). \end{aligned}$$

2.3 *Least Squares Regression*

In this section, we define the least squares method for the linear regression problem (5). The ordinary least squares (OLS) technique is the most commonly used method for estimating the unknown parameters in a linear regression model by minimizing the residual some of squares. The ordinary least squares estimator can be viewed as an unconstrained quadratic programming problem:

$$\hat{\beta}_{OLS} := \underset{\beta \in \mathbb{R}^p}{\arg\min} \left\{ \frac{1}{n} \|\mathbf{Y} - \mathbf{X}\beta\|_2^2 \right\}. \tag{24}$$

Assuming the design matrix $\mathbf{X}$ has full column rank (otherwise one may use generalized inverse), when $p \leq n$, the OLS estimator, $\hat{\beta}_{OLS}$, has a closed form solution:

$$\beta_{OLS} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}.$$

The variable selection problem is also referred to as the problem of subset selection, that is to select an optimal subset of predictors for estimation and prediction.

The OLS estimator has well known properties, i.e., Gauss-Markov and Maximum Likelihood etc., for more details we refer to chapter 3 of [2] and chapter 3 of [8].

2.4 *The Variable Selection Problem*

The variable selection problem is also referred to as the problem of subset selection, that is to select an optimal subset of predictors for estimation and prediction. Subset selection is an important issue particularly when p is large and it is believed that many covariates are redundant or irrelevant. In the context of linear regression, the subset selection problem is to select and fit a model of the form:

$$\mathbf{Y} = \mathbf{X}_S \beta_S + \epsilon,$$

where $S \subset \{1, ..., p\}$ is the active set, $\mathbf{X}_S$ is the columns and β_S is the vector of regression coefficients corresponding to subset S. Since the active set S is not known, there is uncertainty about which subset (from 2^p subsets) to use. Some standard methods of subset selection are forward selection, backward elimination, and the combination of the two (i.e., forward selection steps followed by backward elimination steps). There is a large literature on variable selection methods for linear models (see [9] and [10]) and for high dimensional problems (see [11] and [12]).

3. The LASSO

We consider the linear regression model (5) for high dimensional cases, where the number of unknown parameters to be estimated is much higher than the number of observations. For $p > n$, the linear regression model (5), is an ill-posed problem (a problem which may have more than one solution). In order to solve this ill-posed problem, we need to introduce some constraints or regularizations to the estimation process. As mentioned previously, the LASSO is an ℓ_1-regularized regression method. It estimates the regression coefficients by solving the following constrained

least squares problem:

$$\underset{\beta\in\mathbb{R}^p}{minimize}\ \left\{\frac{1}{n}\|\mathbf{Y}-\mathbf{X}\beta\|_2^2\right\} \quad \text{subject to } \|\beta\|_1 \leq t, \qquad (25)$$

where t is a budget on the ℓ_1-norm and the LASSO finds the best fit within this constraint. If t is equal to or greater than the ℓ_1-norm of the OLS estimator, then the LASSO estimator is the same as the OLS estimator. When t is smaller than the ℓ_1-norm of the OLS estimator then the LASSO shrinks the estimated regression coefficients towards zero, and it may set some of the coefficients to exactly equal to zero.

The Lagrange function (penalized regression) corresponding to the constrained regression problem (25) is given by equation (6). It can be shown that there is a one-to-one correspondence between t and λ. In other words, for a given $\lambda \geq 0$, there exists a $t \geq 0$ such that the two problems share the same solution (see [13]). We consider Lagrangian or penalized LASSO problem (6) for the rest of the article. In general, the LASSO lacks in closed form solution because the the objective function is not differentiable. However, it is possible to obtain closed form solutions for the special case of an orthonormal design matrix. Different interpretations (for the different scenarios) of the LASSO solution is discussed as follows.

3.1 *Single Variable Case*

We first illustrate the LASSO solution for simple linear regression, where $p = 1$ and $\mathbf{Y} = \mathbf{X}_1\beta_1 + \epsilon$. The optimization problem is given as:

$$\underset{\beta_1\in\mathbb{R}}{minimize}\left\{\frac{1}{n}\|\mathbf{Y}-\mathbf{X}_1\beta_1\|_2^2 + \lambda\,|\beta_1|\right\}.$$

Suppose $\hat{\beta}_1$ is a solution of the above optimization problem, then from the KKT stationarity condition, the sub-differential must

contain zero.

$$-\frac{2}{n}\mathbf{X}_1^T(\mathbf{Y}-\mathbf{X}_1\hat{\beta}_1)+\lambda\ sign(\hat{\beta}_1)=0,$$
$$\frac{1}{n}\mathbf{X}_1^T(\mathbf{Y}-\mathbf{X}_1\hat{\beta}_1)=\frac{\lambda}{2}\ sign(\hat{\beta}_1).$$

Note that $\frac{1}{n}\mathbf{X}_1^T\mathbf{X}_1 = 1$, as we are assuming predictors are standardized.

$$\hat{\beta}_1=\frac{1}{n}\mathbf{X}_1^T\mathbf{Y}-\frac{\lambda}{2}\ sign(\hat{\beta}_1)$$
$$\hat{\beta}_1=\begin{cases}\frac{1}{n}\mathbf{X}_1^T\mathbf{Y}+\frac{\lambda}{2} & \text{if } \frac{1}{n}\mathbf{X}_1^T\mathbf{Y}<-\frac{\lambda}{2}\\ 0 & \text{if } \frac{1}{n}|\mathbf{X}_1^T\mathbf{Y}|\leq\frac{\lambda}{2}\\ \frac{1}{n}\mathbf{X}_1^T\mathbf{Y}-\frac{\lambda}{2} & \text{if } \frac{1}{n}\mathbf{X}_1^T\mathbf{Y}>\frac{\lambda}{2}\end{cases}$$

which is the same as the term $\frac{\mathbf{X}_1^T\mathbf{Y}}{n}$ soft-thresholded by $\frac{\lambda}{2}$,

$$\hat{\beta}_1=\mathbb{S}_{\frac{\lambda}{2}}(\frac{\mathbf{X}_1^T\mathbf{Y}}{n}). \tag{26}$$

Hence, the LASSO estimator for the single variable case can also be computed by soft-thresholding the OLS estimator by amount $\frac{\lambda}{2}$

$$\hat{\beta}_1=\mathbb{S}_{\frac{\lambda}{2}}(\hat{\beta}_{OLS}),$$

where $\hat{\beta}_{OLS}=\frac{\mathbf{X}_1^T\mathbf{Y}}{n}$.

3.2 *Orthonormal Design Case*

Next, we derive the LASSO estimator for orthonormal design case. Here we assume variables are uncorrelated that implies $\mathbf{X}_i^T\mathbf{X}_j = 0$ for each $i \neq j$ and $\frac{1}{n}\mathbf{X}^T\mathbf{X} = I_p$.

Suppose $\hat{\beta}$ is a solution of the optimization problem given in (6), then from the KKT stationarity condition we get:

$$-\frac{2}{n}\mathbf{X}^T(\mathbf{Y}-\mathbf{X}\hat{\beta})+\lambda\ sign(\hat{\beta})=0,$$
$$\frac{1}{n}\mathbf{X}^T(\mathbf{Y}-\mathbf{X}\hat{\beta})=\frac{\lambda}{2}\ sign(\hat{\beta}).$$

Here $\frac{1}{n}\mathbf{X}^T\mathbf{X} = I_p$. It follows that,

$$\hat{\beta} = \frac{1}{n}\mathbf{X}^T\mathbf{Y} - \frac{\lambda}{2}\, sign(\hat{\beta})$$

$$\hat{\beta}_j = \begin{cases} \frac{1}{n}(\mathbf{X}^T\mathbf{Y})_j + \frac{\lambda}{2} & \text{if } \frac{1}{n}(\mathbf{X}^T\mathbf{Y})_j < -\frac{\lambda}{2} \\ 0 & \text{if } \frac{1}{n}|(\mathbf{X}_1^T\mathbf{Y})_j| \leq \frac{\lambda}{2} \\ \frac{1}{n}(\mathbf{X}^T\mathbf{Y})_j - \frac{\lambda}{2} & \text{if } \frac{1}{n}(\mathbf{X}^T\mathbf{Y})_j > \frac{\lambda}{2} \end{cases}$$

The coefficient $\hat{\beta}_j$ is then computed by soft-thresholding the j^{th} row of $(\hat{\beta}_{OLS})_j = (\frac{1}{n}\mathbf{X}^T\mathbf{Y})_j$, by $\frac{\lambda}{2}$.

3.3 *Multiple Predictors Case*

In the following, we show that in general, the LASSO estimator has no closed form solution. Basically, we try to solve it for one component and we show that our solution for one component is dependent on all other components. Here, we assume that $\mathbf{X}$ has full column rank, therefore $\mathbf{X}^T\mathbf{X}$ is invertible. Let $\mathbf{X}_{-j}$ denotes all the columns except j^{th} column, and similarly β_{-j} denotes the parameter vector except β_j. Suppose that $\hat{\beta}_j$ is a solution of j^{th} component β_j, then from the KKT stationarity condition we get the following:

$$-\frac{2}{n}\mathbf{X}_j^T\left(\mathbf{Y} - \mathbf{X}_{-j}\beta_{-j} - \hat{\beta}_j\mathbf{X}_j\right) + \lambda\, sign(\hat{\beta}_j) = 0.$$

Further simplification lead to the following:

$$-\frac{2}{n}\mathbf{X}_j^T\mathbf{Y} + \frac{2}{n}\mathbf{X}_j^T\mathbf{X}_{-j}\beta_{-j} + 2\hat{\beta}_j\frac{\mathbf{X}_j^T\mathbf{X}_j}{n} + \lambda\, sign(\hat{\beta}_j) = 0.$$

Since $\frac{\mathbf{X}_j^T\mathbf{X}_j}{n} = 1$, we have the following solution for $\hat{\beta}_j$.

$$\hat{\beta}_j = \frac{1}{n}\mathbf{X}_j^T\left(\mathbf{Y} - \mathbf{X}_{-j}\beta_{-j}\right) - \frac{\lambda}{2}\, sign(\hat{\beta}).$$

Now, we notice that the solution of one β_j is dependent upon all the other components $\beta_{i\neq j}$, therefore there is no closed form solution. For the orthonormal design case, the cross term $\mathbf{X}_j^T\mathbf{X}_{-j}$ vanishes due to orthogonality and we have a closed form solution. Though in general, the LASSO has no closed form solution, it can be solved efficiently due to its convex optimization form. We discuss coordinate descent algorithm for the LASSO in section 4.

3.4 *High-dimensional Case*

For the high-dimensional case, the OLS estimator does not make any sense. In fact the optimization problem (24) can not be solved unless we make some assumption. Here, we assume sparsity that is the underlying true model is sparse, and we seek a sparse solution where many components of the vector $\hat{\beta}$ are zero. The LASSO gives sparse solutions. Depending on the amount of regularization, the LASSO sets some of the coefficients to exactly zero. So the LASSO performs estimation as well as variable selection. As we mentioned earlier, though the LASSO lacks a closed form solution in general, it can be solved efficiently due to its convex optimization form. There are various iterative algorithms for computing solution of the LASSO for high-dimensional setting (i.e., see [14], [15], and [16]). We discuss coordinate descent algorithm for the LASSO in section 4.

3.5 *Bayesian Interpretation of the LASSO Method*

The estimated values of regression coefficients through LASSO can also be interpreted as Bayesian maximum a posteriori (MAP) estimate; that is, if we assume IID double-exponential (Laplace) prior on regression coefficients, then the Bayesian MAP estimates are the same as the LASSO estimates. Here, we consider a hierarchical Bayesian model $\mathbf{Y} \sim N_n(\mathbf{X}\beta, \sigma^2 I_n)$ and $\beta_i \sim DoubleExp(\lambda|\beta_i|)$. Without loss of generality we can assume $\sigma^2 = 1$. It is described as follows, see [17] for the complete derivation.

$$\mathbf{Y}|\mathbf{X}, \beta \sim N_n(\mathbf{X}\beta, I_n)$$

$$\beta_1, \beta_2, ..., \beta_p|\lambda \overset{iid}{\sim} \frac{\lambda}{2}\exp(-\lambda|\beta_i|)$$

$$P(\beta|\mathbf{X},\mathbf{Y}, \lambda) \propto P(\mathbf{Y}|\mathbf{X}, \beta)P(\beta|\lambda).$$

Consider the MAP estimation of β under this model.

$$\begin{aligned}\hat{\beta}_{MAP} &= \arg\max_{\beta\in\mathbb{R}^p} \{\log P(\beta|\mathbf{X},\mathbf{Y},\lambda)\} \\ &= \arg\max_{\beta\in\mathbb{R}^p} \{\log [P(\mathbf{Y}|\mathbf{X},\beta)\, P(\beta|\lambda)]\} \\ &= \arg\max_{\beta\in\mathbb{R}^p} \left\{\log \left[P(\mathbf{Y}|\mathbf{X},\beta) \prod_{i=1}^{p} P(\beta_i|\lambda)\right]\right\} \\ &= \arg\max_{\beta\in\mathbb{R}^p} \left\{\log P(\mathbf{Y}|\mathbf{X},\beta) + \sum_{j=1}^{p} \log P\left(\beta_j|\lambda\right)\right\}.\end{aligned}$$

The first term of the RHS can be given as:

$$P(\mathbf{Y}|\mathbf{X},\beta) = \frac{1}{(2\pi)^{(n/2)}} \exp\left(-\frac{1}{2}||\mathbf{Y}-\mathbf{X}\beta||_2^2\right).$$

Taking log and ignoring constant terms we get:

$$\arg\max_{\beta\in\mathbb{R}^p} \{\log P(\mathbf{Y}|\mathbf{X},\beta)\} = \arg\max_{\beta\in\mathbb{R}^p} \left\{-\frac{1}{2}||\mathbf{Y}-\mathbf{X}\beta||_2^2\right\}. \qquad (27)$$

We can simplify the second term as:

$$\arg\max_{\beta\in\mathbb{R}^p} \left\{\sum_{j=1}^{p} \log P\left(\beta_j|\lambda\right)\right\} = -\lambda \sum_{j=1}^{p} |\beta_j| = -\lambda||\beta||_1. \qquad (28)$$

Thus, from equations (27) and (28) we get,

$$\begin{aligned}\hat{\beta}_{MAP} &= \arg\max_{\beta\in\mathbb{R}^p} \left\{-\frac{1}{2}||\mathbf{Y}-\mathbf{X}\beta||_2^2 - \lambda||\beta||_1\right\} \\ &= \arg\min_{\beta\in\mathbb{R}^p} \left\{\frac{1}{2}||\mathbf{Y}-\mathbf{X}\beta||_2^2 + \lambda||\beta||_1\right\} \\ &= \hat{\beta}\left(\frac{2\lambda}{n}\right) \text{ (the LASSO estimator, estimated at } 2\lambda/n \text{).}\end{aligned}$$

3.6 *LASSO as a Variable Selection Method*

The classical variables selection methods (i.e., subset selection method) are not feasible in high dimension case because the number of feature subsets, 2^p, is too large. The LASSO method can

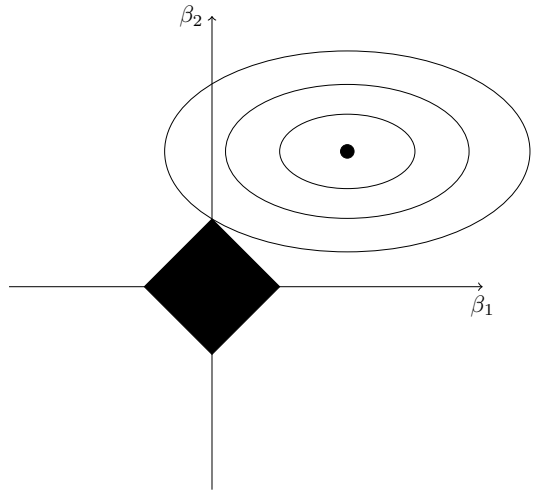


Figure 3. Constrained region for the LASSO.

be used as a variable selection method. The geometric of the ℓ_1-norm penalty of the LASSO leads to variable selection. For example, for $p = 2$ case, the constrained region for the LASSO is a rotated square $|\beta_1| + |\beta_2| \leq t$, as illustrated in *Figure* 3. The plot of the residual sum of squares are ellipses centred at the OLS estimate. The first point where the ellipse touches the rotated square corresponds to the LASSO solution. If that first point is a vertex of the square, then the LASSO solution can have one coefficient equal to zero, $\beta_1 = 0$ in this case. For more details we refer to the original LASSO paper [4] and section 3.4 of [2].

4. Computation of the LASSO Solution

In this section, we study coordinate descent method for computing LASSO solutions.

We note that for a fixed λ, (6) is a quadratic programming problem in parameter β. So for each λ, we have a solution for (6). Since λ controls the amount of regularization, we need a disciplined way of selecting λ, i.e., cross-validation, bootstrapping, etc. If $\lambda = 0$ the LASSO is the same as OLS. As λ increases, the number of non-zero components of $\hat{\beta}$ decreases, at $\lambda = \infty$, the LASSO gives the null model where $\hat{\beta} = 0$. Typically, we choose the value of λ that minimizes the expected prediction error, this can be handled

separately as a model selection problem. For details see [18].

4.1 *Coordinate Descent for the LASSO*

The LASSO objective function can be split into two parts – differentiable part $fd = \|\mathbf{Y} - \mathbf{X}\beta\|_2^2$ and non-differentiable part $fc = \sum_{j=1}^{p} |\beta_j|$. The non-differentiable part $fc = \sum_{j=1}^{p} |\beta_j|$ is strictly convex in each coordinate. Hence, we can apply coordinate wise minimization. We have seen that with a single predictor, the LASSO solution has a closed form solution, and is a soft-thresholded version of the least squares estimate. We exploit this property to implement the coordinate descent algorithm for the LASSO as follows.

As discussed previously, coordinate descent is an iterative method that solves exactly for one variable, keeping all other variables fixed. For each coordinate sub-problem, we fix all components of β except the jth component β_j. Let $\mathbf{X}_j$ denote the jth column of $\mathbf{X}$ and $\mathbf{X}_{-j}$ denote all the columns except jth column, then the problem is to find,

$$\underset{\beta_j \in \mathbb{R}}{\arg\min} \left\{ \frac{1}{n} \|\mathbf{Y} - \mathbf{X}_{-j}\beta_{-j} - \beta_j \mathbf{X}_j\|_2^2 + \lambda|\beta_j| + \lambda \sum_{l \neq j} |\beta_l| \right\}. \quad (29)$$

Define $r_j := \mathbf{Y} - \mathbf{X}_{-j}\beta_{-j}$, as partial residual (the partial residual is the difference between actual response $\mathbf{Y}$ and that portion of the fitted model that does not involve variable $\mathbf{X}_j$). Then the the problem (29) can be viewed as a univariate LASSO problem with vector r_j being the response variable.

$$\underset{\beta_j \in \mathbb{R}}{\arg\min} \left\{ \frac{1}{n} \|r_j - \beta_j \mathbf{X}_j\|_2^2 + \lambda|\beta_j| + \lambda \sum_{l \neq j} |\beta_l| \right\}. \quad (30)$$

Suppose $\hat{\beta}_j$ is a solution of the above optimization problem, then from the KKT stationarity condition we get the following:

$$-\frac{2}{n}\mathbf{X}_j^T(r_j - \hat{\beta}_j \mathbf{X}_j) + \lambda \; sign(\hat{\beta}_j) = 0,$$

$$\frac{1}{n} r_j^T \mathbf{X}_j - \hat{\beta}_j = \frac{\lambda}{2} \; sign(\hat{\beta}.)$$

Then the OLS estimator for the j^{th} variable can be computed as $(\hat{\beta}_{OLS})_j = \frac{1}{n} r^T \mathbf{X}_j$. Therefore univariate LASSO solution can be computed by soft-thresholding the OLS estimator as follows.

$$\hat{\beta}_j = \mathbb{S}_{\frac{\lambda}{2}}((\hat{\beta}_{OLS})_j)$$

Algorithm 1: CoordDesc Algorithm

Input: dataset (**Y, X**)

Output: $\hat{\beta}$:= LASSO estimated vector of regression coefficients

Initialize $\beta = 0$

repeat

  for ***each*** $j \in \{1, ..., p\}$ **do**

    Compute the partial residual r_j, where

$$r_j = \mathbf{Y} - \sum_{l \neq j} \mathbf{X}^l \beta_l$$

    Compute OLS coefficient for single predictor

$$(\hat{\beta}_{OLS})_j = \frac{1}{n} r_j^T \mathbf{X}_j$$

    Update β_j (LASSO solution: single variable case)

$$\beta_j = \mathbb{S}_{\frac{\lambda}{2}}((\hat{\beta}_{OLS})_j)$$

  end

until *convergence*;

$\hat{\beta} = \beta$

return $\hat{\beta}$

5. Applications

In this section, we consider a low dimensional simulation example for comparing prediction performance of the LASSO and the OLS regression. We also consider a high dimensional real world

problems, where we apply the LASSO method for estimation and variable selection.

5.1 *Simulation Example*

We consider the same simulation setup as given in the introduction section but with modified simulation steps as follows.

Modified Simulation Steps

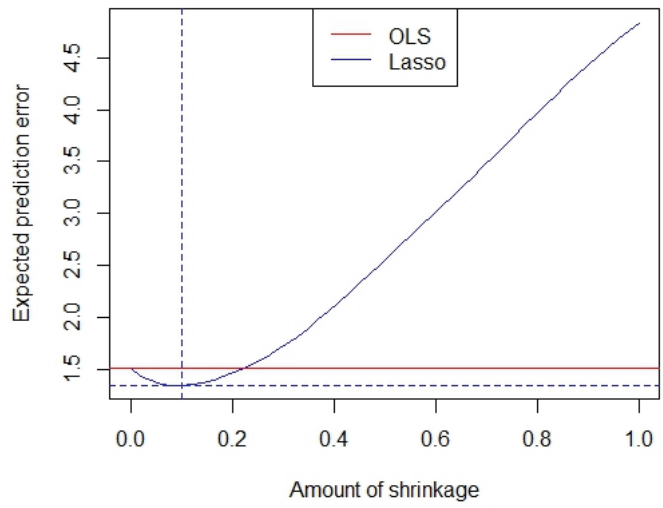
1. Generate an error vector as $\epsilon_{n\times 1} \sim N_n(0, I_n)$ and then compute a response vector as $\mathbf{Y} = \mathbf{X}\beta^0 + \epsilon$.
2. Generate another error vector as $\epsilon_{n\times 1} \sim N_n(0, I_n)$ and then compute a new response vector as $\mathbf{Y}_{test} = \mathbf{X}\beta^0 + \epsilon$.
3. Compute the OLS fit $\mathbf{X}\hat{\beta}_{OLS} = \mathbf{X}(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}$.
4. Compute the prediction error for OLS, as $PE(OLS) = \frac{1}{n}\|\mathbf{Y}_{test} - \mathbf{X}\hat{\beta}_{OLS}\|_2^2$.
5. Define a grid of 50 values of λ equally spaced between 0 and 1, as $\lambda_{seq} = [0, 0.02, ..., 1]$.
6. Compute the LASSO estimator $\hat{\beta}(\lambda)$ for each $\lambda \in \lambda_{seq}$ (using coordinate descent or LARS algorithm, etc.), and then compute the LASSO fit for each LASSO estimator as $\mathbf{X}\hat{\beta}(\lambda)$.
7. Compute the prediction error for the LASSO for each $\lambda \in \lambda_{seq}$, as $PE(LASSO, \lambda) = \frac{1}{n}\|\mathbf{Y}_{test} - \mathbf{X}\hat{\beta}(\lambda)\|_2^2$.
8. Compute the number of correct zero coefficients for each LASSO estimator $\hat{\beta}(\lambda)$ as, $NZC(\hat{\beta}(\lambda)) = \sum_{i=11}^{p} \mathbf{1}(\hat{\beta}(\lambda))_j = 0)$, where $\mathbf{1}$ is an indicator function.

We run the modified simulation steps for 100 times, and compute the expected prediction error (EPE) for the OLS and the LASSO by taking the average of the prediction error over 100 simulations. For the LASSO, the optimal estimator is the one that minimizes the EPE, and the corresponding λ value is the optimal tuning parameter say λ_{opt}. We also compute the expected (correct) zero coefficients at λ_{opt}. These results are reported in the *Table* 2 and *Figure* 4. From the *Table* 2, it is clear that the LASSO estimator

Table 2. Simulation results for the OLS and the LASSO.

Method	EPE	Average NZC
OLS	1.508	0
LASSO (0.102)	1.349	9.03

Figure 4. Comparision of the OLS and the LASSO.



at λ_{opt} almost correctly (9 out of 10) identifies the zero components of the true β^0. *Figure* 4 and *Table* 2, shows that EPE of the LASSO estimator at λ_{opt} is less than the EPE of the OLS estimator.

5.2 *Real Data Example*

We consider a real dataset of riboflavin. It consists of $n = 71$ observations of $p = 4088$ predictors (gene expressions) and a univariate response, riboflavin production rate (log-transformed). We refer to [19] for details on riboflavin dataset.

We use the ten-fold cross-validation procedure to select the optimal tuning parameters from a suitable grid of values. The performance measures (EPE and number of non-zero coefficients) are reported in *Table* 3. We do not report OLS results here, since the number of observations is much less than the number of covariates, and hence the OLS estimates are unstable.

Method (λ_{opt})	EPE	Average NZC
LASSO (0.036)	0.183	4047

Table 3. LASSO results on riboflavin dataset.

6. Computational Details

Statistical analysis was performed in *R* 3.2.2. We used the package 'MASS' for OLS regression and the package 'glmnet' for penalized regression method (the LASSO). All of our simulation code is available at request.

7. Concluding Remarks

The aim of this paper was to provide an introduction of the LASSO method as a constrained quadratic programming problem, and to discuss convex optimization based approach to solve the LASSO problem. We also discussed the situations when the LASSO problem has closed form solutions and for high dimensional case we considered an iterative method, coordinate descent algorithm. We also described the Bayesian interpretation of the LASSO estimates and the LASSO as a variable selection method. We have illustrated the applications of the LASSO using simulated and real data examples.

Acknowledgement

The author would like to thank the anonymous reviewer of this paper for his/her thoughtful suggestions, insights, highly constructive comments, and for the time spent reviewing various drafts of the paper.

Suggested Reading

[1] Simon Foucart and Holger Rauhut, *A Mathematical Introduction to Compressive Sensing*, New York: Springer, 2013.

[2] Trevor Hastie, Robert Tibshirani, and Jerome Friedman, *The Elements of Statistical Learning; Data Mining, Inference and Prediction*, New York: Springer, 2001.

[3] Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani, *An Introduction to Statistical Learning with Applications in R*, Springer, 2013.

[4] Tibshirani R, Regression Analysis and Selection via the Lasso, *Royal Statistical Society Series*, 58:267288, 1996.

[5] Stephen Boyd and Lieven Vandenberghe, *Convex Optimization*, Cambridge University Press, 2004.

[6] Dimitri P Bertsekas, *Convex Optimization Algorithms*, Athena Scientific, 2015.

[7] Tseng P, *Coordinate Ascent for Maximizing Non-differentiable Concave Functions*, Technical Report LIDS-P(MIT), 1988.

[8] George A F Seber and Alan J Lee, *Linear Regression Analysis*, Wiley, 2003.

[9] Edward I George, The Variable Selection Problem, *Journal of the American Statistical Association*, 95:13041308, 2000.

[10] Dziak J, Li R, and Collins L, *Critical Review and Comparison of Variable Selection Procedures for Linear Regression*, Technical report, 2005.

[11] Jianqing Fan and Jinchi Lv, A Selective Overview of Variable Selection in High-dimensional Feature Space, *Statistica Sinica*, 20(1):101148, 2010.

[12] Alois Kneip and Pascal Sarda, Factor Models and Variable Selection in High-dimensional Regression Analysis, *The Annals of Statistics*, 39(5):24102447, 2011.

[13] Trevor Hastie, Robert Tibshirani, and MartinWainwright, *Statistical Learning with Sparsity: The Lasso and Generalizations*, CRC Press, 2015.

[14] Bradley Efron, Trevor Hastie, Iain Johnstone, and Robert Tibshirani, Least Angle Regression, *Ann. Statist.*, 32(2):407499, 2004.

[15] B A Turlach, On Algorithms for Solving Least Squares Problems Under an l1 Penalty or an l1 Constraint, *Proceedings of the American Statistical Association, Statistical Computing Section*, pages 2572-2577, 2004.

[16] M R Osborne, B Presnell, and B A Turlach, A New Approach to Variable Selection in Least Squares Problems, *IMA Journal of Numerical Analysis*, 20(3):389403, 2000.

[17] Trevor Park and George Casella, The Bayesian Lasso, *Journal of the American Statistical Association*, 103(482):681686, 2008.

[18] Yiyun Zhang Runze Li and Chih-Ling Tsai, Regularization Parameter selections via Generalized Information Criterion, *Journal of the American Statistical Association*, 105(489):312323, 2010.

[19] Peter Bühlmann, Markus Kalisch, and Lukas Meier, High-dimensional Statistics with a View Towards Applications in Biology, *Annual Review of Statistics and its Applications*, 1:255278., 2014.

Address for Correspondence
Niharika Gauraha
Department of Pharmaceutical Biosciences
Uppsala Biomedicinska Centrum BMC
Husarg. 3
751 24 Uppsala, Sweden.
Email:
niharika.gauraha@gmail.com