J Fourier Anal Appl (2008) 14: 877–905
DOI 10.1007/s00041-008-9045-x

Enhancing Sparsity by Reweighted ℓ_1 Minimization

Emmanuel J. Candès · Michael B. Wakin · Stephen P. Boyd

Received: 10 October 2007 / Published online: 15 October 2008

Abstract It is now well understood that (1) it is possible to reconstruct sparse signals exactly from what appear to be highly incomplete sets of linear measurements and (2) that this can be done by constrained ℓ_1 minimization. In this paper, we study a novel method for sparse signal recovery that in many situations outperforms ℓ_1 minimization in the sense that substantially fewer measurements are needed for exact recovery. The algorithm consists of solving a sequence of weighted ℓ_1-minimization problems where the weights used for the next iteration are computed from the value of the current solution. We present a series of experiments demonstrating the remarkable performance and broad applicability of this algorithm in the areas of sparse signal recovery, statistical estimation, error correction and image processing. Interestingly, superior gains are also achieved when our method is applied to recover signals with assumed near-sparsity in overcomplete representations—not by reweighting the ℓ_1 norm of the coefficient sequence as is common, but by reweighting the ℓ_1 norm of the transformed object. An immediate consequence is the possibility of highly efficient data acquisition protocols by improving on a technique known as Compressive Sensing.

Communicated by Albert Cohen.

E.J. Candès
Applied and Computational Mathematics, Caltech, Pasadena, CA 91125, USA
e-mail: emmanuel@acm.caltech.edu

M.B. Wakin (✉)
Engineering, Colorado School of Mines, Golden, CO 80401, USA
e-mail: mwakin@mines.edu

S.P. Boyd
Department of Electrical Engineering, Stanford University, Stanford, CA 94305, USA
e-mail: boyd@stanford.edu

Keywords ℓ_1-Minimization · Iterative reweighting · Underdetermined systems of linear equations · Compressive sensing · Dantzig selector · Sparsity · FOCUSS

1 Introduction

What makes some scientific or engineering problems at once interesting and challenging is that often, one has fewer equations than unknowns. When the equations are linear, one would like to determine an object $x_0 \in \mathbb{R}^n$ from data $y = \Phi x_0$, where Φ is an $m \times n$ matrix with fewer rows than columns; i.e., $m < n$. The problem is of course that a system with fewer equations than unknowns usually has infinitely many solutions and thus, it is apparently impossible to identify which of these candidate solutions is indeed the "correct" one without some additional information.

In many instances, however, the object we wish to recover is known to be structured in the sense that it is sparse or compressible. This means that the unknown object depends upon a smaller number of unknown parameters. In a biological experiment, one could measure changes of expression in 30,000 genes and expect at most a couple hundred genes with a different expression level. In signal processing, one could sample or sense signals which are known to be sparse (or approximately so) when expressed in the correct basis. This premise radically changes the problem, making the search for solutions feasible since the simplest solution now tends to be the right one.

Mathematically speaking and under sparsity assumptions, one would want to recover a signal $x_0 \in \mathbb{R}^n$, e.g., the coefficient sequence of the signal in the appropriate basis, by solving the combinatorial optimization problem

$$(\mathrm{P}_0) \quad \min_{x \in \mathbb{R}^n} \|x\|_{\ell_0} \quad \text{subject to} \quad y = \Phi x, \tag{1}$$

where $\|x\|_{\ell_0} = |\{i : x_i \neq 0\}|$. This is a common sense approach which simply seeks the simplest explanation fitting the data. In fact, this method can recover sparse solutions even in situations in which $m \ll n$. Suppose for example that all sets of m columns of Φ are in general position. Then the program (P_0) perfectly recovers all sparse signals x_0 obeying $\|x_0\|_{\ell_0} \leq m/2$. This is of little practical use, however, since the optimization problem (1) is nonconvex and generally impossible to solve as its solution usually requires an intractable combinatorial search.

A common alternative is to consider the convex problem

$$(\mathrm{P}_1) \quad \min_{x \in \mathbb{R}^n} \|x\|_{\ell_1} \quad \text{subject to} \quad y = \Phi x, \tag{2}$$

where $\|x\|_{\ell_1} = \sum_{i=1}^n |x_i|$. Unlike (P_0), this problem is convex—it can actually be recast as a linear program—and is solved efficiently [1]. The programs (P_0) and (P_1) differ only in the choice of objective function, with the latter using an ℓ_1 norm as a proxy for the literal ℓ_0 sparsity count. As summarized below, a recent body of work has shown that perhaps surprisingly, there are conditions guaranteeing a formal equivalence between the combinatorial problem (P_0) and its relaxation (P_1).

The use of the ℓ_1 norm as a sparsity-promoting functional traces back several decades. A leading early application was reflection seismology, in which a sparse reflection function (indicating meaningful changes between subsurface layers) was sought from bandlimited data. In 1979, Taylor, Banks and McCoy [2] proposed the use of ℓ_1 to deconvolve seismic traces by improving on earlier ideas of Claerbout and Muir [3]. Over the next decade this idea was refined to better handle observation noise [4], and the sparsity-promoting nature of ℓ_1 minimization was empirically confirmed. Rigorous results began to appear in the late-1980's, with Donoho and Stark [5] and Donoho and Logan [6] quantifying the ability to recover sparse reflectivity functions. The application areas for ℓ_1 minimization began to broaden in the mid-1990's, as the LASSO algorithm [7] was proposed as a method in statistics for sparse model selection, Basis Pursuit [8] was proposed in computational harmonic analysis for extracting a sparse signal representation from highly overcomplete dictionaries, and a related technique known as total variation minimization was proposed in image processing [9, 10].

Some examples of ℓ_1 type methods for sparse design in engineering include Vandenberghe et al. [11, 12] for designing sparse interconnect wiring, and Hassibi et al. [13] for designing sparse control system feedback gains. In [14], Dahleh and Diaz-Bobillo solve controller synthesis problems with an ℓ_1 criterion, and observe that the optimal closed-loop responses are sparse. Lobo et al. used ℓ_1 techniques to find sparse trades in portfolio optimization with fixed transaction costs in [15]. In [16], Ghosh and Boyd used ℓ_1 methods to design well connected sparse graphs; in [17], Sun et al. observe that optimizing the rates of a Markov process on a graph leads to sparsity. In [1, Sect. 6.5.4 and 11.4.1], Boyd and Vandenberghe describe several problems involving ℓ_1 methods for sparse solutions, including finding small subsets of mutually infeasible inequalities, and points that violate few constraints. In a recent paper, Koh et al. used these ideas to carry out piecewise-linear trend analysis [18].

Over the last decade, the applications and understanding of ℓ_1 minimization have continued to increase dramatically. Donoho and Huo [19] provided a more rigorous analysis of Basis Pursuit, and this work was extended and refined in subsequent years, see [20–22]. Much of the recent focus on ℓ_1 minimization, however, has come in the emerging field of Compressive Sensing [23–25]. This is a setting where one wishes to recover a signal x_0 from a small number of compressive measurements $y = \Phi x_0$. It has been shown that ℓ_1 minimization allows recovery of sparse signals from remarkably few measurements [26, 27]: supposing Φ is chosen randomly from a suitable distribution, then with very high probability, all sparse signals x_0 for which $\|x_0\|_{\ell_0} \leq m/\alpha$ with $\alpha = O(\log(n/m))$ can be *perfectly* recovered by using (P_1). Moreover, it has been established [27] that Compressive Sensing is robust in the sense that ℓ_1 minimization can deal very effectively (a) with only approximately sparse signals and (b) with measurement noise. The implications of these facts are quite far-reaching, with potential applications in data compression [24, 28], digital photography [29], medical imaging [23, 30], error correction [31, 32], analog-to-digital conversion [33], sensor networks [34, 35], and so on. (We will touch on some more concrete examples in Sect. 3.)

The use of ℓ_1 regularization has become so widespread that it could arguably be considered the “modern least squares”. This raises the question of whether we can improve upon ℓ_1 minimization? It is natural to ask, for example, whether a different (but perhaps again convex) alternative to ℓ_0 minimization might also find the correct solution, but with a lower measurement requirement than ℓ_1 minimization.

In this paper, we consider one such alternative, which aims to help rectify a key difference between the ℓ_1 and ℓ_0 norms, namely, the dependence on magnitude: larger coefficients are penalized more heavily in the ℓ_1 norm than smaller coefficients, unlike the more democratic penalization of the ℓ_0 norm. To address this imbalance, we propose a weighted formulation of ℓ_1 minimization designed to more democratically penalize nonzero coefficients. In Sect. 2, we discuss an iterative algorithm for constructing the appropriate weights, in which each iteration of the algorithm solves a convex optimization problem, whereas the overall algorithm does not. Instead, this iterative algorithm attempts to find a local minimum of a concave penalty function that more closely resembles the ℓ_0 norm. Finally, we would like to draw attention to the fact that each iteration of this algorithm simply requires solving one ℓ_1 minimization problem, and so the method can be implemented readily using existing software.

In Sect. 3, we present a series of experiments demonstrating the superior performance and broad applicability of this algorithm, not only for recovery of sparse signals, but also pertaining to compressible signals, noisy measurements, error correction, and image processing. This section doubles as a brief tour of the applications of Compressive Sensing. In Sect. 4, we demonstrate the promise of this method for efficient data acquisition. Finally, we conclude in Sect. 5 with a final discussion of related work and future directions.

2 An Iterative Algorithm for Reweighted ℓ_1 Minimization

2.1 Weighted ℓ_1 Minimization

Consider the "weighted" ℓ_1 minimization problem

$$(\mathrm{WP}_1) \quad \min_{x\in\mathbb{R}^n} \sum_{i=1} w_i|x_i| \quad \text{subject to} \quad y=\Phi x, \tag{3}$$

where $w_1, w_2, \ldots, w_n$ are positive weights. Just like its "unweighted" counterpart (P_1), this convex problem can be recast as a linear program. In the sequel, it will be convenient to denote the objective functional by $\|Wx\|_{\ell_1}$ where W is the diagonal matrix with $w_1, \ldots, w_n$ on the diagonal and zeros elsewhere.

The weighted ℓ_1 minimization (WP_1) can be viewed as a relaxation of a weighted ℓ_0 minimization problem

$$(\mathrm{WP}_0) \quad \min_{x\in\mathbb{R}^n} \|Wx\|_{\ell_0} \quad \text{subject to} \quad y=\Phi x. \tag{4}$$

Whenever the solution to (P_0) is unique, it is also the unique solution to (WP_0) provided that the weights do not vanish. However, the corresponding ℓ_1 relaxations (P_1) and (WP_1) will have different solutions in general. Hence, one may think of the weights (w_i) as free parameters in the convex relaxation, whose values—if set wisely—could improve the signal reconstruction.

This raises the immediate question: what values for the weights will improve signal reconstruction? One possible use for the weights could be to counteract the influence of the signal magnitude on the ℓ_1 penalty function. Suppose, for example, that

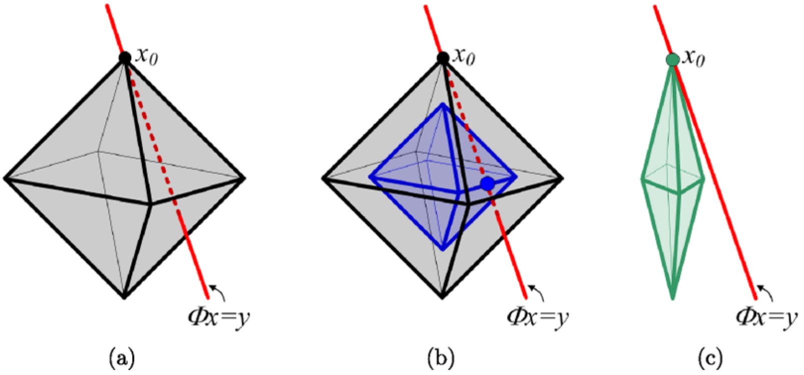


Fig. 1 Weighting ℓ_1 minimization to improve sparse signal recovery. (**a**) Sparse signal x_0, feasible set $\Phi x = y$, and ℓ_1 ball of radius $\|x_0\|_{\ell_1}$. (**b**) There exists an $x \neq x_0$ for which $\|x\|_{\ell_1} < \|x_0\|_{\ell_1}$. (**c**) Weighted ℓ_1 ball. There exists no $x \neq x_0$ for which $\|Wx\|_{\ell_1} \leq \|Wx_0\|_{\ell_1}$

the weights were inversely proportional to the true signal magnitude, i.e., that

$$w_i = \begin{cases} \frac{1}{|x_{0,i}|}, & x_{0,i} \neq 0, \\ \infty, & x_{0,i} = 0. \end{cases} \tag{5}$$

If the true signal x_0 is k-sparse, i.e., obeys $\|x_0\|_{\ell_0} \leq k$, then (WP$_1$) is guaranteed to find the correct solution with this choice of weights, assuming only that $m \geq k$ and that just as before, the columns of Φ are in general position. The large (actually infinite) entries in w_i force the solution x to concentrate on the indices where w_i is small (actually finite), and by construction these correspond precisely to the indices where x_0 is nonzero. It is of course impossible to construct the precise weights (5) without knowing the signal x_0 itself, but this suggests more generally that large weights could be used to discourage nonzero entries in the recovered signal, while small weights could be used to encourage nonzero entries.

For the sake of illustration, consider the simple 3-D example in Fig. 1, where $x_0 = [0\ 1\ 0]^T$ and

$$\Phi = \begin{bmatrix} 2 & 1 & 1 \\ 1 & 1 & 2 \end{bmatrix}.$$

We wish to recover x_0 from $y = \Phi x_0 = [1\ 1]^T$. Figure 1(a) shows the original signal x_0, the set of points $x \in \mathbb{R}^3$ obeying $\Phi x = \Phi x_0 = y$, and the ℓ_1 ball of radius 1 centered at the origin. The interior of the ℓ_1 ball intersects the feasible set $\Phi x = y$, and thus (P$_1$) finds an incorrect solution, namely, $x^\star = [1/3\ 0\ 1/3]^T \neq x_0$ (see Fig. 1(b)).

Consider now a hypothetical weighting matrix $W = \mathrm{diag}([3\ 1\ 3]^T)$. Figure 1(c) shows the "weighted ℓ_1 ball" of radius $\|Wx\|_{\ell_1} = 1$ centered at the origin. Compared to the unweighted ℓ_1 ball (Fig. 1(a)), this ball has been sharply pinched at x_0. As a result, the interior of the weighted ℓ_1 ball does not intersect the feasible set, and consequently, (WP$_1$) will find the correct solution $x^\star = x_0$. Indeed, it is not difficult to show that the same statements would hold true for any positive weighting matrix for which $w_2 < (w_1 + w_3)/3$. Hence there is a range of valid weights for which (WP_1) will find the correct solution. As a rough rule of thumb, the weights should relate inversely to the true signal magnitudes.

2.2 An Iterative Algorithm

The question remains of how a valid set of weights may be obtained without first knowing x_0. As Fig. 1 shows, there may exist a range of favorable weighting matrices W for each fixed x_0, which suggests the possibility of constructing a favorable set of weights based solely on an approximation x to x_0 or on other side information about the vector magnitudes.

We propose a simple iterative algorithm that alternates between estimating x_0 and redefining the weights. The algorithm is as follows:

1. Set the iteration count ℓ to zero and $w_i^{(0)} = 1, i = 1, \dots, n$.
2. Solve the weighted ℓ_1 minimization problem

$$x^{(\ell)} = \arg\min \|W^{(\ell)} x\|_{\ell_1} \quad \text{subject to} \quad y = \Phi x. \tag{6}$$

3. Update the weights: for each $i = 1, \dots, n$,

$$w_i^{(\ell+1)} = \frac{1}{|x_i^{(\ell)}| + \epsilon}. \tag{7}$$

4. Terminate on convergence or when ℓ attains a specified maximum number of iterations $\ell_{\max}$. Otherwise, increment ℓ and go to step 2.

We introduce the parameter $\epsilon > 0$ in step 3 in order to provide stability and to ensure that a zero-valued component in $x^{(\ell)}$ does not strictly prohibit a nonzero estimate at the next step. As empirically demonstrated in Sect. 3, ϵ should be set slightly smaller than the expected nonzero magnitudes of x_0. In general, the recovery process tends to be reasonably robust to the choice of ϵ.

Using an iterative algorithm to construct the weights (w_i) tends to allow for successively better estimation of the nonzero coefficient locations. Even though the early iterations may find inaccurate signal estimates, the largest signal coefficients are most likely to be identified as nonzero. Once these locations are identified, their influence is downweighted in order to allow more sensitivity for identifying the remaining small but nonzero signal coefficients.

Figure 2 illustrates this dynamic by means of an example in sparse signal recovery. Figure 2(a) shows the original signal of length $n = 512$, which contains 130 nonzero spikes. We collect $m = 256$ measurements where the matrix Φ has independent standard normal entries. We set $\epsilon = 0.1$ and $\ell_{\max} = 2$. Figures 2(b)–(d) show scatter plots, coefficient-by-coefficient, of the original signal coefficient x_0 versus its reconstruction $x^{(\ell)}$. In the unweighted iteration (Fig. 2(b)), we see that all large coefficients in x_0 are properly identified as nonzero (with the correct sign), and that $\|x_0 - x^{(0)}\|_{\ell_\infty} = 0.4857$. In this first iteration, $\|x^{(0)}\|_{\ell_0} = 256 = m$, with 15 nonzero spikes in x_0 reconstructed as zeros and 141 zeros in x_0 reconstructed as nonzeros. These numbers improve after one reweighted iteration (Fig. 2(c)) with now

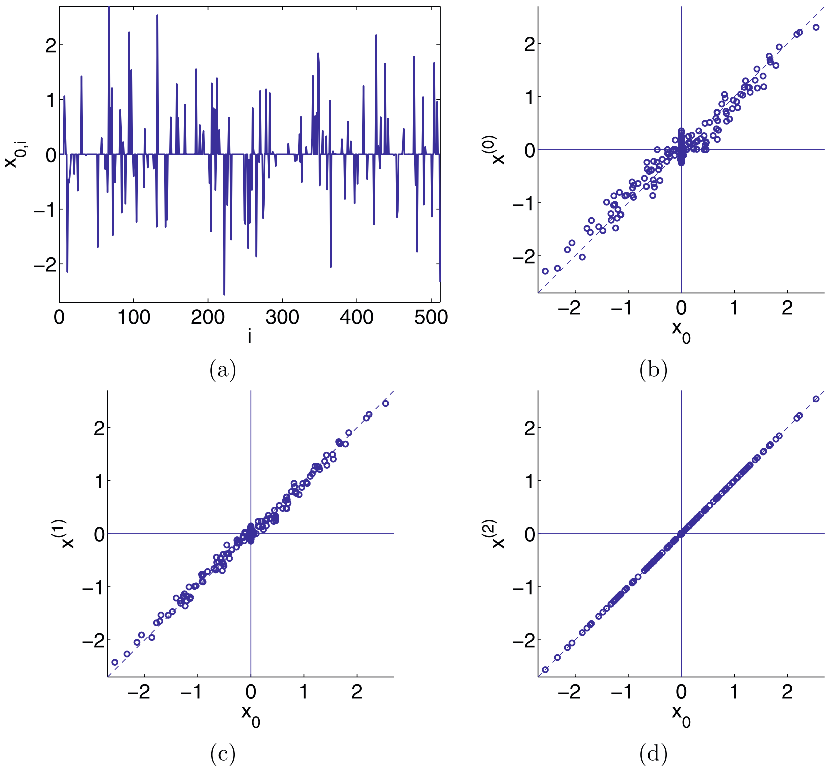


Fig. 2 Sparse signal recovery through reweighted ℓ_1 iterations. (**a**) Original length $n = 512$ signal x_0 with 130 spikes. (**b**) Scatter plot, coefficient-by-coefficient, of x_0 versus its reconstruction $x^{(0)}$ using unweighted ℓ_1 minimization. (**c**) Reconstruction $x^{(1)}$ after the first reweighted iteration. (**d**) Reconstruction $x^{(2)}$ after the second reweighted iteration

$\|x - x^{(1)}\|_{\ell_\infty} = 0.2407$, $\|x^{(1)}\|_{\ell_0} = 256 = m$, 6 nonzero spikes in x_0 reconstructed as zeros and 132 zeros in x_0 reconstructed as nonzeros. This improved signal estimate is then sufficient to allow perfect recovery in the second reweighted iteration (Fig. 2(d)).

2.3 Analytical Justification

The iterative reweighted algorithm falls in the general class of Majorization-Minimization (MM) algorithms, see [36] and references therein. In a nutshell, MM algorithms are more general than EM algorithms, and work by iteratively minimizing a simple surrogate function majorizing a given objective function. To establish this connection, consider the problem

$$\min_{x \in \mathbb{R}^n} \sum_{i=1}^{n} \log(|x_i| + \epsilon) \quad \text{subject to} \quad y = \Phi x, \tag{8}$$

which is equivalent to

$$\min_{x,u\in\mathbb{R}^n} \sum_{i=1}^n \log(u_i+\epsilon) \quad \text{subject to} \quad y=\Phi x, |x_i|\le u_i,\ i=1,\dots,n. \tag{9}$$

The equivalence means that if $x^\star$ is a solution to (8), then $(x^\star, |x^\star|)$ is a solution to (9). And conversely, if $(x^\star, u^\star)$ is a solution to (9), then $x^\star$ is a solution to (8).

Problem (9) is of the general form

$$\min_v\ g(v) \quad \text{subject to} \quad v\in\mathcal{C},$$

where $\mathcal{C}$ is a convex set. In (9), the function g is concave and, therefore, below its tangent. Thus, one can improve on a guess v at the solution by minimizing a linearization of g around v. This simple observation yields the following MM algorithm: starting with $v^{(0)}\in\mathcal{C}$, inductively define

$$v^{(\ell+1)} = \arg\min\ g(v^{(\ell)}) + \nabla g(v^{(\ell)})\cdot(v-v^{(\ell)}) \quad \text{subject to} \quad v\in\mathcal{C}.$$

Each iterate is now the solution to a convex optimization problem. In the case (9) of interest, this gives

$$\begin{aligned}(x^{(\ell+1)}, u^{(\ell+1)}) &= \arg\min \sum_{i=1}^n \frac{u_i}{u_i^{(\ell)}+\epsilon}\\ \text{subject to} \quad y &= \Phi x, \quad |x_i|\le u_i,\ i=1,\dots,n,\end{aligned}$$

which is of course equivalent to

$$x^{(\ell+1)} = \arg\min \sum_{i=1}^n \frac{|x_i|}{|x_i^{(\ell)}|+\epsilon} \quad \text{subject to} \quad y=\Phi x.$$

One now recognizes our iterative algorithm.

In the Ph.D. thesis [37] and two subsequent papers [38, 39], Fazel et al. have considered the same reweighted ℓ_1 minimization algorithm as in Sect. 2.2, first as a heuristic algorithm for applications in portfolio optimization [38], and second as a special case of an iterative algorithm for minimizing the rank of a matrix subject to convex constraints [39]. Using general theory, they argue that $\sum_{i=1}^n \log(|x_i^{(\ell)}|+\epsilon)$ converges to a local minimum of $g(x)=\sum_{i=1}^n \log(|x_i|+\epsilon)$ (note that this not saying that the sequence $(x^{(\ell)})$ converges). Because the log-sum penalty function is concave, one cannot expect this algorithm to always find a global minimum. As a result, it is important to choose a suitable starting point for the algorithm. Like [39], we have suggested initializing with the solution to (P_1), the unweighted ℓ_1 minimization. In practice we have found this to be an effective strategy. Further connections between our work and FOCUSS strategies are discussed at the end of the paper.

The connection with the log-sum penalty function provides a basis for understanding why reweighted ℓ_1 minimization can improve the recovery of sparse signals. In

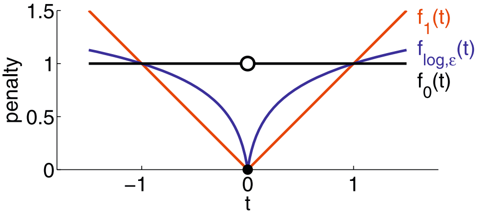


Fig. 3 At the origin, the canonical ℓ_0 sparsity count $f_0(t)$ is better approximated by the log-sum penalty function $f_{\log,\epsilon}(t)$ than by the traditional convex ℓ_1 relaxation $f_1(t)$

particular, the log-sum penalty function has the potential to be much more sparsity-encouraging than the ℓ_1 norm. Consider, for example, three potential penalty functions for scalar magnitudes t:

$$f_0(t) = 1_{\{t\neq 0\}}, \qquad f_1(t) = |t| \quad \text{and} \quad f_{\log,\epsilon}(t) \propto \log(1 + |t|/\epsilon),$$

where the constant of proportionality is set such that $f_{\log,\epsilon}(1) = 1 = f_0(1) = f_1(1)$, see Fig. 3. The first (ℓ_0-like) penalty function f_0 has infinite slope at $t = 0$, while its convex (ℓ_1-like) relaxation f_1 has unit slope at the origin. The concave penalty function $f_{\log,\epsilon}$, however, has slope at the origin that grows roughly as $1/\epsilon$ when $\epsilon \to 0$. Like the ℓ_0 norm, this allows a relatively large penalty to be placed on small nonzero coefficients and more strongly encourages them to be set to zero. In fact, $f_{\log,\epsilon}(t)$ tends to $f_0(t)$ as $\epsilon \to 0$. Following this argument, it would appear that ϵ should be set arbitrarily small, to most closely make the log-sum penalty resemble the ℓ_0 norm. Unfortunately, as $\epsilon \to 0$, it becomes more likely that the iterative reweighted ℓ_1 algorithm will get stuck in an undesirable local minimum. As shown in Sect. 3, a cautious choice of ϵ (slightly smaller than the expected nonzero magnitudes of x) provides the stability necessary to correct for inaccurate coefficient estimates while still improving upon the unweighted ℓ_1 algorithm for sparse recovery.

2.4 Variations

One could imagine a variety of possible reweighting functions in place of (7). We have experimented with alternatives, including a binary (large/small) setting of w_i depending on the current guess. Though such alternatives occasionally provide superior reconstruction of sparse signals, we have found the rule (7) to perform well in a variety of experiments and applications.

Alternatively, one can attempt to minimize a concave function other than the log-sum penalty. For instance, we may consider

$$g(x) = \sum_{i=1}^{n} \operatorname{atan}(|x_i|/\epsilon)$$

in lieu of $\sum_{i=1}^{n} \log(1 + |x_i|/\epsilon)$. The function atan is bounded above and ℓ_0-like. If x is the current guess, this proposal updates the sequence of weights as $w_i = 1/(x_i^2 + \epsilon^2)$. There are of course many possibilities of this nature and they tend to work well (sometimes better than the log-sum penalty). As one example, Figueiredo et al. [40, 41] have proposed an MM algorithm for solving penalized likelihood signal restoration problems of the form

$$\min_{x\in\mathbb{R}^n} \|y-\Phi x\|_2^2+\lambda\|x\|_p^p, \quad 0<p<1.$$

Using an ℓ_1 penalty to majorize the nonconvex ℓ_p penalty, the resulting algorithm can be interpreted as a sequence of weighted ℓ_1 minimization problems, with each set of weights depending on the previous iteration. As a second example, Wipf and Nagarajan [42] have proposed an iterative reweighted ℓ_1 minimization algorithm for a feature selection problem known as automatic relevance determination (ARD). In this case, it is also shown that ARD is equivalent to solving a MAP estimation problem of the form

$$\min_{x\in\mathbb{R}^n} \|y-\Phi x\|_2^2+g(x),$$

where $g(x)$ is a concave, non-decreasing function that is generally not factorable into individual components x_i. Zou [43] has also proposed an iterative reweighted version of the LASSO algorithm, while Zou and Li [44] have examined connections with a penalty similar to the log-sum. Because of space limitations, however, we will limit ourselves to empirical studies of the performance of the log-sum penalty.

2.5 Historical Progression

The development of the reweighted ℓ_1 algorithm has an interesting historical parallel with the use of Iteratively Reweighted Least Squares (IRLS) for robust statistical estimation [45–47]. Consider a regression problem $Ax=b$ where the observation matrix A is overdetermined. It was noticed that standard least squares regression, in which one minimizes $\|r\|_2$ where $r=Ax-b$ is the residual vector, lacked robustness vis a vis outliers. To defend against this, IRLS was proposed as an iterative method to minimize instead the objective

$$\min_x \sum_i \rho(r_i(x)),$$

where $\rho(\cdot)$ is a penalty function such as the ℓ_1 norm [45, 48]. This minimization can be accomplished by solving a sequence of weighted least-squares problems where the weights $\{w_i\}$ depend on the previous residual $w_i=\rho'(r_i)/r_i$. For typical choices of ρ this dependence is in fact inversely proportional—large residuals will be penalized less in the subsequent iteration and vice versa—as is the case with our reweighted ℓ_1 algorithm. Interestingly, just as IRLS involved iteratively reweighting the ℓ_2-norm in order to better approximate an ℓ_1-like criterion, our algorithm involves iteratively reweighting the ℓ_1-norm in order to better approximate an ℓ_0-like criterion.

3 Numerical Experiments

We present a series of experiments demonstrating the benefits of reweighting the ℓ_1 penalty. We will see that the requisite number of measurements to recover or approximate a signal is typically reduced, in some cases by a substantial amount. We also demonstrate that the reweighting approach is robust and broadly applicable, providing examples of sparse and compressible signal recovery, noise-aware recovery, model selection, error correction, and 2-dimensional total-variation minimization. Meanwhile, we address important issues such as how one can choose ϵ wisely and how robust is the algorithm to this choice, and how many reweighting iterations are needed for convergence.

Throughout Sects. 3 and 4, to implement Step 2 of each reweighted algorithm, we use the existing solver in the ℓ_1-MAGIC software package (available at www.l1-magic.org) or make straightforward modifications therein. For example, the optimization problem (6) can be posed equivalently as

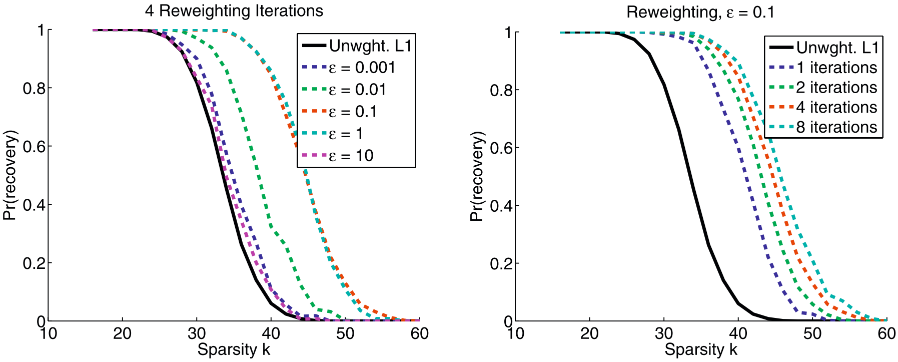
$$z^{(\ell)} = \arg\min \|z\|_{\ell_1} \quad \text{subject to} \quad y = \Phi(W^{(\ell)})^{-1} z,$$

which can be fed directly into ℓ_1-MAGIC; one must then let the signal estimate $x^{(\ell)} = (W^{(\ell)})^{-1} z^{(\ell)}$. The algorithms in ℓ_1-MAGIC are based on standard primal-dual and log-barrier methods [1] for solving linear programs (LPs) and second-order cone programs (SOCPs), respectively. To give the reader a sense of the added computational complexity over an unweighted implementation, we indicate the number $\ell_{\max}$ of reweighting iterations below—that is, the total number of calls to Step 2 of the reweighted algorithm. The computational complexity of any individual call to Step 2 depends on the complexity of solving a single LP or SOCP, and for the type of problems common in Compressive Sensing, the structure of the operator Φ often facilitates fast solutions using on the order of $1000 n \log n$ operations. Other fast solvers abound that could also be adapted to reweighting (e.g., [49]); in general, the development and theory of fast algorithms for ℓ_1-minimization remains a topic of active research.

3.1 Sparse Signal Recovery

The purpose of this first experiment is to demonstrate (1) that reweighting reduces the necessary sampling rate for sparse signals (2) that this recovery is robust with respect to the choice of ϵ and (3) that few reweighting iterations are typically needed in practice. The setup for each trial is as follows. We select a sparse signal x_0 of length $n = 256$ with $\|x_0\|_{\ell_0} = k$. The k nonzero spike positions are chosen randomly, and the nonzero values are chosen randomly from a zero-mean unit-variance Gaussian distribution. We set $m = 100$ and sample a random $m \times n$ matrix Φ with i.i.d. Gaussian entries, giving the data $y = \Phi x_0$. To recover the signal, we run several reweighting iterations with equality constraints (see Sect. 2.2). The parameter ϵ remains fixed during these iterations. Finally, we run 500 trials for various fixed combinations of k and ϵ.

Figure 4(a) compares the performance of unweighted ℓ_1 to reweighted ℓ_1 for various values of the parameter ϵ. The solid line plots the probability of perfect signal recovery (declared when $\|x_0 - x\|_{\ell_\infty} \le 10^{-3}$) for the unweighted ℓ_1 algorithm as a function of the sparsity level k. The dashed curves represent the performance after 4 reweighted iterations for several different values of the parameter ϵ. We see a marked improvement over the unweighted ℓ_1 algorithm; with the proper choice of ϵ, the requisite oversampling factor m/k for perfect signal recovery has dropped from approximately $100/25 = 4$ to approximately $100/33 \approx 3$. This improvement is also



(a) (b)

Fig. 4 Sparse signal recovery from $m = 100$ random measurements of a length $n = 256$ signal. The probability of successful recovery depends on the sparsity level k. The *dashed curves* represent a reweighted ℓ_1 algorithm that outperforms the traditional unweighted ℓ_1 approach (*solid curve*). (**a**) Performance after 4 reweighting iterations as a function of ϵ. (**b**) Performance with fixed $\epsilon = 0.1$ as a function of the number of reweighting iterations

fairly robust with respect to the choice of ϵ, with a suitable rule being about 10% of the standard deviation of the nonzero signal coefficients.

Figure 4(b) shows the performance, with a fixed value of $\epsilon = 0.1$, of the reweighting algorithm for various numbers of reweighted iterations. We see that much of the benefit comes from the first few reweighting iterations, and so the added computational cost for improved signal recovery is quite moderate.

3.2 Sparse and Compressible Signal Recovery with Adaptive Choice of ϵ

We would like to confirm the benefits of reweighted ℓ_1 minimization for compressible signal recovery and consider the situation when the parameter ϵ is not provided in advance and must be estimated during reconstruction. We propose an experiment in which each trial is designed as follows. We sample a signal of length $n = 256$ from one of three types of distribution: (1) k-sparse with i.i.d. Gaussian entries, (2) k-sparse with i.i.d. symmetric Bernoulli ± 1 entries, or (3) compressible, constructed by randomly permuting the sequence $\{i^{-1/p}\}_{i=1}^n$ for a fixed p, applying random sign flips, and normalizing so that $\|x_0\|_{\ell_\infty} = 1$; that is, we construct the ordered sequence $\{i^{-1/p}\}_{i=1}^n$, multiply each entry by a random sign and then apply a random permutation to the signed sequence. The output is x_0. We set $m = 128$ and sample a random $m \times n$ matrix Φ with i.i.d. Gaussian entries. To recover the signal, we again solve a reweighted ℓ_1 minimization with equality constraints $y = \Phi x_0 = \Phi x$. In this case, however, we adapt ϵ at each iteration as a function of the current guess $x^{(\ell)}$; Step 3 of the algorithm is modified as follows:

3. Let $(|x|_{(i)})$ denote a reordering of $(|x_i|)$ in decreasing order of magnitude. Set

$$\epsilon = \max\left\{|x^{(\ell)}|_{(i_0)}, 10^{-3}\right\},$$

where $i_0 = m/[4\log(n/m)]$. Define $w^{(\ell+1)}$ as in (7).

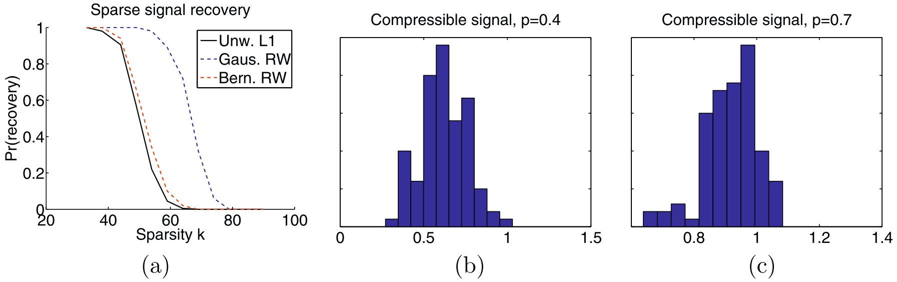


Fig. 5 (**a**) Improvements in sparse signal recovery from reweighted ℓ_1 minimization when compared to unweighted ℓ_1 minimization (*solid black curve*). The *dashed blue curve* corresponds to sparse signals with Gaussian coefficients; the *dashed red curve* corresponds to sparse signals with Bernoulli coefficients. (**b**), (**c**) Improvements in compressible signal recovery from reweighted ℓ_1 minimization when compared to unweighted ℓ_1 minimization; signal coefficients decay as $i^{-1/p}$ with (**b**) $p = 0.4$ and (**c**) $p = 0.7$. Histograms indicate the ℓ_2 reconstruction error improvements afforded by the reweighted algorithm

Our motivation for choosing this value for ϵ is based on the anticipated accuracy of ℓ_1 minimization for arbitrary signal recovery. In general, the reconstruction quality afforded by ℓ_1 minimization is comparable (approximately) to the best i_0-term approximation to x_0, and so we expect approximately this many signal components to be approximately correct. Put differently, we cannot really expect to recover the components of the signal whose absolute value is below $|x^{(\ell)}|_{(i_0)}$. This motivates a choice of ϵ on the order of $|x^{(\ell)}|_{(i_0)}$. Suppose, however, that the recovered signal is sparser than this and has fewer than i_0 nonzero terms. Then ϵ would be zero and some of the weights would be infinite. To prevent this from occurring, we put a lower cap on ϵ and thus an upper limit on the value of the weights. The choice $\epsilon \geq 10^{-3}$ may appear a little arbitrary but works well in practice.

We run 100 trials of the above experiment for each signal type. The results for the k-sparse experiments are shown in Fig. 5(a). The solid black line indicates the performance of unweighted ℓ_1 recovery (success is declared when $\|x_0 - x\|_{\ell_\infty} \leq 10^{-3}$). This curve is the same for both the Gaussian and Bernoulli coefficients, as the success or failure of unweighted ℓ_1 minimization depends only on the support and sign pattern of the original sparse signal. The dashed curves indicate the performance of reweighted ℓ_1 minimization for Gaussian coefficients (blue curve) and Bernoulli coefficients (red curve) with $\ell_{\max} = 4$. We see a substantial improvement for recovering sparse signals with Gaussian coefficients, yet we see only very slight improvement for recovering sparse signals with Bernoulli coefficients. This discrepancy likely occurs because the decay in the sparse Gaussian coefficients allows large coefficients to be easily identified and significantly downweighted early in the reweighting algorithm. With Bernoulli coefficients there is no such "low-hanging fruit".

The results for compressible signals are shown in Figs. 5(b), (c). Each plot represents a histogram, over 100 trials, of the ℓ_2 reconstruction error improvement afforded by reweighting, namely, $\|x_0 - x^{(4)}\|_{\ell_2} / \|x_0 - x^{(0)}\|_{\ell_2}$. We see the greatest improvements for smaller p corresponding to sparser signals, with reductions in ℓ_2 reconstruction error up to 50% or more. As $p \to 1$, the improvements diminish.

3.3 Recovery from Noisy Measurements

Reweighting can be applied to a noise-aware version of ℓ_1 minimization, further improving the recovery of signals from noisy data. We observe $y = \Phi x_0 + z$, where z is a noise term which is either stochastic or deterministic. To recover x_0, we adapt quadratically-constrained ℓ_1 minimization [7, 27], and modify step 2 of the reweighted ℓ_1 algorithm with equality constraints (see Sect. 2.2) as

$$x^{(\ell)} = \arg\min \|W^{(\ell)} x\|_{\ell_1} \quad \text{subject to} \quad \|y - \Phi x\|_{\ell_2} \le \delta. \tag{10}$$

The parameter δ is adjusted so that the true vector x_0 be feasible (resp. feasible with high probability) for (10) in the case where z is deterministic (resp. stochastic).

To demonstrate how this proposal improves on plain ℓ_1 minimization, we sample a vector of length $n = 256$ from one of three types of distribution: (1) k-sparse with $k = 38$ and i.i.d. Gaussian entries, (2) k-sparse with $k = 38$ and i.i.d. symmetric Bernoulli ± 1 entries, or (3) compressible, constructed by randomly permuting the sequence $\{i^{-1/p}\}_{i=1}^n$ for a fixed p, applying random sign flips, and normalizing so that $\|x_0\|_{\ell_\infty} = 1$. The matrix Φ is 128×256 with i.i.d. Gaussian entries whose columns are subsequently normalized, and the noise vector z is drawn from an i.i.d. Gaussian zero-mean distribution properly rescaled so that $\|z\|_{\ell_2} = \beta \|\Phi x\|_{\ell_2}$ with $\beta = 0.2$; i.e., $z = \sigma z_0$ where z_0 is standard white noise and $\sigma = \beta \|\Phi x\|_{\ell_2} / \|z_0\|_{\ell_2}$. The parameter δ is set to $\delta^2 = \sigma^2(m + 2\sqrt{2m})$ as this provides a likely upper bound on $\|z\|_{\ell_2}$. In details, the probability that $\|z\|_{\ell_2}^2$ exceeds δ^2 is the probability that a chi square with m degrees of freedom exceeds its mean by at least two standard deviations. This quantity is about 2.5% when m is not too small thanks to the approximation given by the central limit theorem. We set ϵ to be the empirical maximum value of $\|\Phi^* \xi\|_{\ell_\infty}$ over several realizations of a random vector $\xi \sim \mathcal{N}(0, \sigma^2 I_m)$; for the values $m = 128$ and $\beta = 0.2$ we have $\epsilon \approx 1$. (This gives a rough estimate for the noise amplitude in the signal domain, and hence, a baseline above which significant signal components could be identified.)

We run 100 trials for each signal type. Figure 6 shows histograms of the ℓ_2 reconstruction error improvement afforded by 9 iterations, i.e., each histogram documents $\|x_0 - x^{(9)}\|_{\ell_2} / \|x_0 - x^{(0)}\|_{\ell_2}$ over 100 trials. We see in these experiments that the reweighted quadratically-constrained ℓ_1 minimization typically offers improvements $\|x_0 - x^{(9)}\|_{\ell_2} / \|x_0 - x^{(0)}\|_{\ell_2}$ in the range $0.5 - 1$ in many examples. The results for sparse Gaussian spikes are slightly better than for sparse Bernoulli spikes, though both are generally favorable. Similar behavior holds for compressible signals, and we have observed that smaller values of p (sparser signals) allow the most improvement.

3.4 Statistical Estimation

Reweighting also enhances statistical estimation as well. Suppose we observe $y = \Phi x_0 + z$, where Φ is $m \times n$ with $m \le n$, and z is a noise vector $z \sim \mathcal{N}(0, \sigma^2 I_m)$ drawn from an i.i.d. Gaussian zero-mean distribution, say. To estimate x_0, we adapt the Dantzig selector [50] and modify step 2 of the reweighted ℓ_1 algorithm as

$$x^{(\ell)} = \arg\min \|W^{(\ell)} x\|_{\ell_1} \quad \text{subject to} \quad \|\Phi^*(y - \Phi x)\|_{\ell_\infty} \le \delta. \tag{11}$$

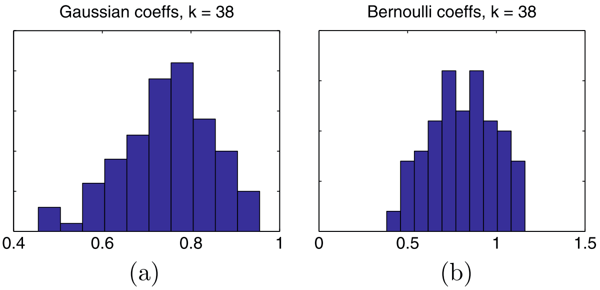


Fig. 6 Sparse and compressible signal reconstruction from noisy measurements. Histograms indicate the ℓ_2 reconstruction error improvements afforded by the reweighted quadratically-constrained ℓ_1 minimization for various signal types

Again δ is a parameter making sure that the true unknown vector is feasible with high probability.

To judge this proposal, we consider a sequence of experiments in which x_0 is of length $n = 256$ with $k = 8$ nonzero entries in random positions. The nonzero entries of x_0 have i.i.d. entries according to the model $x_i = s_i(1 + |a_i|)$ where the sign $s_i = \pm 1$ with probability $1/2$ and $a_i \sim \mathcal{N}(0, 1)$. The matrix Φ is 72×256 with i.i.d. Gaussian entries whose columns are subsequently normalized just as before. The noise vector (z_i) has i.i.d. $\mathcal{N}(0, \sigma^2)$ components with $\sigma = 1/3\sqrt{k/m} \approx 0.11$. The parameter δ is set to be the empirical maximum value of $\|\Phi^* z\|_{\ell_\infty}$ over several realizations of a random vector $z \sim \mathcal{N}(0, \sigma^2 I_m)$. We set $\epsilon = 0.1$.

After each iteration of the reweighted Dantzig selector, we also refine our estimate $x^{(\ell)}$ using the Gauss–Dantzig technique to correct for a systematic bias [50]. Let $I = \{i : |x_i^{(\ell)}| > \alpha \cdot \sigma\}$ with $\alpha = 1/4$. Then one substitutes $x^{(\ell)}$ with the least squares estimate which solves

$$\min_{x \in \mathbb{R}^n} \|y - \Phi x\|_{\ell_2} \quad \text{subject to} \quad x_i = 0,\ i \notin I;$$

that is, by regressing y onto the subset of columns indexed by I.

We first report on one trial with $\ell_{\max} = 4$. Figure 7(a) shows the original signal x_0 along with the recovery $x^{(0)}$ using the first (unweighted) Dantzig selector iteration; the error is $\|x_0 - x^{(0)}\|_{\ell_2} = 1.46$. Figure 7(b) shows the Dantzig selector recovery after 4 iterations; the error has decreased to $\|x_0 - x^{(4)}\|_{\ell_2} = 1.25$. Figure 7(c) shows the Gauss–Dantzig estimate $x^{(0)}$ obtained from the first (unweighted) Dantzig selector iteration; this decreases the error to $\|x_0 - x^{(0)}\|_{\ell_2} = 0.57$. The estimator correctly includes all 8 positions at which x_0 is nonzero, but also incorrectly includes 4 positions at which x_0 should be zero. In Fig. 7(d) we see, however, that all of these mistakes are rectified in the Gauss–Dantzig estimate $x^{(4)}$ obtained from the reweighted Dantzig selector; the total error also decreases to $\|x_0 - x^{(4)}\|_{\ell_2} = 0.29$.

We repeat the above experiment across 5000 trials. Figure 8 shows a histogram of the ratio ρ^2 between the squared error loss of some estimate x and the ideal squared

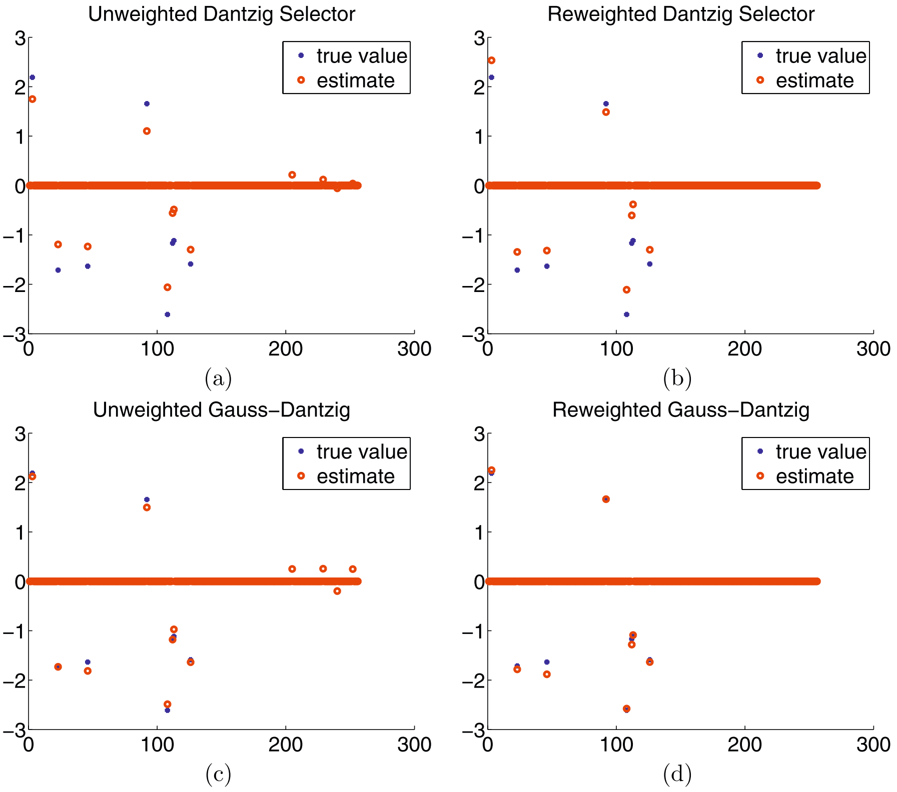


Fig. 7 Reweighting the Dantzig selector. *Blue asterisks* indicate the original signal x_0; *red circles* indicate the recovered estimate. (**a**) Unweighted Dantzig selector. (**b**) Reweighted Dantzig selector. (**c**) Unweighted Gauss–Dantzig estimate. (**d**) Reweighted Gauss–Dantzig estimate

error

$$\rho^2 := \frac{\sum_{i=1}^{n}(x_i - x_{0,i})^2}{\sum_{i=1}^{n}\min(x_{0,i}^2, \sigma^2)}$$

for both the unweighted and reweighted Gauss–Dantzig estimators. (The results are also summarized in Table 1.) For an interpretation of the denominator, the ideal squared error $\sum \min(x_{0,i}^2, \sigma^2)$ is roughly the mean-squared error one could achieve if one had available an oracle supplying perfect information about which coordinates of x_0 are nonzero, and which are actually worth estimating. We see again a significant reduction in reconstruction error; the median value of ρ^2 decreases from 2.43 to 1.21. As pointed out, a primary reason for this improvement comes from a more accurate identification of significant coefficients: on average the unweighted Gauss–Dantzig estimator includes 3.2 “false positives,” while the reweighted Gauss–Dantzig estimator includes only 0.5. Both algorithms correctly include all 8 nonzero positions in a large majority of trials.

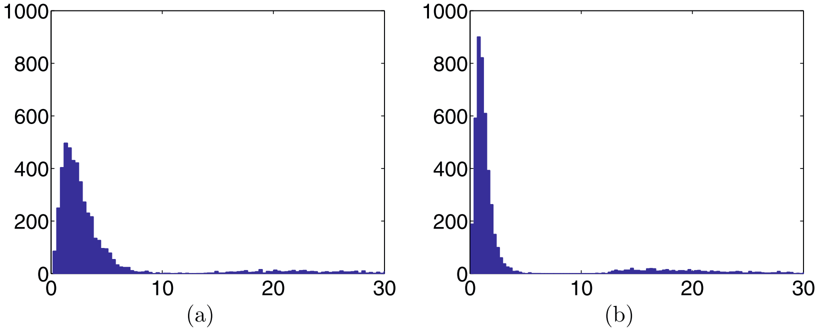


Fig. 8 Histogram of the ratio ρ^2 between the squared error loss and the ideal squared error for (**a**) unweighted Gauss–Dantzig estimator and (**b**) reweighted Gauss–Dantzig estimator. Approximately 5% of the tail of each histogram has been truncated for display; across 5000 trials the maximum value observed was $\rho^2 \approx 165$

Table 1 Model selection results for unweighted and reweighted versions of the Gauss–Dantzig estimator. In each of 5000 trials the true sparse model contains $k = 8$ nonzero entries

	Unweighted Gauss–Dantzig	Reweighted Gauss–Dantzig
Median error ratio ρ^2	2.43	1.21
Mean error ratio ρ^2	6.12	5.63
Avg. false positives	3.25	0.50
Avg. correct detections	7.86	7.80

3.5 Error Correction

Suppose we wish to transmit a real-valued signal $x_0 \in \mathbb{R}^n$, a block of n pieces of information, to a remote receiver. The vector x_0 is arbitrary and in particular, nonsparse. The difficulty is that errors occur upon transmission so that a fraction of the transmitted codeword may be corrupted in a completely arbitrary and unknown fashion. In this setup, the authors in [31] showed that one could transmit n pieces of information reliably by encoding the information as Φx_0 where $\Phi \in \mathbb{R}^{m \times n}$, $m \geq n$, is a suitable coding matrix, and by solving

$$\min_{x \in \mathbb{R}^n} \|y - \Phi x\|_{\ell_1} \tag{12}$$

upon receiving the corrupted codeword $y = \Phi x_0 + e$; here, e is the unknown but sparse corruption pattern. The conclusion of [31] is then that the solution to this program recovers x_0 exactly provided that the fraction of errors is not too large. Continuing on our theme, one can also enhance the performance of this error-correction strategy, further increasing the number of corrupted entries that can be overcome.

Select a vector of length $n = 128$ with elements drawn from a zero-mean unit-variance Gaussian distribution, and sample an $m \times n$ coding matrix Φ with i.i.d. Gaussian entries yielding the codeword Φx. For this experiment, $m = 4n = 512$, and k random entries of the codeword are corrupted with a sign flip. For the recovery, we simply use a reweighted version of (12). Our algorithm is as follows:

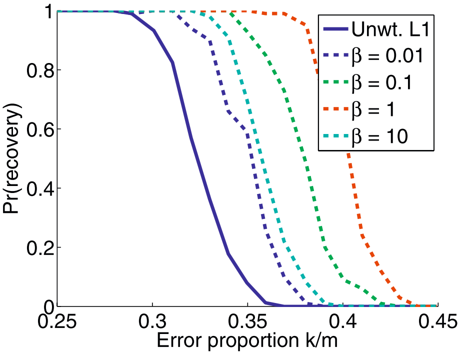


Fig. 9 Unweighted (*solid curve*) and reweighted (*dashed curve*) ℓ_1 signal recovery from corrupted measurements $y = \Phi x_0 + e$. The signal x_0 has length $n = 128$, the codeword y has size $m = 4n = 512$, and the number of corrupted entries $\|e\|_{\ell_0} = k$

1. Set $\ell = 0$ and $w_i^{(0)} = 1$ for $i = 1, 2, \dots, m$.
2. Solve the weighted ℓ_1 minimization problem

$$x^{(\ell)} = \arg\min \|W^{(\ell)}(y - \Phi x)\|_{\ell_1}. \tag{13}$$

3. Update the weights; let $r^{(\ell)} = y - \Phi x^{(\ell)}$ and for each $i = 1, \dots, m$, define

$$w_i^{(\ell+1)} = \frac{1}{|r_i^{(\ell)}| + \epsilon}. \tag{14}$$

4. Terminate on convergence or when ℓ attains a specified maximum number of iterations $\ell_{\max}$. Otherwise, increment ℓ and go to Step 2.

We set ϵ to be some factor β times the standard deviation of the corrupted codeword y. We run 100 trials for several values of β and of the size k of the corruption pattern.

Figure 9 shows the probability of perfect signal recovery (declared when $\|x_0 - x\|_{\ell_\infty} \le 10^{-3}$) for both the unweighted ℓ_1 decoding algorithm and the reweighted versions for various values of β (with $\ell_{\max} = 4$). Across a wide range of values β (and hence ϵ), we see that reweighting increases the number of corrupted entries (as a percentage of the codeword size m) that can be overcome, from approximately 28% to 35%.

3.6 Total Variation Minimization for Sparse Image Gradients

In a different direction, reweighting can also boost the performance of total-variation (TV) minimization for recovering images with sparse gradients. Recall the total-variation norm of a 2-dimensional array $(x_{i,j})$, $1 \le i, j \le n$, defined as the ℓ_1 norm of the magnitudes of the discrete gradient,

$$\|x\|_{\mathrm{TV}} = \sum_{1\le i,j\le n} \|(Dx)_{i,j}\|,$$

where $(Dx)_{i,j}$ is the 2-dimensional vector of forward differences $(Dx)_{i,j} = (x_{i+1,j} - x_{i,j}, x_{i,j+1} - x_{i,j})$ when $1 \le i, j \le n - 1$ while at the boundary, $(Dx)_{n,j} = (0, x_{n,j+1} - x_{n,j})$ for $1 \le j \le n - 1$, $(Dx)_{i,n} = (x_{i+1,n} - x_{i,n}, 0)$ for $1 \le i \le n - 1$, and $(Dx)_{n,n} = (0,0)$. Because many natural images have a sparse or nearly sparse gradient, it makes sense to search for the reconstruction with minimal TV norm, i.e.,

$$\min \|x\|_{\mathrm{TV}} \quad \text{subject to} \quad y = \Phi x; \tag{15}$$

see [9, 10], for example. It turns out that this problem can be recast as a second-order cone program [51], and thus solved efficiently. The insight here is that the total variation of x is equal to the optimal value of the optimization problem

$$\min \sum_{1 \le i,j \le n} u_{i,j} \quad \text{subject to} \quad \|(Dx)_{i,j}\| \le u_{i,j},$$

with optimization variables $\{u_{i,j}\}_{1 \le i,j \le n}$. When searching for an array x minimizing the total variation under linear equality constraints, we introduce the slack variables $u_{i,j}$ and minimize $\sum_{i,j} u_{i,j}$ under the constraints $\|(Dx)_{i,j}\| \le u_{i,j}$ and the linear equalities for x; the constraints $\|(Dx)_{i,j}\| \le u_{i,j}$ are second-order cone constraints.

We adapt (15) by minimizing a sequence of weighted TV norms as follows:

1. Set $\ell = 0$ and $w_{i,j}^{(0)} = 1$, $1 \le i, j \le n$.
2. Solve the weighted TV minimization problem

$$x^{(\ell)} = \arg\min \sum_{1 \le i,j \le n} w_{i,j}^{(\ell)} \|(Dx)_{i,j}\| \quad \text{subject to} \quad y = \Phi x.$$

3. Update the weights; for each (i, j), $1 \le i, j \le n$,

$$w_{i,j}^{(\ell+1)} = \frac{1}{\|(Dx^{(\ell)})_{i,j}\| + \epsilon}. \tag{16}$$

4. Terminate on convergence or when ℓ attains a specified maximum number of iterations $\ell_{\max}$. Otherwise, increment ℓ and go to Step 2.

Naturally, this iterative algorithm corresponds to minimizing a sequence of linearizations of the log-sum function $\sum_{1 \le i,j \le n} \log(\|(Dx)_{i,j}\| + \epsilon)$ around the previous signal estimate.

To show how this can enhance the performance of the recovery, consider the following experiment. Our test image is the Shepp-Logan phantom of size $n = 256 \times 256$ (see Fig. 10(a)). The pixels take values between 0 and 1, and the image has a nonzero gradient at 2184 pixels. We measure y by sampling the discrete Fourier transform of the phantom along 10 pseudo-radial lines (see Fig. 10(b)). That is, $y = \Phi x_0$, where Φ represents a subset of the Fourier coefficients of x_0. In total, we take $m = 2521$ real-valued measurements.

Figure 10(c) shows the result of the classical TV minimization, which gives a relative error equal to $\|x_0 - x^{(0)}\|_{\ell_2} / \|x_0\|_{\ell_2} \approx 0.43$. As shown in Fig. 10(d), however, we see a substantial improvement after 6 iterations of reweighted TV minimization (we used 0.1 for the value of ϵ). The recovery is near-perfect, with a relative error obeying $\|x_0 - x^{(6)}\|_{\ell_2} / \|x_0\|_{\ell_2} \approx 2 \times 10^{-3}$.

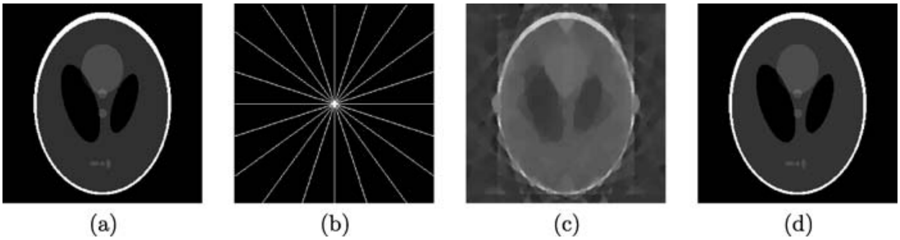


Fig. 10 Image recovery from reweighted TV minimization. (**a**) Original 256 × 256 phantom image. (**b**) Fourier-domain sampling pattern. (**c**) Minimum-TV reconstruction; total variation = 1336. (**d**) Reweighted TV reconstruction; total variation (unweighted) = 1464

For point of comparison it takes approximately 17 radial lines ($m = 4257$ real-valued measurements) to perfectly recover the phantom using unweighted TV minimization. Hence, with respect to the sparsity of the image gradient, we have reduced the requisite oversampling factor significantly, from $\frac{4257}{2184} \approx 1.95$ down to $\frac{2521}{2184} \approx 1.15$. It is worth noting that comparable reconstruction performance on the phantom image has also been recently achieved by directly minimizing a nonconvex ℓ_p norm, $p < 1$, of the image gradient [52]; we discuss this approach further in Sect. 5.1.

4 Reweighted ℓ_1 Analysis

In many problems, a signal may assume sparsity in a possibly overcomplete representation. To make things concrete, suppose we are given a dictionary Ψ of waveforms $(\psi_j)_{j \in J}$ (the columns of Ψ) which allows representing any signal as $x = \Psi\alpha$. The representation α is deemed sparse when the vector of coefficients α has comparably few significant terms. In some applications, it may be natural to choose Ψ as an orthonormal basis but in others, a sparse representation of the signal x may only become possible when Ψ is a *redundant* dictionary; that is, it has more columns than rows. A good example is provided by an audio signal which often is sparsely represented as a superposition of waveforms of the general shape $\sigma^{-1/2} g((t - t_0)/\sigma) e^{i\omega t}$, where t_0, ω, and σ are discrete shift, modulation and scale parameters.

In this setting, the common approach for sparsity-based recovery from linear measurements goes by the name of *Basis Pursuit* [8] and is of the form

$$\min \|\alpha\|_{\ell_1} \quad \text{subject to} \quad y = \Phi\Psi\alpha; \tag{17}$$

that is, we seek a sparse set of coefficients α that synthesize the signal $x = \Psi\alpha$. We call this *synthesis-based ℓ_1 recovery*. A far less common approach, however, seeks a signal x whose coefficients $\alpha = \Psi^* x$ (when x is analyzed in the dictionary Ψ) are sparse

$$\min \|\Psi^* x\|_{\ell_1} \quad \text{subject to} \quad y = \Phi x. \tag{18}$$

We call this *analysis-based ℓ_1 recovery*. When Ψ is an orthonormal basis, these two programs are identical, but in general they find *different* solutions. When Ψ is redundant, (18) involves fewer unknowns than (17) and may be computationally simpler to solve [53]. Moreover, in some cases the analysis-based reconstruction may in fact be superior, a phenomenon which is not very well understood; see [54] for some insights.

Both programs are amenable to reweighting but what is interesting is the combination of analysis-based ℓ_1 recovery and iterative reweighting which seems especially powerful. This section provides two typical examples. For completeness, the iterative reweighted ℓ_1-analysis algorithm is as follows:

1. Set $\ell = 0$ and $w_j^{(\ell)} = 1$, $j \in J$ (J indexes the dictionary).
2. Solve the weighted ℓ_1 minimization problem

$$x^{(\ell)} = \arg\min \|W^{(\ell)}\Psi^* x\|_{\ell_1} \quad \text{subject to} \quad y = \Phi x.$$

3. Put $\alpha^{(\ell)} = \Psi^* x^{(\ell)}$ and define

$$w_j^{(\ell+1)} = \frac{1}{|\alpha_j^{(\ell)}| + \epsilon}, \qquad j \in J.$$

4. Terminate on convergence or when ℓ attains a specified maximum number of iterations $\ell_{\max}$. Otherwise, increment ℓ and go to Step 2.

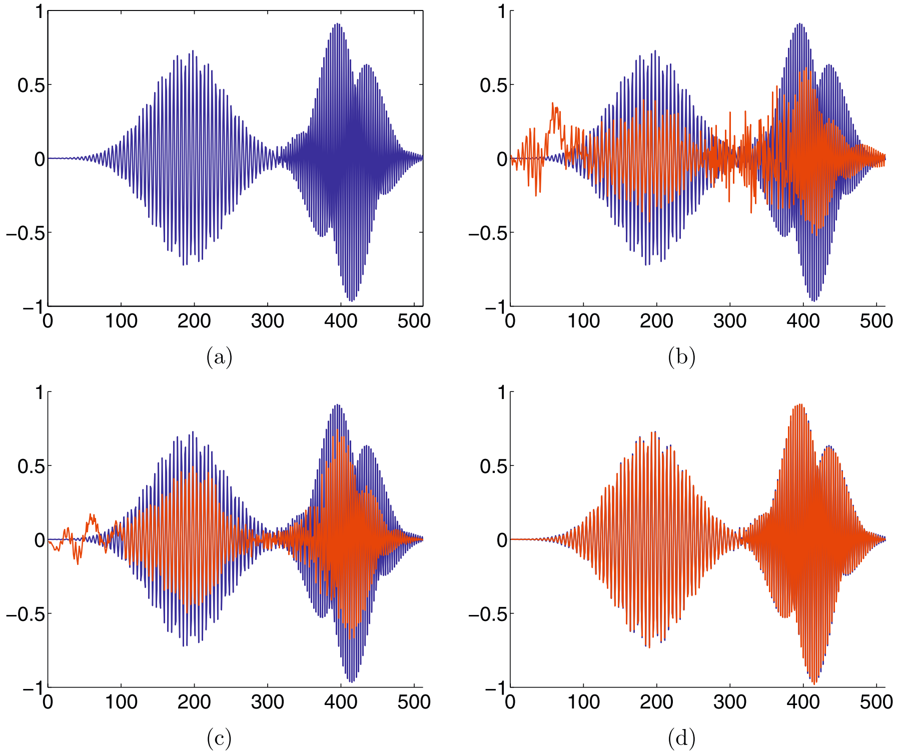
Unless specified otherwise, we will use this algorithm in the remainder of this section.

4.1 Incoherent Sampling of Radar Pulses

Our first example is motivated by our own research focused on advancing devices for analog-to-digital conversion of high-bandwidth signals. To cut a long story short, standard analog-to-digital converter (ADC) technology implements the usual quantized Shannon representation; that is, the signal is uniformly sampled at or above the Nyquist rate. The hardware brick wall is that conventional analog-to-digital conversion technology is currently limited to sample rates on the order of 1 GHz, and hardware implementations of high precision Shannon-based conversion at substantially higher rates seem out of sight for decades to come. This is where the theory of Compressive Sensing becomes relevant.

Whereas it may not be possible to digitize an analog signal at a very high rate, it may be quite possible to change its polarity at a high rate. The idea is then to multiply the signal by a pseudo-random sequence of plus and minus ones, integrate the product over time windows, and digitize the integral at the end of each time interval. This is a parallel architecture and one has several of these random multiplier-integrator pairs running in parallel using distinct or event nearly independent pseudo-random sign sequences.

To show the promise of this approach, we take x_0 to be a 1-D signal of length $n = 512$ which is a superposition of two modulated pulses (see Fig. 11(a)). From this signal, we collect $m = 30$ measurements using an $m \times n$ matrix Φ populated with i.i.d. Bernoulli ± 1 entries. This is an unreasonably small amount of data corresponding to an undersampling factor exceeding 17. For reconstruction we consider a time-frequency Gabor dictionary that consists of a variety of sine waves modulated by Gaussian windows, with different locations and scales. For each location and scale,



(a)

(b)

(c)

(d)

Fig. 11 (**a**) Original two-pulse signal (*blue*) and reconstructions (*red*) via (**b**) ℓ_1 synthesis, (**c**) ℓ_1 analysis, (**d**) reweighted ℓ_1 analysis

Table 2 Relative ℓ_2 reconstruction error as a function of reweighting iteration for two-pulse signal reconstruction

Iteration count ℓ	0	1	2	3	4	5	6	7
Error $\|x_0 - x^{(\ell)}\|_{\ell_2}/\|x\|_{\ell_2}$	0.460	0.101	0.038	0.024	0.022	0.022	0.022	0.022

the corresponding set of Gabor coefficients may be computed simply by multiplying the signal by the appropriate Gaussian window and then applying a zero-padded FFT. (The transpose of this operation simply requires an inverse FFT, truncation, and multiplication by the same windowing function.) Overall the dictionary is approximately 43 times overcomplete and does not contain the two pulses that comprise x_0.

Figure 11(b) shows the result of minimizing ℓ_1 synthesis (17) in this redundant dictionary. The reconstruction shows pronounced artifacts and $\|x_0 - x\|_{\ell_2}/\|x\|_{\ell_2} \approx 0.67$. These artifacts are somewhat reduced by analysis-based ℓ_1 recovery (18), as demonstrated in Fig. 11(c); here, we see $\|x_0 - x\|_{\ell_2}/\|x\|_{\ell_2} \approx 0.46$. However, reweighting the ℓ_1 analysis problem offers a very substantial improvement. Figure 11(d) shows the result after four iterations; $\|x_0 - x^{(4)}\|_{\ell_2}/\|x\|_{\ell_2}$ is now about

0.022. Further, Table 2 shows the relative reconstruction error $\|x_0 - x^{(\ell)}\|_{\ell_2}/\|x\|_{\ell_2}$ as a function of the iteration count ℓ. Massive gains are achieved after just 4 iterations.

4.2 Frequency Sampling of Biomedical Images

Compressive Sensing can help reduce the scan time in Magnetic Resonance Imaging (MRI) and offer sharper images of living tissues. This is especially important because time consuming MRI scans have traditionally limited the use of this sensing modality in important applications. Simply put, faster imaging here means novel applications. In MR, one collects information about an object by measuring its Fourier coefficients and faster acquisition here means fewer measurements.

We mimic an MR experiment by taking our unknown image x_0 to be the $n = 256 \times 256 = 65536$ pixel MR angiogram image shown in Fig. 12(a). We sample the image along 80 lines in the Fourier domain (see Fig. 12(b)), effectively taking $m = 18737$ real-valued measurements $y = \Phi x_0$. In plain terms, we undersample by a factor of about 3.

Figure 12(c) shows the minimum energy reconstruction which solves

$$\min \|x\|_{\ell_2} \quad \text{subject to} \quad y = \Phi x. \tag{19}$$

Figure 12(d) shows the result of TV minimization. The minimum ℓ_1-analysis (18) solution where Ψ is a three-scale redundant D4 wavelet dictionary that is 10 times overcomplete, is shown in Fig. 12(e). Figure 12(f) shows the result of reweighting the ℓ_1 analysis with $\ell_{\max} = 4$ and ϵ set to 100. For a point of comparison, the maximum wavelet coefficient has amplitude 4020, and approximately 108000 coefficients (out of 655360) have amplitude greater than 100.

We can reinterpret these results by comparing the reconstruction quality to the best k-term approximation to the image x_0 in a nonredundant wavelet dictionary. For example, an ℓ_2 reconstruction error equivalent to the ℓ_2 reconstruction of Fig. 12(c) would require keeping the $k = 1905 \approx m/9.84$ largest wavelet coefficients from the orthogonal wavelet transform of our test image. In this sense, the requisite oversampling factor can be thought of as being 9.84. Of course this can be substantially improved by encouraging sparsity, and the factor is reduced to 3.33 using TV minimization, to 3.25 using ℓ_1 analysis, and to 3.01 using reweighted ℓ_1 analysis.

We would like to be clear about what this means. Consider the image in Fig. 12(a) and its best k-term wavelet approximation with $k = 6225$; that is, the approximation obtained by computing all the D4 wavelet coefficients and retaining the k largest in the expansion of the object (and throwing out the others). Then we have shown that the image obtained by measuring $3k$ real-valued Fourier measurements and solving the iterative reweighted ℓ_1 analysis has just about the same accuracy. That is, the oversampling factor needed to obtain an image of the same quality as if one knew ahead of time the locations of the k most significant pieces of information and their value, is just 3.

5 Discussion

In summary, reweighted ℓ_1 minimization outperforms plain ℓ_1 minimization in a variety of setups. Therefore, this technique might be of interest to researchers in the

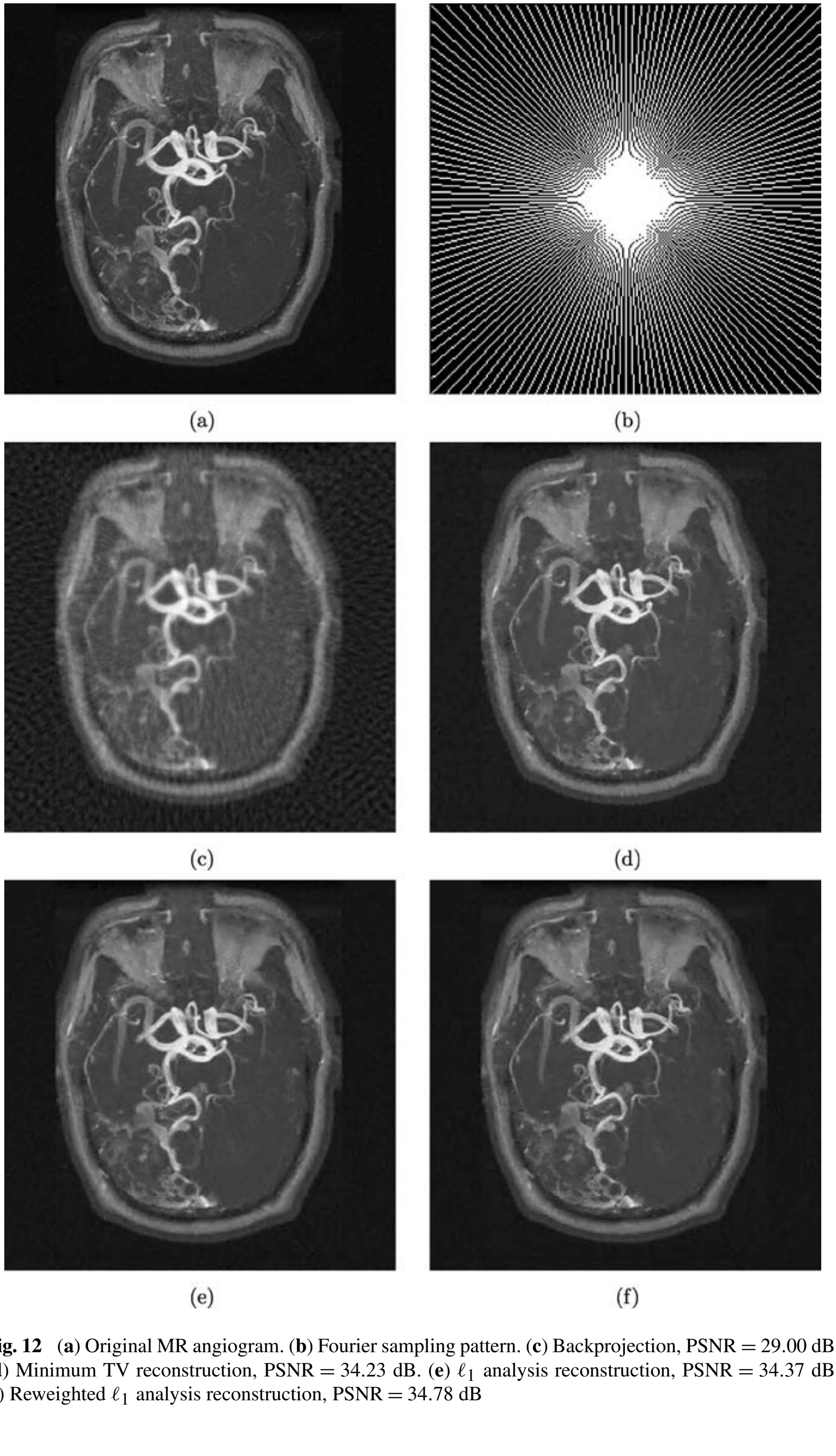


Fig. 12 (**a**) Original MR angiogram. (**b**) Fourier sampling pattern. (**c**) Backprojection, PSNR = 29.00 dB. (**d**) Minimum TV reconstruction, PSNR = 34.23 dB. (**e**) ℓ_1 analysis reconstruction, PSNR = 34.37 dB. (**f**) Reweighted ℓ_1 analysis reconstruction, PSNR = 34.78 dB

field of Compressive Sensing and/or statistical estimation as it might help to improve the quality of reconstructions and/or estimations. Further, this technique is easy to deploy as (1) it can be built on top of existing ℓ_1 solvers and (2) the number of iterations is typically very low so that the additional computational cost is not prohibitive. We conclude this paper by discussing related work and possible future directions.

5.1 Related Work

Whereas we have focused on modifying the ℓ_1 norm, a number of algorithms have been proposed that involve successively reweighting alternative penalty functions. In addition to IRLS (see Sect. 2.5), several such algorithms deserve mention.

Gorodnitsky and Rao [55] propose FOCUSS as an iterative method for finding sparse solutions to underdetermined systems. At each iteration, FOCUSS solves a reweighted ℓ_2 minimization with weights

$$w_i^{(\ell)} = \frac{1}{x_i^{(\ell-1)}} \tag{20}$$

for $i = 1, 2, \dots, n$. For nonzero signal coefficients, it is shown that each step of FOCUSS is equivalent to a step of the modified Newton's method for minimizing the function

$$\sum_{i:x_i \neq 0} \log |x_i|$$

subject to $y = \Phi x$. As the iterations proceed, it is suggested to identify those coefficients apparently converging to zero, remove them from subsequent iterations, and constrain them instead to be identically zero.

In a small series of experiments, we have observed that reweighted ℓ_1 minimization recovers sparse signals with lower error (or from fewer measurements) than the FOCUSS algorithm. We attribute this fact, for one, to the natural tendency of unweighted ℓ_1 minimization to encourage sparsity (while unweighted ℓ_2 minimization does not). We have also experimented with an ϵ-regularization to the reweighting function (20) that is analogous to (7). However we have found that this formulation fails to encourage strictly sparse solutions. It must be noted, however, that since the initial submission of our manuscript, we have learned of ongoing work in developing reweighting strategies for FOCUSS as well [56, 57]; in some situations the performance can be competitive with reweighted ℓ_1 minimization, and a more comprehensive study is warranted.

Harikumar and Bresler [58] propose an iterative algorithm that can be viewed as a generalization of FOCUSS. At each stage, the algorithm solves a convex optimization problem with a reweighted ℓ_2 cost function that encourages sparse solutions. The algorithm allows for different reweighting rules; for a given choice of reweighting rule, the algorithm converges to a local minimum of some concave objective function (analogous to the log-sum penalty function in (8)). These methods build upon ℓ_2 minimization rather than ℓ_1 minimization.

Delaney and Bresler [59] also propose a general algorithm for minimizing functionals having concave regularization penalties, again by solving a sequence of reweighted convex optimization problems (though not necessarily ℓ_2 problems) with weights that decrease as a function of the prior estimate. With the particular choice of a log-sum regularization penalty, the algorithm resembles the noise-aware reweighted ℓ_1 minimization discussed in Sect. 3.3.

Finally, in a slightly different vein, Chartrand [52] has recently proposed an iterative algorithm to minimize the concave objective $\|x\|_{\ell_p}$ with $p < 1$. (The algorithm alternates between gradient descent and projection onto the constraint set $y = \Phi x$.) While a global optimum cannot be guaranteed, experiments suggests that a local minimum may be found—when initializing with the minimum ℓ_2 solution—that is often quite sparse. This algorithm seems to outperform (P_1) in a number of instances and offers further support for the utility of nonconvex penalties in sparse signal recovery. There is also some theory showing that just like for ℓ_0 or ℓ_1 minimization, the ℓ_p minimizer for $p < 1$ enjoys nice theoretical guarantees, perhaps a bit better than those available for the ℓ_1 minimizer [52, 60] (the latter reference is posterior to the present paper and has been included in the revised version). We anticipate that similar theoretical guarantees would be available for the minimizer of the log-sum penalty as well. The problem with these results is that unlike ℓ_1 minimization, there are no known polynomial-time algorithms producing the ℓ_p minimizer or the log-sum minimizer. To reiterate, a major advantage of reweighted ℓ_1 minimization in this thrust is that (1) it can be implemented in a variety of settings (see Sects. 3 and 4) on top of existing and mature linear programming solvers and (2) it typically converges in very few steps. The log-sum penalty is also more ℓ_0-like and as we discuss in Sect. 2.4, additional concave penalty functions can be considered simply by adapting the reweighting rule.

5.2 Future Directions

In light of the promise of reweighted ℓ_1 minimization, it seems desirable to further investigate the properties of this algorithm.

- Under what conditions does the algorithm converge? That is, when do the successive iterates $x^{(\ell)}$ have a limit $x^{(\infty)}$?
- As shown in Sect. 2, when there is a sparse solution and the reweighted algorithm finds it, convergence may occur in just very few steps. It would be of interest to understand this phenomenon more precisely.
- What are smart and robust rules for selecting the parameter ϵ? That is, rules that would automatically adapt to the dynamic range and the sparsity of the object under study as to ensure reliable performance across a broad array of signals. Of interest are ways of updating ϵ as the algorithm progresses towards a solution. Of course, ϵ does not need to be uniform across all coordinates. As demonstrated in [61], some possible guidance in these directions may come from studying connections in robust statistics.
- We mentioned the use of other functionals and reweighting rules. How do they compare?
- Finally, any result quantifying the improvement of the reweighted algorithm for special classes of sparse or nearly sparse signals would be significant.

Acknowledgements E. C. was partially supported by a National Science Foundation grant CCF-515362, by the 2006 Waterman Award (NSF) and by a grant from DARPA. This work was performed while M. W. was an NSF Postdoctoral Fellow (NSF DMS-0603606) in the Department of Applied and Computational Mathematics at Caltech. S. B. was partially supported by NSF award 0529426, NASA award NNX07AEIIA, and AFOSR awards FA9550-06-1-0514 and FA9550-06-1-0312. We would like to thank Nathaniel Braun and Peter Stobbe for fruitful discussions about this project. Parts of this work were presented at the Fourth IEEE International Symposium on Biomedical Imaging (ISBI '07) held April 12–15, 2007 and at the Von Neumann Symposium on Sparse Representation and High-Dimensional Geometry held July 8–12, 2007. Related work was first developed as lecture notes for the course *EE364b: Convex Optimization II*, given at Stanford Winter quarter 2006-07 [62].

References

1. Boyd, S., Vandenberghe, L.: Convex Optimization. Cambridge University Press, Cambridge (2004)
2. Taylor, H.L., Banks, S.C., McCoy, J.F.: Deconvolution with the ℓ_1 norm. Geophysics **44**(1), 39–52 (1979)
3. Claerbout, J.F., Muir, F.: Robust modeling with erratic data. Geophysics **38**(5), 826–844 (1973)
4. Santosa, F., Symes, W.W.: Linear inversion of band-limited reflection seismograms. SIAM J. Sci. Stat. Comput. **7**(4), 1307–1330 (1986)
5. Donoho, D.L., Stark, P.B.: Uncertainty principles and signal recovery. SIAM J. Appl. Math. **49**(3), 906–931 (1989)
6. Donoho, D.L., Logan, B.F.: Signal recovery and the large sieve. SIAM J. Appl. Math. **52**(2), 577–591 (1992)
7. Tibshirani, R.: Regression shrinkage and selection via the lasso. J. R. Stat. Soc. B **58**(1), 267–288 (1996)
8. Chen, S., Donoho, D., Saunders, M.: Atomic decomposition by basis pursuit. SIAM J. Sci. Comput. **20**(1), 33–61 (1998)
9. Rudin, L.I., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. Physica D **60**(1–4), 259–268 (1992)
10. Blomgren, P., Chan, T.F.: Color TV: total variation methods for restoration of vector-valued images. IEEE Trans. Image Process. **7**, 304–309 (1998)
11. Vandenberghe, L., Boyd, S., El Gamal, A.: Optimal wire and transistor sizing for circuits with non-tree topology. In: Proceedings of the 1997 IEEE/ACM International Conference on Computer Aided Design, pp. 252–259 (1997)
12. Vandenberghe, L., Boyd, S., El Gamal, A.: Optimizing dominant time constant in RC circuits. IEEE Trans. Comput.-Aided Des. **2**(2), 110–125 (1998)
13. Hassibi, A., How, J., Boyd, S.: Low-authority controller design via convex optimization. AIAA J. Guid. Control Dyn. **22**(6), 862–872 (1999)
14. Dahleh, M., Diaz-Bobillo, I.: Control of Uncertain Systems: A Linear Programming Approach. Prentice-Hall, Englewood Cliffs (1995)
15. Lobo, M., Fazel, M., Boyd, S.: Portfolio optimization with linear and fixed transaction costs. Ann. Oper. Res. **152**(1), 341–365 (2006)
16. Ghosh, A., Boyd, S.: Growing well-connected graphs. In: Proceedings of the 45th IEEE Conference on Decision and Control, December 2006, pp. 6605–6611
17. Sun, J., Boyd, S., Xiao, L., Diaconis, P.: The fastest mixing Markov process on a graph and a connection to a maximum variance unfolding problem. SIAM Rev. **48**(4), 681–699 (2006)
18. Kim, S.-J., Koh, K., Boyd, S., Gorinevsky, D.: ℓ_1 trend filtering. SIAM Rev. (2008, to appear); available at www.stanford.edu/~boyd/l1_trend_filter.html
19. Donoho, D.L., Huo, X.: Uncertainty principles and ideal atomic decomposition. IEEE Trans. Inf. Theory **47**(7), 2845–2862 (2001)
20. Elad, M., Bruckstein, A.M.: A generalized uncertainty principle and sparse representation in pairs of bases. IEEE Trans. Inf. Theory **48**(9), 2558–2567 (2002)
21. Gribonval, R., Nielsen, M.: Sparse representations in unions of bases. IEEE Trans. Inf. Theory **49**(12), 3320–3325 (2003)
22. Tropp, J.A.: Just relax: convex programming methods for identifying sparse signals in noise. IEEE Trans. Inf. Theory **52**, 1030–1051 (2006)

23. Candès, E.J., Romberg, J., Tao, T.: Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. IEEE Trans. Inf. Theory **52**(2), 489–509 (2006)
24. Candès, E.J., Tao, T.: Near optimal signal recovery from random projections: Universal encoding strategies? IEEE Trans. Inf. Theory **52**(12), 5406–5425 (2006)
25. Donoho, D.: Compressed sensing. IEEE Trans. Inf. Theory **52**(4) (2006)
26. Donoho, D.L., Tanner, J.: Counting faces of randomly-projected polytopes when the projection radically lowers dimension. J. Am. Math. Soc. **22**, 1–53 (2009)
27. Candès, E.J., Romberg, J., Tao, T.: Stable signal recovery from incomplete and inaccurate measurements. Commun. Pure Appl. Math. **59**(8), 1207–1223 (2006)
28. Donoho, D., Tsaig, Y.: Extensions of compressed sensing. Signal Process. **86**(3), 533–548 (2006)
29. Takhar, D., Bansal, V., Wakin, M., Duarte, M., Baron, D., Kelly, K.F., Baraniuk, R.G.: A compressed sensing camera: New theory and an implementation using digital micromirrors. In: Proceedings of Comp. Imaging IV at SPIE Electronic Imaging, San Jose, California, January 2006
30. Lustig, M., Donoho, D., Pauly, J.M.: Sparse MRI: The application of compressed sensing for rapid MR imaging. Preprint (2007)
31. Candès, E.J., Tao, T.: Decoding by linear programming. IEEE Trans. Inf. Theory **51**(12), 4203–4215 (2005)
32. Candès, E.J., Randall, P.A.: Highly robust error correction by convex programming. Available on the ArXiV preprint server (cs/0612124) (2006)
33. Healy, D.L.: Analog-to-information (A-to-I). DARPA/MTO Broad Agency Announcement BAA 05-35 (July 2005)
34. Bajwa, W., Haupt, J., Sayeed, A., Nowak, R.: Compressive wireless sensing. In: Proceedings of Fifth Int. Conf. on Information Processing in Sensor Networks, pp. 134–142 (2006)
35. Baron, D., Wakin, M.B., Duarte, M.F., Sarvotham, S., Baraniuk, R.G.: Distributed compressed sensing. Preprint (2005)
36. Lange, K.: Optimization, Springer Texts in Statistics. Springer, New York (2004)
37. Fazel, M.: Matrix rank minimization with applications. Ph.D. thesis, Electrical Engineering Department, Stanford University (2002)
38. Lobo, M.S., Fazel, M., Boyd, S.: Portfolio optimization with linear and fixed transaction costs. Ann. Oper. Res. **152**(1), 341–365 (2007)
39. Fazel, M., Hindi, H., Boyd, S.: Log-det heuristic for matrix rank minimization with applications to Hankel and Euclidean distance matrices. In: Proceedings of Am. Control Conf., June 2003
40. Figueiredo, M.A.T., Nowak, R.D.: A bound optimization approach to wavelet-based image deconvolution. In: Proceedings of IEEE Int. Conf. on Image Processing (ICIP), vol. 2 (2005)
41. Figueiredo, M., Bioucas-Dias, J.M., Nowak, R.D.: Majorization–minimization algorithms for wavelet-based image restoration. IEEE Trans. Image Process. **16**(12), 2980–2991 (2007)
42. Wipf, D.P., Nagarajan, S.: A new view of automatic relevance determination. In: Proceedings on Neural Information Processing Systems (NIPS), vol. 20 (2008)
43. Zou, H.: The adaptive Lasso and its oracle properties. J. Am. Stat. Assoc. **101**(476), 1418–1429 (2006)
44. Zou, H., Li, R.: One-step sparse estimates in nonconcave penalized likelihood models. Ann. Stat. **36**(4), 1509–1533 (2008)
45. Schlossmacher, E.J.: An iterative technique for absolute deviations curve fitting. J. Am. Stat. Assoc. **68**(344), 857–859 (1973)
46. Holland, P., Welsch, R.: Robust regression using iteratively reweighted least-squares. Commun. Stat. Theor. Methods **A6**, 813–827 (1977)
47. Huber, P.J.: Robust Statistics. Wiley-Interscience, New York (1981)
48. Yarlagadda, R., Bednar, J.B., Watt, T.L.: Fast algorithms for ℓ_p deconvolution. IEEE Trans. Acoust. Speech Signal Process. **33**(1), 174–182 (1985)
49. Yin, W., Osher, S., Goldfarb, D., Darbon, J.: Bregman iterative algorithms for ℓ_1-minimization with applications to compressed sensing. SIAM J. Imaging Sci. **1**(1), 143–168 (2008)
50. Candès, E.J., Tao, T.: The Dantzig selector: Statistical estimation when p is much larger than n. Ann. Stat. **35**(6), 2313–2351 (2007)
51. Goldfarb, D., Yin, W.: Second-order cone programming methods for total variation-based image restoration. SIAM J. Sci. Comput. **27**(2), 622–645 (2005)
52. Chartrand, R.: Exact reconstruction of sparse signals via nonconvex minimization. Signal Process. Lett. **14**(10), 707–710 (2007)
53. Starck, J.-L., Elad, M., Donoho, D.L.: Redundant multiscale transforms and their application for morphological component analysis. Adv. Imaging Electron. Phys. **132**, 288–348 (2004)

54. Elad, M., Milanfar, P., Rubinstein, R.: Analysis versus synthesis in signal priors. Inverse Probl. **23**(3), 947–968 (2007)
55. Gorodnitsky, I.F., Rao, B.D.: Sparse signal reconstruction from limited data using FOCUSS: A re-weighted minimum norm algorithm. IEEE Trans. Signal Process. **45**(3), 600–616 (1997)
56. Chartrand, R., Yin, W.: Iteratively reweighted algorithms for compressive sensing. In: Proceedings of Int. Conf. on Acoustics, Speech, Signal Processing (ICASSP), pp. 3869–3872 (2008)
57. Wipf, D.P.: Personal communication (January 2008)
58. Harikumar, G., Bresler, Y.: A new algorithm for computing sparse solutions to linear inverse problems. In: Proceedings of Int. Conf. on Acoustics, Speech, Signal Processing (ICASSP). IEEE, New York (1996)
59. Delaney, A.H., Bresler, Y.: Globally convergent edge-preserving regularized reconstruction: An application to limited-angle tomography. IEEE Trans. Image Process. **7**(2), 204–221 (1998)
60. Saab, R., Chartrand, R., Yilmaz, O.: Stable sparse approximations via nonconvex optimization. In: 33rd International Conference on Acoustics, Speech, and Signal Processing (ICASSP), 2008
61. Black, M.J., Sapiro, G., Marimont, D.H., Heeger, D.: Robust anisotropic diffusion. IEEE Trans. Image Process. **7**(3), 421–432 (1998)
62. Boyd, S.: Lecture notes for EE364B: convex optimization II. Available at www.stanford.edu/class/ee364b/ (2007)