

# An Interior-Point Method for Large-Scale $\ell_1$ -Regularized Least Squares

Seung-Jean Kim\*      Kwangmoo Koh\*      Michael Lustig<sup>†</sup>  
Stephen Boyd\*      Dmitry Gorinevsky\*

To appear in *IEEE Journal on Selected Topics in Signal Processing*

## Abstract

Recently, a lot of attention has been paid to  $\ell_1$  regularization based methods for sparse signal reconstruction (*e.g.*, basis pursuit denoising and compressed sensing) and feature selection (*e.g.*, the Lasso algorithm) in signal processing, statistics, and related fields. These problems can be cast as  $\ell_1$ -regularized least squares programs (LSPs), which can be reformulated as convex quadratic programs (QPs), and then solved by several standard methods such as interior-point methods (IPMs), at least for small and medium size problems. In this paper we describe a specialized interior-point method for solving large-scale  $\ell_1$ -regularized LSPs that uses the preconditioned conjugate gradients (PCG) algorithm to compute the search direction. The interior-point method can solve large sparse problems, with a million variables and observations, in a few tens of minutes on a PC. It can efficiently solve large dense problems, that arise in sparse signal recovery with orthogonal transforms, by exploiting fast algorithms for these transforms. The method is illustrated on a magnetic resonance imaging (MRI) data set.

**Key words:** compressed sensing, basis pursuit denoising, compressive sampling, convex optimization,  $\ell_1$  regularization, interior-point methods, least squares, preconditioned conjugate gradients.

## 1 Introduction

We consider a linear model of the form

$$y = Ax + v, \tag{1}$$

---

\*Information Systems Lab, Electrical Engineering Department, Stanford University, Stanford, CA 94305-9510 USA. Email: {sjkim, deneb1, boyd, gorin}@stanford.edu

<sup>†</sup>Magnetic Resonance Systems Research Lab, Electrical Engineering Department, Stanford University, Stanford, CA 94305-9510 USA. Email: mlustig@stanford.edu

where  $x \in \mathbf{R}^n$  is the vector of unknowns,  $y \in \mathbf{R}^m$  is the vector of observations,  $v \in \mathbf{R}^m$  is the noise, and  $A \in \mathbf{R}^{m \times n}$  is the data matrix.

When  $m \geq n$  and the columns of  $A$  are linearly independent, we can determine  $x$  by solving the least squares problem of minimizing the quadratic loss  $\|Ax - y\|_2^2$ , where  $\|u\|_2 = (\sum_i u_i^2)^{1/2}$  denotes the  $\ell_2$  norm of  $u$ . When  $m$ , the number of observations, is not large enough compared to  $n$ , simple least squares regression leads to over-fit.

## 1.1 $\ell_2$ -regularized least squares

A standard technique to prevent over-fitting is  $\ell_2$  or Tikhonov regularization [37], which can be written as

$$\text{minimize } \|Ax - y\|_2^2 + \lambda \|x\|_2^2, \quad (2)$$

where  $\lambda > 0$  is the regularization parameter. The Tikhonov regularization problem or  $\ell_2$ -regularized LSP has the analytic solution

$$x^{l2} = (A^T A + \lambda I)^{-1} A^T y. \quad (3)$$

We list some basic properties of Tikhonov regularization, that we refer to later when we compare it to  $\ell_1$ -regularized least squares.

- *Linearity.* From (3), we see that the solution  $x^{l2}$  to the Tikhonov regularization problem is a linear function of  $y$ .
- *Convergence to Moore-Penrose solution as  $\lambda \rightarrow 0$ .* As  $\lambda \rightarrow 0$ ,  $x^{l2}$  converges to the Moore-Penrose solution  $A^\dagger y$ , where  $A^\dagger$  is the Moore-Penrose pseudoinverse of  $A$ . The limit point has the minimum  $\ell_2$ -norm among all points that satisfy  $A^T(Ax - y) = 0$ :

$$A^\dagger y = \arg \min_{A^T(Ax - y) = 0} \|x\|_2.$$

- *Convergence to zero as  $\lambda \rightarrow \infty$ .* The optimal solution  $x^{l2}$  tends to zero, as  $\lambda \rightarrow \infty$ .
- *Regularization path.* The optimal solution  $x^{l2}$  is a smooth function of the regularization parameter  $\lambda$ , as it varies over  $[0, \infty)$ . As  $\lambda$  decreases to zero,  $x^{l2}$  converges to the Moore-Penrose solution; as  $\lambda$  increases,  $x^{l2}$  converges to zero.

The solution to the Tikhonov regularization problem can be computed by direct methods, which require  $O(n^3)$  flops (assuming that  $m$  is order  $n$  or less), when no structure is exploited. The solution can also be computed by applying iterative (non direct) methods (*e.g.*, the conjugate gradients method) to the linear system of equations  $(A^T A + \lambda I)x = A^T y$ . Iterative methods are efficient especially when there are fast algorithms for the matrix-vector multiplications with the data matrix  $A$  and its transpose  $A^T$  (*i.e.*,  $Au$  and  $A^T v$  with  $u \in \mathbf{R}^n$  and  $v \in \mathbf{R}^m$ ), which is the case when  $A$  is sparse or has a special form such as Fourier and wavelet matrices.

## 1.2 $\ell_1$ -regularized least squares

In  $\ell_1$ -regularized least squares (LS), we substitute a sum of absolute values for the sum of squares used in Tikhonov regularization, to obtain

$$\text{minimize } \|Ax - y\|_2^2 + \lambda \|x\|_1, \quad (4)$$

where  $\|x\|_1 = \sum_i |x_i|$  denotes the  $\ell_1$  norm of  $x$  and  $\lambda > 0$  is the regularization parameter. We call (4) an  $\ell_1$ -regularized LSP. This problem always has a solution, but it need not be unique.

We list some basic properties of  $\ell_1$ -regularized LS, pointing out similarities and differences with  $\ell_2$ -regularized LS.

- *Nonlinearity.* From (3) we see that Tikhonov regularization yields a vector  $x$  which is a linear function of the vector of observations  $y$ . By contrast,  $\ell_1$ -regularized least squares yields a vector  $x$  which is not linear in  $y$ .
- *Limiting behavior as  $\lambda \rightarrow 0$ .*  $\ell_1$ -regularized LS shows a different limiting behavior with  $\ell_1$ -regularized LS, as  $\lambda \rightarrow 0$ . The solution to the  $\ell_1$ -regularized LSP (4) converges to the point with the minimum  $\ell_1$  norm among all points that satisfy  $A^T(Ax - y) = 0$ , as  $\lambda \rightarrow 0$ .
- *Finite convergence to zero as  $\lambda \rightarrow \infty$ .* As in Tikhonov regularization, the optimal solution tends to zero, as  $\lambda \rightarrow \infty$ . For  $\ell_1$ -regularized LS, however, the convergence occurs for a finite value of  $\lambda$ ,

$$\lambda \geq \lambda_{\max} = \|2A^T y\|_{\infty}, \quad (5)$$

where  $\|u\|_{\infty} = \max_i |u_i|$  denotes the  $\ell_{\infty}$  norm of the vector  $u$ . For  $\lambda \geq \lambda_{\max}$ , the optimal solution is 0. In contrast, the optimal solution to the Tikhonov regularization problem is zero only in the limit as  $\lambda \rightarrow \infty$ . (The derivation of the formula (5) for  $\lambda_{\max}$  is given in §3.1.)

- *Regularization path.* The solution  $x^{\text{ls}}$  to the Tikhonov regularization problem varies smoothly as the regularization parameter  $\lambda$  varies over  $[0, \infty)$ . By contrast, the regularization path of the  $\ell_1$ -regularized LSP (4), *i.e.*, the family of solutions as  $\lambda$  varies over  $(0, \infty)$ , has the piecewise-linear solution path property [15]: There are values  $\lambda_1, \dots, \lambda_k$ , with  $0 = \lambda_k < \dots < \lambda_1 = \lambda_{\max}$ , such that the regularization path is a piecewise linear curve on  $\mathbf{R}^n$ ,

$$x^{\text{ls}} = \frac{\lambda_i - \lambda}{\lambda_i - \lambda_{i+1}} x^{(i+1)} + \frac{\lambda - \lambda_{i+1}}{\lambda_i - \lambda_{i+1}} x^{(i)}, \quad \lambda_{i+1} \leq \lambda \leq \lambda_i, \quad i = 1, \dots, k-1,$$

where  $x^{(i)}$  solves the  $\ell_1$ -regularized LSP (4) with  $\lambda = \lambda_i$ . (So  $x^{(1)} = 0$  and  $x^{\text{ls}} = 0$  when  $\lambda \geq \lambda_1$ .)

More importantly,  $\ell_1$ -regularized least squares typically yields a *sparse* vector  $x$ , *i.e.*,  $x$  which typically has relatively few nonzero coefficients. (As  $\lambda$  decreases, it tends to be sparser but not necessarily [25, 50].) In contrast, the solution  $x^{l_2}$  to the Tikhonov regularization problem typically has all coefficients nonzero.

Recently, the idea of  $\ell_1$  regularization has been receiving a lot of interest in signal processing and statistics. In signal processing, the idea of  $\ell_1$  regularization comes up in several contexts including basis pursuit denoising [8] and a signal recovery method from incomplete measurements (*e.g.*, [4, 7, 6, 11, 12, 52, 51]). In statistics, the idea of  $\ell_1$  regularization is used in the well-known Lasso algorithm [50] for feature selection and its extensions including the elastic net [60].

Some of these problems do not have the standard form (4) but have a more general form

$$\text{minimize } \|Ax - y\|_2^2 + \sum_{i=1}^n \lambda_i |x_i|, \quad (6)$$

where  $\lambda_i \geq 0$  are regularization parameters. (The variables  $x_i$  that correspond to  $\lambda_i = 0$  are not regularized.) This general problem can be reformulated as a problem of the form (4).

We now turn to the computational aspect of  $\ell_1$ -regularized least squares, the main topic of this paper. There is no analytic formula or expression for the optimal solution to the  $\ell_1$ -regularized LSP (4), analogous to (3); its solution must be computed numerically. The objective function in the  $\ell_1$ -regularized LSP (4) is convex but not differentiable, so solving it is more of a computational challenge than solving the  $\ell_2$ -regularized LRP (2).

Generic methods for nondifferentiable convex problems, such as the ellipsoid method or subgradient methods [49, 41], can be used to solve the  $\ell_1$ -regularized LSP (4). These methods are often very slow.

The  $\ell_1$ -regularized LSP (4) can be transformed to a convex quadratic problem, with linear inequality constraints. The equivalent QP can be solved by standard convex optimization methods such as interior-point methods [31, 36, 55, 56]. Standard interior-point methods are implemented in general purpose solvers including MOSEK [33], which can readily handle small and medium size problems. Standard methods cannot handle large problems in which there are fast algorithms for the matrix-vector operations with  $A$  and  $A^T$ . Specialized interior-point methods that exploit such algorithms can scale to large problems, as demonstrated in [8, 26]. High-quality implementations of specialized interior-point methods include **11-magic** [5] and PDCO [48], which use iterative algorithms, such as the conjugate gradients (CG) or LSQR algorithm [40], to compute the search step.

Recently, several researchers have proposed homotopy methods and variants for solving  $\ell_1$ -regularized LSPs [14, 23, 15, 44, 39]. Using the piecewise linear property of the regularization path, path-following methods can compute efficiently the entire solution path in an  $\ell_1$ -regularized LSP. When the solution of (15) is extremely sparse, these methods can be very fast, since the number of kinks the methods need to find is modest [14]. Otherwise, the path-following methods can be slow, which is often the case for large-scale problems. Other recently developed computational methods for  $\ell_1$ -regularized LSPs include coordinate-wise descent methods [19], sequential subspace optimization methods [34], bound optimization methods [17], iterated shrinkage methods [9, 16], gradient methods [35], and gradient projec-

tion algorithms [18]. Some of these methods including the gradient projection algorithms [18] can handle efficiently very large problems.

### 1.3 Outline

The main purpose of this paper is to describe a specialized interior-point method for solving large  $\ell_1$ -regularized LSPs that uses the preconditioned conjugate gradients algorithm to compute the search step. The method which we will describe in §4 can solve large sparse problems with  $m$  and  $n$  on the order of a million in a few tens of minutes on a PC. It can efficiently solve large dense problems, that arise in sparse signal recovery with orthogonal transforms, by exploiting fast algorithms for these transforms. The specialized method is far more efficient than interior-point methods that use direct or CG methods to compute the search step, as demonstrated with several examples in §5. Compared with first-order methods such as coordinate descent methods, our method is comparable in solving large problems with modest accuracy, but is able to solve them with high accuracy with relatively small additional computational cost.

We illustrate the method on a real magnetic resonance imaging data set in §5. The method can solve the QP that arises in reconstructing the object of interest with adequate accuracy from partially sampled Fourier coefficients, within around one hundred PCG steps, which amounts to performing a few hundred fast Fourier transform (FFT) operations on the object.

Although the interior-point method (we will describe in §4) is tailored toward a specific type of convex problem, the main idea behind the method can be generalized to other convex problems involving  $\ell_1$  regularization. We describe some generalizations of the method in §6. We also show how the method can easily handle the general  $\ell_1$ -regularized LSP (6), without forming explicitly the data matrix, which is typically dense, of its equivalent formulation of the form (4).

## 2 $\ell_1$ regularization in signal processing and statistics

Several signal processing and estimation methods based on the idea of  $\ell_1$  regularization can be formulated as problems of the form (4) or (6). In this section, we consider two important  $\ell_1$  regularization based methods. One of these is related to the MRI example give in §5.

### 2.1 Compressed sensing/compressive sampling

Let  $z$  be an unknown vector in  $\mathbf{R}^n$  (which can represent a two- or three-dimensional object of interest). Suppose that we have  $m$  linear measurements of an unknown signal  $z \in \mathbf{R}^n$ ,

$$y_i = \langle \phi_i, z \rangle + v_i, \quad i = 1, \dots, m,$$

where  $\langle \cdot, \cdot \rangle$  denotes the usual inner product  $v \in \mathbf{R}^m$  is the noise, and  $\phi_i \in \mathbf{R}^n$  are known signals. Standard reconstruction methods require at least  $n$  samples. Suppose we know a

priori that  $z$  is compressible or has a sparse representation in a transform domain, described by  $W \in \mathbf{R}^{n \times n}$  (after expanding the real and imaginary parts if necessary). In this case, if  $\phi_i$  are well chosen, then the number of measurements  $m$  can be dramatically smaller than the size  $n$  usually considered necessary [4, 11].

Compressed sensing [11] or compressive sampling [4] attempts to exploit the sparsity or compressibility of the true signal in the transform domain by solving a problem of the form

$$\text{minimize} \quad \|\Phi z - y\|_2^2 + \lambda \|Wz\|_1, \quad (7)$$

where the variable is  $z \in \mathbf{R}^n$ . Here,  $\Phi = [\phi_1 \cdots \phi_m]^T \in \mathbf{R}^{m \times n}$  is called the compressed sensing matrix,  $\lambda > 0$  is the regularization parameter, and  $W$  is called the sparsifying transform. Compressed sensing has a variety of potential applications including analog-to-information conversion and sparse MRI.

When  $W$  is invertible, we can reformulate the compressed sensing problem (7) as the  $\ell_1$ -regularized LSP

$$\text{minimize} \quad \|Ax - y\|_2^2 + \lambda \|x\|_1 \quad (8)$$

where the variable is  $x \in \mathbf{R}^n$  and the problem data or parameters are  $A = \Phi W^{-1} \in \mathbf{R}^{m \times n}$  and  $y \in \mathbf{R}^m$ . (If  $x^*$  solves (8), then  $W^{-1}x^*$  solves (7), and conversely, if  $z^*$  solves (7), then  $x^* = Wz^*$  solves (8).) This problem is a basis pursuit denoising (BPDN) problem with the dictionary matrix  $A$ ; see, *e.g.*, [8] for more on basis pursuit denoising.

Before proceeding, we should mention an important feature of the  $\ell_1$ -regularized LSPs that arise in compressed sensing. The data matrix  $A$  in the  $\ell_1$ -regularized LSP (8) is typically fully dense, and so its equivalent QP formulation is fully dense. But the equivalent dense QP has an important difference from general dense QPs: there are fast algorithms for the matrix-vector operations with  $A$  and  $A^T$ , based on fast algorithms for the sparsifying transform and its inverse transform.

## 2.2 $\ell_1$ regularized linear regression

Let  $u \in \mathbf{R}^n$  denote a vector of explanatory or feature variables, and  $y \in \mathbf{R}$  denote the associated output or outcome. A linear model predicts the output as

$$\hat{y} = \beta^T u + \beta_{\text{bias}},$$

where  $\beta_{\text{bias}} \in \mathbf{R}$  is the bias or intercept and  $\beta \in \mathbf{R}^n$  is the weight vector.

Suppose we are given a set of (observed or training) examples,  $(u_i, y_i) \in \mathbf{R}^n \times \mathbf{R}$ ,  $i = 1, \dots, m$ . To estimate the weight coefficients and intercept, the Lasso algorithm [50] solves the  $\ell_1$ -regularized LSP

$$\text{minimize} \quad \sum_{i=1}^m (\beta^T u_i + \beta_{\text{bias}} - y_i)^2 + \sum_{i=1}^n \lambda |\beta_i|, \quad (9)$$

with variables  $\beta_{\text{bias}} \in \mathbf{R}$  and  $\beta \in \mathbf{R}^n$ . The Lasso problem (9) has the form (6) with the variables  $x = (\beta, \beta_{\text{bias}}) \in \mathbf{R}^{n+1}$ . Here, the intercept  $\beta_{\text{bias}}$  is not regularized.

Extensive research has shown that  $\ell_1$ -regularized linear regression can outperform  $\ell_2$ -regularized linear regression (also called ridge regression), especially when the number of observations is smaller than the number of features [15, 50]. Recently, theoretical properties of  $\ell_1$ -regularized linear regression have been studied by several researchers; see, *e.g.*, [20, 28, 32, 54, 59, 58].

### 3 Optimality conditions and dual problem

In this section, we give some preliminaries needed later.

#### 3.1 Optimality conditions

The objective function of the  $\ell_1$ -regularized LSP (4) is convex but not differentiable, so we use a first-order optimality condition based on subdifferential calculus. We can obtain the following necessary and sufficient conditions for  $x$  to be optimal for the  $\ell_1$ -regularized LSP (4):

$$(2A^T(Ax - y))_i \in \begin{cases} \{+\lambda_i\} & x_i > 0, \\ \{-\lambda_i\} & x_i < 0, \\ [-\lambda_i, \lambda_i] & x_i = 0, \end{cases} \quad i = 1, \dots, n. \quad (10)$$

We can now derive the formula (5) for  $\lambda_{\max}$ . The condition that 0 is optimal is that  $(2A^T y)_i \in [-\lambda, \lambda]$  for  $i = 1, \dots, n$ , *i.e.*,  $\|2A^T y\|_\infty \leq \lambda$ .

#### 3.2 Dual problem and suboptimality bound

We derive a Lagrange dual of the  $\ell_1$ -regularized LSP (4). We start by introducing a new variable  $z \in \mathbf{R}^m$ , as well as new equality constraints  $z = Ax - y$ , to obtain the equivalent problem

$$\begin{aligned} & \text{minimize} && z^T z + \lambda \|x\|_1 \\ & \text{subject to} && z = Ax - y. \end{aligned} \quad (11)$$

Associating dual variables  $\nu_i \in \mathbf{R}$ ,  $i = 1, \dots, m$  with the equality constraints  $z_i = (Ax - y)_i$ , the Lagrangian is

$$L(x, z, \nu) = z^T z + \lambda \|x\|_1 + \nu^T (Ax - y - z).$$

The dual function is

$$\inf_{x, z} L(x, z, \nu) = \begin{cases} -(1/4)\nu^T \nu - \nu^T y, & |(A^T \nu)_i| \leq \lambda_i, \quad i = 1, \dots, m, \\ -\infty, & \text{otherwise.} \end{cases}$$

The Lagrange dual of (11) is therefore

$$\begin{aligned} & \text{maximize} && G(\nu) \\ & \text{subject to} && |(A^T \nu)_i| \leq \lambda_i, \quad i = 1, \dots, m, \end{aligned} \quad (12)$$

where the dual objective  $G(\nu)$  is

$$G(\nu) = -\frac{1}{4}\nu^T\nu - \nu^Ty.$$

(See, *e.g.*, [3, Chap.5] or [2] for more on convex duality.) The dual problem (12) is a convex optimization problem with variable  $\nu \in \mathbf{R}^m$ . We say that  $\nu \in \mathbf{R}^m$  is dual feasible if it satisfies the constraints of (12), *i.e.*,  $|(A^T\nu)_i| \leq \lambda_i$ ,  $i = 1, \dots, m$ .

Any dual feasible point  $\nu$  gives a lower bound on the optimal value  $p^*$  of the primal problem (4), *i.e.*,  $G(\nu) \leq p^*$ , which is called weak duality. Furthermore, the optimal values of the primal and dual are equal since the primal problem (4) satisfies Slater's condition, which is called strong duality [3].

An important property of the  $\ell_1$ -regularized LSP (4) is that from an arbitrary  $x$  we can derive an easily computed bound on the suboptimality of  $x$ , by constructing a dual feasible point

$$\nu = 2s(Ax - y), \quad s = \min \{ \lambda / |2((A^T Ax)_i - 2y_i)| \mid i = 1, \dots, m \}. \quad (13)$$

The point  $\nu$  is dual feasible, so  $G(\nu)$  is a lower bound on  $p^*$ , the optimal value of the  $\ell_1$ -regularized LSP (4).

The difference between the primal objective value of  $x$  and the associated lower bound  $G(\nu)$  is called the *duality gap*. We use  $\eta$  to denote the gap:

$$\eta = \|Ax - y\|_2^2 + \lambda\|x\|_1 - G(\nu). \quad (14)$$

The duality gap is always nonnegative by weak duality, and  $x$  is no more than  $\eta$ -suboptimal. At an optimal point, the duality gap is zero, *i.e.*, strong duality holds.

## 4 A truncated Newton interior-point method

The  $\ell_1$ -regularized LSP (4) can be transformed to a convex quadratic problem, with linear inequality constraints,

$$\begin{aligned} & \text{minimize} && \|Ax - y\|_2^2 + \lambda \sum_{i=1}^n u_i \\ & \text{subject to} && -u_i \leq x_i \leq u_i, \quad i = 1, \dots, n, \end{aligned} \quad (15)$$

where the variables are  $x \in \mathbf{R}^n$  and  $u \in \mathbf{R}^n$ . In this section we describe an interior-point method for solving this equivalent QP.

### 4.1 A custom interior-point method

We start by defining the logarithmic barrier for the bound constraints  $-u_i \leq x_i \leq u_i$  in (15),

$$\Phi(x, u) = -\sum_{i=1}^n \log(u_i + x_i) - \sum_{i=1}^n \log(u_i - x_i)$$



defined over  $\text{dom } \Phi = \{(x, u) \in \mathbf{R}^n \times \mathbf{R}^n \mid |x_i| < u_i, i = 1, \dots, n\}$ . The central path consists of the unique minimizer  $(x^*(t), u^*(t))$  of the convex function

$$\phi_t(x, u) = t\|Ax - y\|_2^2 + t \sum_{i=1}^n \lambda u_i + \Phi(x, u), \quad (16)$$

as the parameter  $t$  varies from 0 to  $\infty$ . With each point  $(x^*(t), u^*(t))$  on the central path we associate  $\nu^*(t) = 2(Ax^*(t) - y)$ , which can be shown to be dual feasible. (Indeed, it coincides with the dual feasible point  $\nu$  constructed from  $x^*(t)$  using the method of Section 3.2.) In particular,  $(x^*(t), u^*(t))$  is no more than  $2n/t$ -suboptimal, so the central path leads to an optimal solution.

In a primal interior-point method, we compute a sequence of points on the central path, for an increasing sequence of values of  $t$ , starting from the previously computed central point. (A typical method uses the sequence  $t = t_0, \mu t_0, \mu^2 t_0, \dots$ , where  $\mu$  is between 2 and 50 [3, §11.3]. The method can be terminated when  $2n/t \leq \epsilon$ , where  $\epsilon$  is the target duality gap, since then we can guarantee  $\epsilon$ -suboptimality of  $(x^*(t), u^*(t))$ . The reader is referred to [3, Chap.11] for more on the primal barrier method.) In the primal barrier method, Newton's method is used to minimize  $\phi_t$ , *i.e.*, the search direction is computed as the exact solution to the Newton system

$$H \begin{bmatrix} \Delta x \\ \Delta u \end{bmatrix} = -g, \quad (17)$$

where  $H = \nabla^2 \phi_t(x, u) \in \mathbf{R}^{2n \times 2n}$  is the Hessian and  $g = \nabla \phi_t(x, u) \in \mathbf{R}^{2n}$  is the gradient at the current iterate  $(x, u)$ .

For a large  $\ell_1$ -regularized LSP, solving the Newton system (17) exactly is not computationally practical. We need to find a search direction which gives a good trade-off of computational effort versus the convergence rate it provides. In the method described below, the search direction is computed as an approximate solution to the Newton system (17), using a truncated Newton method. (Truncated Newton methods have been applied to interior-point methods; see, *e.g.*, [26, 29, 53, 42].)

In the primal barrier method, the parameter  $t$  is held constant until  $\phi_t$  is (approximately) minimized, *i.e.*,  $\|\nabla \phi_t\|_2$  is small. For faster convergence, the method described below attempts to update the parameter  $t$  at each iteration, using the observation made in §3.2 that we can cheaply compute a dual feasible point and the associated duality gap for any  $x$ .

TRUNCATED NEWTON IPM FOR  $\ell_1$ -REGULARIZED LSPs.

**given** relative tolerance  $\epsilon_{\text{rel}} > 0$ .

Initialize.  $t := 1/\lambda$ ,  $x := 0$ ,  $u := \mathbf{1} = (1, \dots, 1) \in \mathbf{R}^n$ .

**repeat**

1. Compute the search direction  $(\Delta x, \Delta u)$  as an approximate solution to the Newton system (17).
2. Compute the step size  $s$  by backtracking line search.
3. Update the iterate by  $(x, u) := (x, u) + s(\Delta x, \Delta u)$ .

4. Construct a dual feasible point  $\nu$  from (13).
5. Evaluate the duality gap  $\eta$  from (14).
6. **quit** if  $\eta/G(\nu) \leq \epsilon_{\text{rel}}$ .
7. Update  $t$ .

As a stopping criterion, the method uses the duality gap divided by the dual objective value. By weak duality, the ratio is an upper bound on the relative suboptimality:

$$\frac{f(x) - p^*}{p^*} \leq \frac{\eta}{G(\nu)},$$

where  $p^*$  is the optimal value of the  $\ell_1$ -regularized LSP (4) and  $f(x)$  is the primal objective computed with the point  $x$  (computed in step 3). Therefore, the method can solve the problem to guaranteed relative accuracy  $\epsilon_{\text{rel}}$ .

Given the search direction  $(\Delta x, \Delta u)$ , the new point is  $(x, u) + s(\Delta x, \Delta u)$ , where  $s \in \mathbf{R}_+$ , the step size, is to be computed. In the backtracking line search, the step size is taken as  $s = \beta^k$  where  $k \geq 0$  is the smallest integer that satisfies

$$\phi_t(x + \beta^k \Delta x, u + \beta^k \Delta u) \leq \phi_t(x, u) + \alpha \beta^k \nabla \phi_t(x, u)^T \begin{bmatrix} \Delta x \\ \Delta u \end{bmatrix},$$

where  $\alpha \in (0, 1/2)$  and  $\beta \in (0, 1)$  are algorithm parameters. (Typical values for the line search parameters are  $\alpha = 0.01$ ,  $\beta = 0.5$ .) The reader is referred to [3, Chap.9] for more on the backtracking line search.

We use the update rule

$$t := \begin{cases} \max\{\mu \min\{2n/\eta, t\}, t\}, & s \geq s_{\min} \\ t, & s < s_{\min}, \end{cases}$$

where  $\mu > 1$  and  $s_{\min} \in (0, 1]$  are parameters to be chosen. It has been used in the custom interior-point method for  $\ell_1$ -regularized logistic regression described in [29]. The choice of  $\mu = 2$  and  $s_{\min} = 0.5$  appears to give good performance for a wide range of problems. This update rule uses the step length  $s$  as a crude measure of proximity to the central path. In particular,  $2n/t$  is the value of  $t$  for which the associated central point has the same duality gap as the current point. The update rule appears to be quite robust and work well when combined with the PCG algorithm we will describe soon. The reader is referred to [29] for an informal justification of convergence of the interior-point method based on this update rule (with exact search directions).

## 4.2 Search direction via PCGs

We can find compact representations of the Hessian and gradient. The Hessian can be written as

$$H = t \nabla^2 \|Ax - y\|_2^2 + \nabla^2 \Phi(x, u) = \begin{bmatrix} 2tA^T A + D_1 & D_2 \\ D_2 & D_1 \end{bmatrix},$$

where

$$\begin{aligned} D_1 &= \text{diag} \left( 2(u_1^2 + x_1^2)/(u_1^2 - x_1^2)^2, \dots, 2(u_n^2 + x_n^2)/(u_n^2 - x_n^2)^2 \right) \in \mathbf{R}^n, \\ D_2 &= \text{diag} \left( -4u_1x_1/(u_1^2 - x_1^2)^2, \dots, -4u_nx_n/(u_n^2 - x_n^2)^2 \right) \in \mathbf{R}^n. \end{aligned}$$

Here we use  $\text{diag}(A_1, \dots, A_p)$  to denote the diagonal matrix with diagonal blocks  $A_1, \dots, A_p$ . The Hessian  $H$  is symmetric and positive definite. The gradient can be written as

$$g = \begin{bmatrix} g_1 \\ g_2 \end{bmatrix} \in \mathbf{R}^{2n},$$

where

$$\begin{aligned} g_1 &= \nabla_x \phi_t(x, u) = 2tA^T(Ax - y) + \begin{bmatrix} 2x_1/(u_1^2 - x_1^2) \\ \vdots \\ 2x_n/(u_n^2 - x_n^2) \end{bmatrix} \in \mathbf{R}^n, \\ g_2 &= \nabla_u \phi_t(x, u) = t\lambda \mathbf{1} - \begin{bmatrix} 2u_1/(u_1^2 - x_1^2) \\ \vdots \\ 2u_n/(u_n^2 - x_n^2) \end{bmatrix} \in \mathbf{R}^n. \end{aligned}$$

We compute the search direction approximately, applying the PCG algorithm [10, §6.6] to the Newton system (17). The PCG algorithm uses a preconditioner  $P \in \mathbf{R}^{2n \times 2n}$ , which is symmetric and positive definite. We will not go into the details of the PCG algorithm, and refer the reader to [27], [46, §6.7], or [38, §5].

The preconditioner used in the PCG algorithm approximates the Hessian of  $t\|Ax - y\|_2^2$  with its diagonal entries, while retaining the Hessian of the logarithmic barrier  $\Phi(x, u)$ :

$$P = \begin{bmatrix} 2t \text{diag}(A^T A) & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} D_1 & D_2 \\ D_2 & D_1 \end{bmatrix}. \quad (18)$$

Here  $\text{diag}(S)$  is the diagonal matrix obtained by setting the off-diagonal entries of the matrix  $S$  to zero.

The cost of computing the diagonal entries of  $A^T A$  can be amortized over all interior-point iterations and multiple problems with the same data matrix and different observation vectors, since we need to compute them only once. When the amortized cost is still expensive, we can approximate the diagonal matrix  $\text{diag}(A^T A)$  with a scaled identity matrix to obtain the preconditioner

$$P = \begin{bmatrix} 2\tau t I & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} D_1 & D_2 \\ D_2 & D_1 \end{bmatrix}, \quad (19)$$

where  $\tau$  is a positive constant. This preconditioner performs well especially when the diagonal elements of  $A^T A$  show relatively small variations.

The PCG algorithm needs a good initial search direction and an effective truncation rule.

- *Initial point.* There are many choices for the initial search direction, *e.g.*, 0, the negative gradient, and the search direction found in the previous step of the method. The previous search direction appears to have a small advantage over the negative gradient and 0.
- *Truncation rule.* The truncation rule for the PCG algorithm gives the condition for terminating the algorithm. The truncation rule in our implementation is simple: the PCG algorithm stops when the cumulative number of PCG steps exceeds the given limit  $N_{\text{pcg}}$  or we compute a point with relative tolerance less than  $\epsilon_{\text{pcg}}$ . We use the adaptive relative tolerance change rule

$$\epsilon_{\text{pcg}} = \min \{0.1, \xi\eta/\|g\|_2\},$$

where  $\eta$  is the duality gap at the current iterate and  $\xi$  is an algorithm parameter. (The choice of  $\xi = 0.01$  appears to work well for a wide range of problems.) In other words, we solve the Newton system with low accuracy (but never worse than 10%) at early iterations, and solve it more accurately as the duality gap decreases.

Each iteration of the PCG algorithm involves a handful of inner products, a matrix-vector product  $Hp$  with  $p = (p_1, p_2) \in \mathbf{R}^{2n}$  and a solve step with  $P$  in computing  $P^{-1}r$  with  $r = (r_1, r_2) \in \mathbf{R}^{2n}$ . (Here, we use the convention that for column vectors  $a$  and  $b$ ,  $(a, b)$  is the column vector obtained by stacking  $a$  on top of  $b$ .) The solve step  $P^{-1}r$  can be computed as

$$P^{-1}r = \begin{bmatrix} (D_1D_3 - D_2^2)^{-1}(D_1r_1 - D_2r_2) \\ (D_1D_3 - D_2^2)^{-1}(-D_2r_1 + D_3r_2) \end{bmatrix},$$

where  $D_3 = 2t \text{diag}(A^T A) + D_1$  for the preconditioner (18) and  $D_3 = 2\tau tI + D_1$  for the preconditioner (19). The computational cost is  $O(n)$  flops.

The computationally most expensive operation for a PCG step is therefore the matrix-vector product  $Hp$  with  $p = (p_1, p_2) \in \mathbf{R}^{2n}$ . The product can be computed as

$$Hp = \begin{bmatrix} 2tA^T v + D_1p_2 \\ D_2p_1 + D_1p_2 \end{bmatrix},$$

where  $v = Ap_1$ . The complexity of  $Hp$  depends on the problem data and determines the cost of a PCG step. We consider two cases.

- *Sparse problems.* The cost of computing  $Hp$  primarily depends on the number of nonzeros in the matrix  $A$ . The cost is  $O(p)$  flops when  $A$  has  $p$  nonzero elements.
- *Dense problems with fast matrix-vector multiplications algorithms.* The cost of computing  $Hp$  is  $O(nm)$  if no structure is exploited. If fast algorithms are available for the matrix-vector multiplications with  $A$  and  $A^T$ , the cost can be reduced substantially. As an example we consider the compressed sensing problem (7) where  $\Phi$  is a Fourier matrix and  $W$  is an orthogonal wavelet matrix ( $W^{-1} = W^T$ ). The matrix-vector multiplication  $Au = \Phi W^{-1}u$  with  $u \in \mathbf{R}^n$  can be done efficiently using the fast algorithms

for the inverse wavelet transform and the discrete Fourier transform (DFT). The multiplication  $A^T v = W^{-T} \Phi^T v = W \Phi^T v$  can be computed efficiently in a similar manner using the fast algorithm for the wavelet transform. For any  $v \in \mathbf{R}^m$ , we can efficiently compute  $\Phi^T \in \mathbf{R}^m$  using zero-filling and then performing the inverse Fourier transform. The cost of a PCG step is therefore  $O(n \log n)$ . (The Hessian-vector product  $H p$  involves one FFT, one inverse FFT, one discrete wavelet transform (DWT), and one inverse DWT operation.)

### 4.3 Performance

Since the memory requirement of the truncated Newton interior-point method is modest, the method is able to solve very large problems, for which forming the Hessian  $H$  (let alone computing the search direction) would be prohibitively expensive. The runtime of the truncated Newton interior-point method is determined by the product of the total number of PCG steps required over all iterations and the cost of a PCG step. The total number of PCG iterations required by the truncated Newton interior-point method depends much on the value of the regularization parameter  $\lambda$  and the relative tolerance  $\epsilon_{\text{rel}}$ . In particular, for very small values of regularization parameters (which lead to large values of  $\text{card}(x)$ ), the truncated Newton interior-point method requires a larger total number of PCG steps.

In extensive testing, we found that the total number of PCG steps ranges between a few tens (for medium size problems) and several hundreds (for very large problems) to compute a solution which is never worse than 1% suboptimal, *i.e.*, with relative tolerance 0.01. In particular, we observed that the total number of PCG steps remains modest (around a few hundreds) nearly irrespective of the size of the problem, when the mutual coherence of the data matrix  $A$  is small, *i.e.*, the off-diagonal elements of  $A^T A$  are relatively small compared with the diagonal elements. (This observation is not very surprising, since as the mutual coherence tends to zero, the Hessian in the Newton system (17) becomes more and more diagonally dominated.)

## 5 Numerical examples

In this section we give some numerical examples to illustrate the performance of the truncated Newton interior-point method (TNIPM)) described in §4. The algorithm parameters are taken as

$$\alpha = 0.01, \quad \beta = 0.5, \quad s_{\min} = 0.5, \quad \mu = 2, \quad \xi = 0.01, \quad N_{\text{pcg}} = 200.$$

Since the iteration limit  $N_{\text{pcg}}$  in the PCG algorithm is set to be large enough, it was never reached in our experiments.

We compare the performance of the Matlab implementation the truncated Newton interior-point method (available from [http://www.stanford.edu/~boyd/l1\\_ls/](http://www.stanford.edu/~boyd/l1_ls/)) and the following five existing solvers: MOSEK, PDCO-CHOL, PDCO-LSQR, l1-magic, and Homotopy. MOSEK [33] is a C implementation of a primal-dual interior-point method with Matlab interface.

	Runtime (in seconds)
TNIPM	6.9
MOSEK	141
PDCO-CHOL	72
PDCO-LSQR	109
l1-magic	538
Homotopy	11.3

**Table 1:** Runtime of the truncated Newton interior-point method (TNIPM) and the five existing methods, for a signal denoising problem.

PDCO-CHOL is a Matlab implementation of a primal-dual interior-point method [48] that uses the Cholesky factorization algorithm to compute the search direction. PDCO-LSQR is a Matlab implementation of a primal-dual interior-point method [48] that uses the LSQR algorithm [40] to compute the search direction. (PDCO-CHOL and PDCO-LSQR are available in **SparseLab** [13], a library of Matlab routines for finding sparse solutions to underdetermined systems.) **l1-magic** [5] is a Matlab package devoted to compressed sensing problems. PDCO-LSQR and **l1-magic** implement interior-point methods that use the CG or LSQR method to compute the search direction and hence are similar in spirit to our method. **Homotopy** is an implementation of the homotopy method [14], available in **SparseLab**.

The existing methods and TNIPM were run on an AMD 270 under Linux.

## 5.1 A sparse signal recovery example

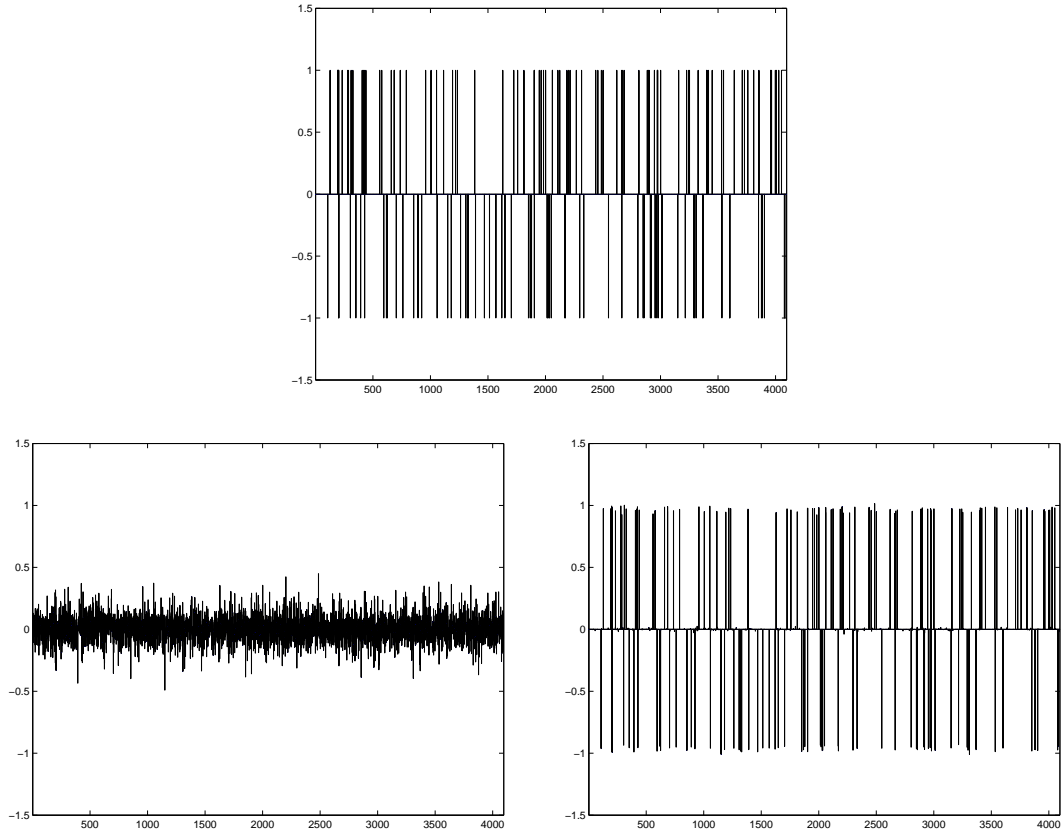
As an example, we consider a sparse signal recovery problem with a signal  $x \in \mathbf{R}^{4096}$  which consists of 160 spikes with amplitude  $\pm 1$ , shown at the top plot of Fig. 1. The measurement matrix  $A \in \mathbf{R}^{1024 \times 4096}$  is created by first generating a matrix of size  $\mathbf{R}^{1024 \times 4096}$  with entries generated independently and identically according to the standard normal distribution and then orthogonalizing the rows, as with the example in [5, §3.1]. In the problem, we have

$$y = Ax + v,$$

where  $v$  is drawn according to the Gaussian distribution  $\mathcal{N}(0, 0.01^2 I)$  on  $\mathbf{R}^{1024}$ .

Our method and the five existing methods above were run to find a point which is not worse than 1% suboptimal, *i.e.*, with relative tolerance 0.01. We compare the reconstruction obtained by solving the  $\ell_1$ -regularized LSP (4), called basis pursuit denoising (BPDN) with  $\lambda = 0.01$  with the minimum energy reconstruction  $A^T y$  (which is the point in the set  $\{x \mid Ax = y\}$  closest to the origin). Fig. 1 shows the reconstruction results. The  $\ell_1$ -regularization based method reconstructs the signal faithfully, although the number of measurements is far less than the number of elements.

Table 1 compares the runtimes of the Matlab implementation of our method and the five existing methods. The truncated Newton interior-point method is most efficient for this medium problem.



**Figure 1:** Reconstruction of the spike signal (top). Bottom left: Minimum energy reconstruction. Bottom right: BPDN reconstruction.





### 5.3 Scalability

To examine the effect of problem size on the runtime of the truncated Newton interior-point method, we generated a family of 10 sparse data sets, with the number of features  $n$  varying from one thousand to one million, and  $m = 0.1n$  examples, *i.e.*, 10 times more features than examples. In doing so, the sparsity of  $A$  was controlled so that the total number of nonzero entries in  $A$  was  $p \approx 30m$ . The elements of  $A$  were independent and identically distributed, drawn from the standard normal distribution. For each data set,  $x$  was generated to be sparse with  $0.25n$  nonzero elements. The measurements  $Ax$  were corrupted by white noise with zero mean and variance  $0.01^2 I$ . The regularization parameter was taken as  $\lambda = 0.1\lambda_{\max}$ , where the value of  $\lambda_{\max}$  was computed using the formula given in (5).

Fig. 3 summarizes the scalability comparison results for the regularization parameter, when our method and the four existing methods except for **Homotopy** were run to find a point which is not worse than 1% suboptimal, *i.e.*, with relative tolerance 0.01. (**Homotopy** was excluded in the scalability comparison, since the current implementation available in [5] does not handle sparse data effectively.) Evidently the truncated Newton interior-point method is more efficient for small problems, and far more efficient for medium and large problems. By fitting an exponent to the data over the range from  $n = 1000$  to the largest problem successfully solved by each method, we found that the empirical complexities of our method and the four existing methods. The empirical complexity of TNIPM was  $O(n^{1.2})$  and that of **11-magic** was  $O(n^{1.3})$ . Empirical complexities of other solvers were more than quadratic.

## 6 Extensions and variations

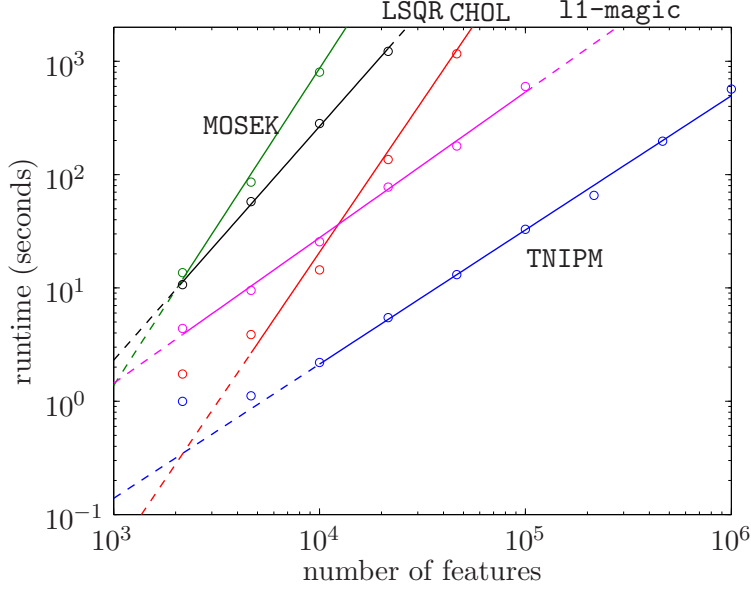
We can solve the general  $\ell_1$ -regularized LSP (6), using The Truncated Newton interior-point method described in §4. We can extend the idea behind this method to some other problems involving  $\ell_1$  regularization. The most important part in the extensions is to find a preconditioner that gives a good trade-off between the computational complexity and the accelerated convergence it provides. We will focus on this part, and will not go into the details of other parts which are rather straightforward.

### 6.1 General $\ell_1$ -regularized LSPs

The general  $\ell_1$ -regularized LSP (6) can be reformulated as

$$\text{minimize} \quad \inf_{x_i, i \in J} \|Ax - y\|_2^2 + \sum_{i \notin J} \lambda_i \|x_i\| \quad (20)$$

where  $J = \{i \in \{1, \dots, n\} \mid \lambda_i = 0\}$  and the variables are  $x_i, i \notin J$ . The first term in the objective can be evaluated analytically and is in fact a quadratic function of the variables  $x_i, i \notin J$ . This problem can be cast as a problem of the form (4) via a simple transform of variables,  $z_i = (\lambda_i/\lambda)x_i$ , and hence can be solved using the truncated Newton interior-point method described above.



**Figure 3:** Runtime of the truncated Newton interior-point method (TNIPM), MOSEK, PDCO-CHOL (abbreviated as CHOL), PDCO-LSQR (abbreviated as LSQR), and `l1-magic`, for 10 randomly generated sparse problems, with the regularization parameter  $\lambda = 0.1\lambda_{\max}$ .

We should point out that the method does not need to form explicitly the data matrix of the equivalent  $\ell_1$ -regularized LSP; it needs an algorithm for multiplying a vector by the data matrix of the equivalent formulation and an algorithm for multiplying a vector by its transpose. These matrix-vector multiplications can be performed with the nearly same computational cost as those with the original data matrix  $A$  and its transpose  $A^T$ , provided that the number of unregularized variables (*i.e.*,  $x_i, i \in J$ ) is small.

As an example, we consider the Lasso problem (9). It can be transformed to a problem of the form

$$\text{minimize} \quad \|Ax - y + (1/n)\mathbf{1}\mathbf{1}^T y\|_2^2 + \lambda\|x\|_1,$$

where the variables are  $x = \beta \in \mathbf{R}^n$  and the problem data are

$$A = \begin{bmatrix} u_1^T \\ \vdots \\ u_n^T \end{bmatrix} - \frac{1}{n}\mathbf{1}\sum_{i=1}^n u_i^T \in \mathbf{R}^{n \times n}.$$

The data matrix  $A$  in the equivalent formulation is the sum of the original data matrix and a rank-one matrix, so the matrix-vector multiplications with  $A$  and  $A^T$  can be performed with the same computational cost as those with the original data matrix and its transpose.

## 6.2 $\ell_1$ -regularized LSPs with nonnegativity constraints

Suppose that the variable  $x \in \mathbf{R}^n$  is known to be nonnegative. We add the nonnegativity constraint to the  $\ell_1$ -regularized LSP (4), to obtain

$$\begin{aligned} & \text{minimize} && \|Ax - y\|_2^2 + \lambda \|x\|_1 = \|Ax - y\|_2^2 + \lambda \sum_{i=1}^n x_i \\ & \text{subject to} && x_i \geq 0, \quad i = 1, \dots, n. \end{aligned} \quad (21)$$

The associated centering problem is to minimize the weighted objective function augmented by the logarithmic barrier for the constraints  $x_i \geq 0$ ,

$$t\|Ax - y\|_2^2 + t\lambda \sum_{i=1}^n x_i - \sum_{i=1}^n \log x_i.$$

The Newton system for the centering problem is

$$(2tA^T A + D)\Delta x_{\text{nt}} = -g,$$

where

$$D = \begin{bmatrix} 1/x_1^2 \\ \vdots \\ 1/x_n^2 \end{bmatrix}, \quad g = 2tA^T(Ax - y) + \begin{bmatrix} t\lambda_1 - 1/x_1 \\ \vdots \\ t\lambda_n - 1/x_n \end{bmatrix}.$$

The diagonal preconditioner of the form

$$P = \text{diag}(2tA^T A) + D$$

works well for this problem, especially when the optimal solution is sparse.

## 6.3 Isotonic regression with $\ell_1$ regularization

We consider a variation of the  $\ell_1$ -regularized LSP (4) subject to the monotonicity constraints,

$$x_1 \leq \dots \leq x_n,$$

which can be written as

$$\begin{aligned} & \text{minimize} && \|Ax - y\|_2^2 + \lambda \|x\|_1 \\ & \text{subject to} && x_1 \leq \dots \leq x_n. \end{aligned}$$

This problem is related to the monotone Lasso [24], which is an extension of isotonic regression which has been extensively studied in statistics [1, 43].

The  $\ell_1$ -regularized isotonic regression problem arises in several contexts. As an example, the problem of finding monotone trends in a Gaussian setting can be cast as a problem of this form [47, 21]. As another example, the problem of estimating accumulating damage trend from a series of structural health monitoring (SHM) images can be formulated as a problem of the form in which the variables are three-dimensional (a time series of images) [22].

We take the change of variables

$$z_1 = x_1, \quad z_k = x_k - x_{k-1}, \quad k = 2, \dots, n,$$

to reformulate the isotonic regression problem as an  $\ell_1$ -regularized LSP with monotonicity constraints:

$$\begin{aligned} & \text{minimize} && \|y - \bar{A}z\|_2^2 + \sum_{k=2}^n \lambda z_k \\ & \text{subject to} && z_k \geq 0, \quad k = 1, \dots, n, \end{aligned}$$

where the new data matrix is

$$\bar{A} = AG, \quad G = \begin{bmatrix} 1 & & & & & \\ 1 & 1 & & & & \\ 1 & 1 & 1 & & & \\ 1 & 1 & 1 & \ddots & & \\ \vdots & \vdots & \vdots & \ddots & 1 & \\ 1 & 1 & 1 & \cdots & 1 & 1 \end{bmatrix} \in \mathbf{R}^{n \times n}.$$

(The entries of  $G$  not shown above are zero.) This problem can be solved by a truncated Newton interior-point method with the preconditioner described above, exploiting the structure of the new data matrix. The two operations, multiplying a vector by  $\bar{A}$  and multiplying a vector by its transpose, can be done as efficiently as those with  $A$ , using the fact that the corresponding operations with  $G$  can be done in  $O(n)$  flops.

## 6.4 $\ell_1$ -regularized LSPs with complex variables

The truncated Newton interior-point method can be extended to a problem of the form

$$\text{minimize} \quad \|Az - b\|_2^2 + \lambda \|z\|_1 \quad (22)$$

where the variable is  $z \in \mathbf{C}^n$  and the problem data are  $b \in \mathbf{C}^n$  and  $A \in \mathbf{C}^{m \times n}$ . Here  $\|z\|_1$ , the  $\ell_1$  norm of the complex vector  $z$ , is defined by

$$\|z\|_1 = \sum_{i=1}^n (x_i^2 + y_i^2)^{1/2}, \quad (23)$$

where  $x$  is the real part of  $z$  and  $y$  is the imaginary part.

We note that (22) cannot be cast as an  $\ell_1$ -regularized LSP of the form (4). Instead, it can be reformulated as the convex problem

$$\begin{aligned} & \text{minimize} && \left\| \tilde{A} \begin{bmatrix} x \\ y \end{bmatrix} - \tilde{b} \right\|_2^2 + \lambda \mathbf{1}^T u \\ & \text{subject to} && \sqrt{x_i^2 + y_i^2} \leq u_i, \quad i = 1, \dots, n, \end{aligned} \quad (24)$$

where the variables are  $x, y, u \in \mathbf{R}^n$  and the problem data are

$$\tilde{A} = \begin{bmatrix} \mathbf{Re} A & -\mathbf{Im} A \\ \mathbf{Im} A & \mathbf{Re} A \end{bmatrix} \in \mathbf{R}^{2m \times 2n}, \quad \tilde{b} = \begin{bmatrix} \mathbf{Re} b \\ \mathbf{Im} b \end{bmatrix} \in \mathbf{R}^{2m}.$$

This problem is a second-order cone program (SOCP) and can be solved by standard interior-point methods; see, *e.g.*, [30] for more on SOCPs.

We show how the truncated Newton interior-point method can be extended to the SOCP (24). Using the standard barrier function for second-order cone constraints, the associated centering problem can be formulated as

$$\text{minimize } t \left\| \tilde{A} \begin{bmatrix} x \\ y \end{bmatrix} - \tilde{b} \right\|_2^2 + t\lambda \mathbf{1}^T u + \sum_{i=1}^n \log(u_i^2 - x_i^2 - y_i^2), \quad (25)$$

with variables  $x, y, u \in \mathbf{R}^n$ . The Hessian of this problem is

$$H = t\tilde{A}^T \tilde{A} + G,$$

where  $G = \nabla^2 \sum_{i=1}^n \log(u_i^2 - x_i^2 - y_i^2)$  is the Hessian of the barrier function. As before, we consider the preconditioner that approximates the Hessian of the first term in the objective of the centering problem (25) with its diagonal entries, while retaining the Hessian of the logarithmic barrier:

$$P = t \text{diag}(\tilde{A}^T \tilde{A}) + G.$$

After appropriate reordering, this preconditioner is a block diagonal matrix consisting of  $n$   $3 \times 3$  diagonal matrices, and so the computational effort of  $P^{-1}r$  is  $O(n)$ . This preconditioner appears to be quite effective in solving the centering problem (25).

In compressed sensing, a problem of the form (22) arises in case when the entries of the matrices  $\Phi$  and  $W$  are complex and we do not expand the real and imaginary parts of the matrices  $\Phi$  and  $W$ . In this case, we have  $A = \Phi W^{-1}$ , which is a matrix with complex entries. The resulting formulation is different from the formulation, considered in §2.1, obtained by expanding the real and imaginary parts of  $z$ ,  $\Phi$ , and  $W$ , with the  $\ell_1$  penalty function  $\sum_{i=1}^n |x_i| + |y_i|$ . Compared to the formulation described in §2.1, the formulation based on the penalty function (23) couples together the real and imaginary parts of entries of  $z$ , so the optimal solution found tends to have more simultaneous zero real and imaginary entries in the entries. (The idea behind the penalty function (23) is used in the grouped Lasso [57] and in total variation minimization with two- or higher-dimensional data [7, 45].)

## 7 Conclusions

In this paper we have described a specialized interior-point method for solving large-scale  $\ell_1$ -regularized LSPs. Numerical experiments show that for sparse problems the method reliably finds accurate solutions with a computational effort that scales well with problem size, up to over one million variables. The method can also handle large dense  $\ell_1$ -regularized LSPs, that arise in sparse signal recovery. It appears to work particularly well, when the  $\ell_1$ -regularized LSP (4) has low mutual coherence (the off-diagonal elements of  $A^T A$  are relatively small compared with the diagonal elements).

The method is similar in spirit to the specialized interior-point method for basis pursuit denoising described in [8]. A major difference is that our method uses the PCG algorithm to

compute the search direction, whereas the method described in [8] uses the LSQR without preconditioning. Another difference lies in the truncation rule for the iterative algorithm: the truncation rule used in our method is more aggressive to find a search direction which gives a good trade-off of computational effort versus the convergence rate it provides.

Our method can be generalized to handle a variety of extensions, as described in §6. The extension to total variation minimization with two- or three-dimensional data is more involved, since it is not straightforward to find a simple but effective preconditioner for the associated centering problem. We hope to extend the method to the total variation minimization problem in future work.

In many applications, the choice of an appropriate value of  $\lambda$  involves computing a portion of the regularization path, or at the very least solving  $\ell_1$ -regularized LSPs with multiple, and often many, values of  $\lambda$ . Incorporating a warm start technique into the truncated Newton interior-point method, we can compute a good approximation of the regularization path much more efficiently than by solving multiple problems independently. The idea has been successfully used in  $\ell_1$ -regularized logistic regression [29].

## Acknowledgments

The authors are grateful to the anonymous reviewers, Emmanuel Candès, Matt Calder, John Duchi, and Michael Grant for helpful comments. This material is based upon work supported by the Focus Center Research Program Center for Circuit & System Solutions award 2003-CT-888, by JPL award I291856, by the Precourt Institute on Energy Efficiency, by Army award W911NF-07-1-0029, by NSF award ECS-0423905, by NSF award 0529426, by DARPA award N66001-06-C-2021, by NASA award NNX07AEI1A, by AFOSR award FA9550-06-1-0514, and by AFOSR award FA9550-06-1-0312.

## References

- [1] R. Barlow, D. Bartholomew, J. Bremner, and H. Brunk. *Statistical Inference Under Order Restrictions; The Theory and Application of Isotonic Regression*. Wiley, New York, 1972.
- [2] J. Borwein and A. Lewis. *Convex Analysis and Nonlinear Optimization*. Springer, 2000.
- [3] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.
- [4] E. Candès. Compressive sampling. In *Proceedings of International Congress of Mathematicians*, 2006.
- [5] E. Candès and J. Romberg.  *$l_1$ -magic: A collection of MATLAB routines for solving the convex optimization programs central to compressive sampling*, 2006. Available from [www.acm.caltech.edu/l1magic/](http://www.acm.caltech.edu/l1magic/).

- [6] E. Candès, J. Romberg, and T. Tao. Stable signal recovery from incomplete and inaccurate measurements. *Communications on Pure and Applied Mathematics*, 59(8):1207–1223, 2005.
- [7] E. Candès, J. Romberg, and T. Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on Information Theory*, 52(2):489–509, 2006.
- [8] S. Chen, D. Donoho, and M. Saunders. Atomic decomposition by basis pursuit. *SIAM Review*, 43(1):129–159, 2001.
- [9] I. Daubechies, M. Defrise, and C. De Mol. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Communications on Pure and Applied Mathematics*, 57:1413–1541, 2004.
- [10] J. Demmel. *Applied Numerical Linear Algebra*. Society for Industrial and Applied Mathematics, 1997.
- [11] D. Donoho. Compressed sensing. *IEEE Transactions on Information Theory*, 52(4):1289–1306, 2006.
- [12] D. Donoho, M. Elad, and V. Temlyakov. Stable recovery of sparse overcomplete representations in the presence of noise. *IEEE Transactions On Information Theory*, 52(1):6–18, 2006.
- [13] D. Donoho, V. Stodden, and Y. Tsaig. *SparseLab: Seeking Sparse Solutions to Linear Systems of Equations*, 2006. Available from <http://sparselab.stanford.edu/>.
- [14] D. Donoho and Y. Tsaig. Fast solution of  $l_1$ -norm minimization problems when the solution may be sparse, 2006. Manuscript. Available from <http://www.stanford.edu/%7Etsaig/Papers/FastL1.pdf>.
- [15] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani. Least angle regression. *Annals of Statistics*, 32(2):407–499, 2004.
- [16] M. Elad, B. Matalon, and M. Zibulevsky. Image denoising with shrinkage and redundant representations. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recogn (CVPR)*, volume 2, pages 1924–1931, 2006.
- [17] M. Figueiredo and R. Nowak. A bound optimization approach to wavelet-based image deconvolution. In *Proceedings of IEEE International Conference on Image Processing (ICIP)*, pages 782–5, 2005.
- [18] M. Figueiredo, R. Nowak, and S. Wright. Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems, 2007. Appeared in the same issue of *IEEE Journal on Selected Topics in Signal Processing*.

- [19] J. Friedman, T. Hastie T, and R. Tibshirani. Pathwise coordinate optimization, 2007. Manuscript available from [www-stat.stanford.edu/~hastie/pub.htm](http://www-stat.stanford.edu/~hastie/pub.htm).
- [20] J. Fuchs. Recovery of exact sparse representations in the presence of noise. *IEEE Transactions on Information Theory*, 51(10):3601–3608, 2005.
- [21] D. Gorinevsky. Monotonic regression filters for trending gradual deterioration faults. In *Proceedings of American Control Conference (ACC)*, pages 5394–5399, 2004.
- [22] D. Gorinevsky, S.-J. Kim, S. Bear, S. Boyd, and G. Gordon. Optimal estimation of deterioration from diagnostic image sequence, 2007. Submitted to *IEEE Transactions on Signal Processing*. Available from <http://www.stanford.edu/~gorin/papers/MonoImage07twocol.pdf>.
- [23] T. Hastie, S. Rosset, R. Tibshirani, and J. Zhu. The entire regularization path for the support vector machine. *Journal of Machine Learning Research*, 5:1391–1415, 2004.
- [24] T. Hastie, J. Taylor, R. Tibshirani, and G. Walther. Forward stagewise regression and the monotone lasso. *Electronic Journal of Statistics*, 1:1–29, 2007.
- [25] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer Series in Statistics. Springer-Verlag, New York, 2001.
- [26] C. Johnson, J. Seidel, and A. Sofer. Interior point methodology for 3-D PET reconstruction. *IEEE Transactions on Medical Imaging*, 19(4):271–285, 2000.
- [27] C. T. Kelley. *Iterative Methods for Linear and Nonlinear Equations*, volume 16 of *Frontiers in Applied Mathematics*. SIAM, Philadelphia, 1995.
- [28] K. Knight and W. Fu. Asymptotics for Lasso-type estimators. *Annals of Statistics*, 28(5):1356–1378, 2000.
- [29] K. Koh, S.-J. Kim, and S. Boyd. An interior-point method for  $\ell_1$ -regularized logistic regression. *Journal of Machine Learning Research*, 8:1519–1555, 2007.
- [30] M. Lobo, L. Vandenbergh, S. Boyd, and H. Lebrecht. Applications of second-order cone programming. *Linear Algebra and its Applications*, 284:193–228, 1998.
- [31] D. Luenberger. *Linear and Nonlinear Programming*. Addison-Wesley, second edition, 1984.
- [32] N. Meinshausen and P. Bühlmann. High-dimensional graphs and variable selection with the Lasso. *Annals of Statistics*, 34(3):1436–1462, 2006.
- [33] MOSEK ApS. *The MOSEK Optimization Tools Version 2.5. User’s Manual and Reference*, 2002. Available from [www.mosek.com](http://www.mosek.com).



- [34] G. Narkiss and M. Zibulevsky. Sequential subspace optimization method for large-scale unconstrained problems, 2005. Tech. Report CCIT No 559, EE Dept., Technion.
- [35] Y. Nesterov. Gradient methods for minimizing composite objective function, 2007. CORE Discussion Paper 2007/76. Available from [http://www.optimization-online.org/DB\\_HTML/2007/09/1784.html](http://www.optimization-online.org/DB_HTML/2007/09/1784.html).
- [36] Y. Nesterov and A. Nemirovsky. *Interior-Point Polynomial Methods in Convex Programming*, volume 13 of *Studies in Applied Mathematics*. SIAM, Philadelphia, PA, 1994.
- [37] A. Neumaier. Solving ill-conditioned and singular linear systems: A tutorial on regularization. *SIAM Review*, 40(3):636–666, 1998.
- [38] J. Nocedal and S. Wright. *Numerical Optimization*. Springer Series in Operations Research. Springer, 1999.
- [39] M. Osborne, B. Presnell, and B. Turlach. A new approach to variable selection in least squares problems. *IMA Journal of Numerical Analysis*, 20(3):389–403, 2000.
- [40] C. Paige and M. Saunders. LSQR: An algorithm for sparse linear equations and sparse least squares. *ACM Transactions on Mathematical Software*, 8(1):43–71, 1982.
- [41] B. Polyak. *Introduction to Optimization*. Optimization Software, 1987. Translated from Russian.
- [42] L. Portugal, M. Resende, G. Veiga, and J. Júdice. A truncated primal-infeasible dual-feasible network interior point method, 1994.
- [43] T. Robertson, F. Wright, R., and Dykstra. *Order Restricted Statistical Inference*. Wiley, New York, 1988.
- [44] S. Rosset. Tracking curved regularized optimization solution paths. In L. Saul, Y. Weiss, and L. Bottou, editors, *Advances in Neural Information Processing Systems 17*. MIT Press, Cambridge, MA, 2005.
- [45] L. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physics D*, 60:259–268, 1992.
- [46] Y. Saad. *Iterative Methods for Sparse Linear Systems*. SIAM, Philadelphia, 2nd edition, 2003.
- [47] S. Samar, D. Gorinevsky, and S. Boyd. Moving horizon filter for monotonic trends. In *Proceedings of the 43rd IEEE Conference on Decision and Control*, volume 3, pages 3115–3120, 2004.
- [48] M. Saunders. *PDCO: Primal-Dual interior method for Convex Objectives*, 2002. Available from <http://www.stanford.edu/group/SOL/software/pdco.html>.

- [49] N. Z. Shor. *Minimization Methods for Non-differentiable Functions*. Springer Series in Computational Mathematics. Springer, 1985.
- [50] R. Tibshirani. Regression shrinkage and selection via the Lasso. *Journal of the Royal Statistical Society, Series B*, 58(1):267–288, 1996.
- [51] J. Tropp. Just relax: Convex programming methods for identifying sparse signals in noise. *IEEE Transactions on Information Theory*, 52(3):1030–1051, 2006.
- [52] Y. Tsaig and D. Donoho. Extensions of compressed sensing. *Signal Processing*, 86(3):549–571, 2006.
- [53] L. Vandenberghe and S. Boyd. A primal-dual potential reduction method for problems involving matrix inequalities. *Mathematical Programming, Series B*, 69:205–236, 1995.
- [54] M. Wainwright. Sharp thresholds for high-dimensional and noisy recovery of sparsity. In *Proceedings of the Forty-Fourth Annual Allerton Conference on Communication, Control, and Computing*, 2006.
- [55] S. Wright. *Primal-dual interior-point methods*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 1997.
- [56] Y. Ye. *Interior Point Algorithms: Theory and Analysis*. John Wiley & Sons, 1997.
- [57] M. Yuan and L. Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society, Series B*, 68(1):49–67, 2006.
- [58] P. Zhao and B. Yu. On model selection consistency of Lasso. *Journal of Machine Learning Research*, 7:2541–2563, 2006.
- [59] H. Zou. The adaptive Lasso and its oracle properties. *Journal of the American Statistical Association*, 101(476):1418–1429, 2006.
- [60] H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society, Series B*, 67(2):301–320, 2005.