The Two Stage $l_1$ Approach to the Compressed Sensing Problem

Stéphane Chrétien

June 2, 2009

Abstract

This paper gives new results on the recovery of sparse signals using $l_1$-norm minimization. We introduce a two-stage $l_1$ algorithm equivalent to the first two iterations of the alternating $l_1$ relaxation introduced in [5] for an appropriate value of the Lagrange multiplier. The first step consists of the standard $\ell_1$ relaxation. The second step consists of optimizing the $\ell_1$ norm of a subvector whose components are indexed by the $\rho m$ largest components in the first stage. If $\rho$ is set to $\frac{1}{4}$, an intuitive choice motivated by the fact that $\frac{m}{4}$ is an empirical breakdown point for the plain $\ell_1$ exact recovery probability curve, Monte Carlo simulations show that the two-stage $\ell_1$ method outperforms the plain $\ell_1$ in practice.

1 Introduction

The Compressed sensing problem is currently the focus of an extensive research activity and can be stated as follows: Given a sparse vector $x^* \in \mathbb{R}^n$ and an observation matrix $A \in \mathbb{R}^{m \times n}$ with $m \ll n$, try to recover the vector $x$ from the small measurement vector $y = Ax^*$. Although the problem consists of solving an overdetermined system of linear equations, enough sparsity will allow to succeed as shown by the following lemma (where $\Sigma_s$ will denote the set of all $s$-sparse vectors, i.e. vectors whose components are all zero except for at most $s$ of them),

**Lemma 1.0.1** [3] If $A$ is any $m \times n$ matrix and $2s \leq m$, then the following properties are equivalent:

i. The decoder $\Delta_0(y)$ given by

$$\Delta_0(y) = \arg\min_{x \in \mathbb{R}^n} \|x\|_0 \text{ s.t. } Ax = y.$$  \hspace{1cm} (1.0.1)

k satisfies $\Delta_0(Ax) = x$, for all $x \in \Sigma_s$.

ii. For any set of indices $T$ with $|T| = 2k$, the matrix $A_T$ has rank $2s$ where $A_T$ stands for the submatrix of $A$ composed of the columns indexed by $T$ only.

1.1 The $l_1$ and the Reweighted $l_1$ relaxations

The main problem with decoder $\Delta_0$ is that the optimization problem (1.0.1) is in general NP-hard. For this reason, the now standard $l_1$ relaxation strategy is adopted, i.e. the decoder $\Delta_1(y)$ is obtained as

$$\Delta_1(y) = \arg\min_{x \in \mathbb{R}^n} \|x\|_1 \text{ s.t. } Ax = y.$$  \hspace{1cm} (1.1.1)

Now, solving (1.1.1) can be done in polynomial time and thus $\Delta_1(y)$ can be efficiently computed. The second problem is to give robust conditions under which exact recovery holds. One such condition was given by Candes Romberg and Tao [1] and is now known as the Uniform Uncertainty Principle (UUP) or as the Restricted Isometry Property (RIP).

One of the main remaining challenges is to reduce the number of observations $m$ needed to recover a given sparse signal $x$. One idea is the use of $l_p$, $p < 0 < 1$ decoders $\Delta_p(y)$. The main drawback of the approach using $l_p$, $p < 0 < 1$ norm minimization is that the resulting decoding scheme is again NP-Hard. Another idea is to use a reweighted $l_1$ approach as proposed in [8].

The main intuition behind this reweighted $\ell_1$ relaxation is the following. The greater the component $x_i$ becomes, the smaller weight it should receive since it can be considered that this component should not be set to zero.

The main drawback of the reweighted $l_1$ approach is that an unknown parameter is to be tuned whose order of magnitude is hard to know ahead of time.

*Mathematics Department, University of Franche Comté, 16 route de Gray, 25000 Besançon, France, Email: stephane.chretien@univ-fcomte.fr
The main remark about the alternating dual function at \( u \) sequence alternating \( l \) the role of an indicator function for the event that Algorithm 2

Another approach was proposed in [5] and uses Lagrange duality. Let us write down problem (1.0.1), to which \( \Delta_0 \) is the solution map, as the following equivalent problem

\[
\max_{z \in \{0,1\}^n, x \in \mathbb{R}^n} e^T z \quad \text{s.t.} \quad z_i x_i = 0, \quad i = 1, \ldots, n, \quad Ax = y
\]

where \( e \) denotes the vector of all ones. Here since the sum of the \( z_i \)'s is maximized, the variable \( z \) plays the role of an indicator function for the event that \( x_i = 0 \). This problem is clearly nonconvex due to the quadratic equality constraints \( z_i x_i = 0 \), \( i = 1, \ldots, n \). However, these constraints can be merged into the unique constraint \( \| D(z) x \|_1 = 0 \), leading to the following equivalent problem

\[
\max_{z \in \{0,1\}^n, x \in \mathbb{R}^n} e^T z \quad \text{s.t.} \quad \| D(z) x \|_1 = 0, \quad Ax = y. \tag{1.2.1}
\]

The Alternating \( l_1 \) algorithm consists of a suboptimal alternating minimization procedure to approximate the dual function at \( u \). The algorithm is as follows.

### Algorithm 1 Reweighted \( l_1 \) algorithm (Rew-\( l_1 \))

**Input** \( u > 0 \) and \( L \in \mathbb{N} \),

\[
z_u^{(0)} = e
\]

\[
x_u^{(0)} \in \min_{x \in \mathbb{R}^n, Ax = y} \| x \|_1
\]

\( l = 1 \)

while \( l \leq N \) do

\[
z_u^{(l)} = \frac{1}{|x^{(l)}| + u} \text{ componentwise}
\]

\[
x_u^{(l)} \in \text{argmax}_{x \in \mathbb{R}^n, Ax = y} \sum_{i=1}^n w^{(l-1)}_i |x_i|.
\]

\( l \leftarrow l + 1 \)

end while

**Output** \( z_u^{(L)} \) and \( x_u^{(L)} \).

### Algorithm 2 Alternating \( l_1 \) algorithm (Alt-\( l_1 \))

**Input** \( u > 0 \) and \( L \in \mathbb{N} \),

\[
z_u^{(0)} = e
\]

\[
x_u^{(0)} \in \max_{x \in \mathbb{R}^n, Ax = y} \mathcal{L}(x, z^{(0)} , u)
\]

\( l = 1 \)

while \( l \leq N \) do

\[
z_u^{(l)} \in \text{argmax}_{z \in \{0,1\}^n} \mathcal{L}(x_u^{(l)}, z, u)
\]

\[
x_u^{(l)} \in \text{argmax}_{x \in \mathbb{R}^n, Ax = y} \mathcal{L}(x, z_u^{(l)}, u)
\]

\( l \leftarrow l + 1 \)

end while

**Output** \( z_u^{(L)} \) and \( x_u^{(L)} \).

Notice that, similarly to the reweighted \( l_1 \) algorithm, the Alternating \( l_1 \) method also requires the tuning of an unknown parameter \( u \). However, the main motivation for this proposal is that this parameter \( u \) has a clear meaning: it is a dual variable which, in the case where the dual function \( \theta(u) \) is well approximated by the sequence \( \mathcal{L}(x^{(l)}, z^{(l)}, u) \), can be efficiently optimized without additional prior information, due to the convexity of the dual function.

### 2 The two stage \( l_1 \) method

The main remark about the alternating \( l_1 \) method is the following (see [5]): for a given dual variable \( u \), the alternating \( l_1 \) algorithm can be seen as a sequence \( (x_u^{(l)})_{l \in \mathbb{N}} \) of truncated \( l_1 \)-norm minimizers of the type

\[
x_u^l = \text{argmin}_{x \in \mathbb{R}^n} \| x T_u^l \|_1 \quad \text{s.t.} \quad Ax = y. \tag{2.0.2}
\]

where \( T_u^l \) is the set of indices for which \( |x_i^{l-1}| < \frac{1}{u} \). Therefore, the Alternating \( l_1 \) algorithm can be seen as an iterative thresholding scheme with threshold value equal to \( \frac{1}{u} \). Now assume for instance that a fraction \( \rho m \) of
the non zero components is well identified by the plain $l_1$ step with solution $x^{(t)}$. Then, the practitioner might ask if the appropriate value for $u$ is the one which imposes an $l_1$ penalty on the index set corresponding to the $n - \rho m$ smallest components of $x^{(t)}$. Moreover, the large scale simulation experiments which have been performed on the plain $l_1$ relaxation seemed to agree on the fact that the breakpoint point occurs near $\frac{m}{n}$. Thus, a practitioner could be tempted to wonder whether $\rho = \frac{1}{4}$ is a sensible value. Motivated by the previous practical considerations, the two stage $l_1$ algorithm is defined as follows (the parameter $u$ is now replaced by the parameter $\rho = \frac{1}{4}$).

Algorithm 3 Two stage $l_1$ algorithm (2Stage-$l_1$)

| Input: $\rho \in (0, \frac{1}{2})$ |
|-------------------|-------------------|
| Step 0: $x^{(0)} \in \arg\max_{x \in \mathbb{R}^n, A x = b} ||x||_1$ and $T =$ index set of the $\rho m$ largest components of $x^{(0)}$ |
| Step 1: $x^{(1)} \in \arg\max_{x \in \mathbb{R}^n, A x = b} ||x||_1$ |
| Output: $x^{(1)}_p$ |

Notice that we restrict $\rho$ to lie in $(0, \frac{1}{2})$. The reason should be obvious since, due to Lemma 3.0.1, even decoder $\Delta_0(y)$ is unable to identify more that $\frac{m}{2}$-sparse vectors. Another remark is that the procedure could be continued for more than 2 steps but simulation experiments of the Alternating $l_1$ method seem to confirm that in most cases two steps suffice to converge.

### 3 Main results

At Step 1 of the method, a subset $T$ is selected with cardinal $\rho m$ and optimization is then performed with objective function $||x_T^*||_1$. In this section, we will adopt the following notations: $S$ will denote the support of $x^*$, $T$ will denote the index set of the $\rho m$ largest components of $x^{(0)}$ as defined in the two-stage $l_1$ algorithm. $T^c_g$ will be an abbreviation of $(T^c)_g$, the ”good” subset of $T^c$ or, in mathematical terms, the subset of indices of $S$ which also belong to $T^c$. On the other hand, $T^c_h$ will denote the complement of $T^c_g$ in $T^c$.

Lemma 3.0.1 Assume the cardinal of $T^c_g$ is less than $\gamma/2$ and that $A$ satisfies RIP($\delta, \gamma s$). Let $h^{(1)} = x^{(1)} - x^*$. Then, there exists a positive number $C^*$ depending on $x^*$ such that $\|h_T^{(1)}\|_1 \leq C^*\|x_T^{(1)}\|_1$. Moreover, if $\|h_T^{(1)}\|_1 = 0$, then $\|x_T^{(1)}\|_1 = 0$.

Proof. Let $N(h_T)$ denote the optimal value of the problem

$$\min_{h_T \in \mathbb{R}^{n-x}} \|x_T^* + h_T\|_1$$

subject to

$$A_{T^c_g}(x_T^{(1)} + h_T^{(1)}) + A_{T^c_g}x_T^{(1)} = y - A_T(x_T^* + h_T).$$

Assume that $x_T^{(1)} = 0$. Then, $N(h_T)$ plays the role of a norm for $A_T$ although it does not satisfy the triangle inequality. In particular, $N(h_T)$ is nonnegative and $N(0) = 0$ implies that $h_T = 0$.

Nonnegativity and convexity are straightforward. Assume that $N(h_T) = 0$, i.e. the solution $\tilde{h}$ of (3.0.3) is null. This implies that $A_{T^c_g}x_T^{(1)} = y - A_T x_T^* = A_{T^c_g}x_T^{(1)}$, which implies that $x_T^{(1)} - x_T^*$ is in the kernel of $A_{T^c_g}$. Using the fact that $T^c_g$ has cardinal less that $\gamma s/2$ and the RIP($\delta, \gamma s$) assumption, we conclude that $x_T^{(1)} = x_T^*$.

In order to finish the proof of the lemma, it remains to recall that $N(h_T)$ is convex and that, by Theorem 1.1 in [7], $\|h_T^{(1)}\|_1$ (and thus $\|h_T^{(1)}\|_1$) is bounded from above by $C \inf_{U \subseteq \gamma s/2} \|x_U\|_1$ in order to obtain existence of a sufficiently small positive constant $C^*$ depending on $x^*$ such that $N(h_T) \geq C^*\|h_T\|_1$ for all $h_T$ in the ball $B(0, C\|x_T^*\|_1)$. The desired result then follows.

To prove that $N(h_T) = 0$ if $h_T = 0$ is a bit harder. Thus, assume that $h_T = 0$. Then, the solution $\tilde{h}$ of (3.0.3) is just the solution of

$$\min_{h_T \in \mathbb{R}^{n-x}} \|h_T\|_1 \quad \quad A_{T^c_h}h_T^{(1)} = y - A_T x_T^* - A_{T^c_g}x_T^{(1)}.$$ 

Now since $y = Ax^*$, we obtain that $y - A_T x_T^* - A_{T^c_g}x_T^{(1)} = A_{T^c_h}(x_T^{(1)} - x_T^*)$ and thus, the right hand side term is nothing but the image of a $\gamma s/2$-sparse vector. Now, recalling that we assumed RIP($\delta, \gamma s$), Theorem 1.1 in [7] implies that $\tilde{h}$ must be the sparsest solution of the system $A_{T^c_h}h_T^{(1)} = y - A_T x_T^* - A_{T^c_g}x_T^{(1)}$ from which we deduce that $\tilde{h}$ is $\gamma s/2$-sparse. Therefore the vector $(h_T^{(1)}, x_T^{(1)})$ is $\gamma s$ which solves $A_T x_T^* = y - A_T x_T^*$. On
the other hand, \( x^*_T \) also solves \( A_T x_T = y - A_T x^*_T \) and its support is included in the support of \( (h^{(1)}_{T_y}, x^{(1)}_{T_y}) \). Therefore, \( (h^{(1)}_{T_y}, x^{(1)}_{T_y}) - x^*_T \) is a \( \gamma/2 \) sparse vector which lies in the kernel of \( A \). Using again the fact that \( RIP(\delta, \gamma s) \) holds, we conclude that \( (h^{(1)}_{T_y}, x^{(1)}_{T_y}) - x^*_T = 0 \). Thus, \( h^{(1)}_{T_y} = 0 \) and \( x^{(1)}_{T_y} = x^*_T \).

Using this lemma, we deduce the following theorem.

**Theorem 3.0.2** Assume that \( RIP(\delta, \gamma s) \) holds and that an index set \( T_y \) of cardinal greater than or equal to \( (1 - \gamma/2)s \) has been recovered at Step 0 after thresholding, then \( x^{(1)} \) satisfies

\[
\| x^{(1)} - x^* \|_1 \leq C \| x^*_T \|_1.
\]

for some constant \( C \) depending on \( x^* \).

**Proof.** The vector \( x^{(1)} \) satisfies

\[
\| x^{(1)} \|_1 \leq \| x^*_T \|_1.
\]

Let us write \( h^{(1)} = x^{(1)} - x^* \). Using (3.0.4), a now standard decomposition gives

\[
\| x^*_T \|_1 - \| h^{(1)} \|_1 + \| h^{(1)} \|_1 - \| x^*_T \|_1 \leq \| x^*_T \|_1 + \| x^*_T \|_1.
\]

We thus obtain

\[
\| h^{(1)} \|_1 \leq \| h^{(1)} \|_1 + 2 \| x^*_T \|_1.
\]

However, since \( RIP(\delta, \gamma s) \) holds, \( NSP(C, \gamma/2) \) holds too, with \( C < 1 \). Therefore, we obtain that

\[
\| h^{(1)} \|_1 \leq C(\| h^{(1)} \|_1 + \| h^T \|_1).
\]

Combining (3.0.5) and (3.0.6), we obtain

\[
\| h^{(1)} \|_1 \leq \frac{C}{1 - C} \| h^T \|_1 + \frac{2}{1 - C} \| x^*_T \|_1.
\]

As a consequence, we obtain that

\[
\| h \|_1 \leq C(\| x^*_T \|_1 + C\| h^T \|_1) + \| h^T \|_1 + 2 \| x^*_T \|_1
\]

\[
+ C\| h^T \|_1 + \| h^T \|_1
\]

\[
= (1 + C + CC') \| h^T \|_1 + 2(1 + C) \| x^*_T \|_1.
\]

which, using Lemma 3.0.4, implies

\[
\| h \|_1 \leq ((1 + C + CC')C^* + 2(1 + C)) \| x^*_T \|_1.
\]

which is the desired bound. □

The following corollary is a straightforward consequence of the previous theorem.

**Corollary 3.0.3** Assume that the assumptions of Theorem 3.0.2 are satisfied. Then, exact reconstruction is obtained if \( x^*_T = 0 \), i.e. \( x^* \) is \( s \)-sparse.

## 4 Monte Carlo experiments

The following Monte Carlo experiments show that the performance of the two-stage \( l_1 \) algorithm which drops the penalty over the index set of the \( m/4 \) largest components of the solution of plain \( l_1 \) is almost as good as the performance of the reweighted \( l_1 \) with the best parameter which is usually unknown in practice. A Python program is available at \texttt{http://stephane.g.chretien.googlepages.com/alternatingl1} and can be used to perform these experiments and other involving the Alternating \( l_1 \) algorithm.

## References

[1] Candes, E., Romberg, J. and Tao T., Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information, IEEE Information Theory, 2006, 2, 52, pp. 489–509.

[2] Candes, E., Compressive sampling, 2006, 3, International Congress of Mathematics, pp. 1433–1452, EMS.
Figure 1: Rate of success over 300 Monte Carlo experiments in recovering the support of the signal vs. signal sparsity $k$ for $n = 128$, $m = 50$, $L = 4$, $u = 3$. $A$ and nonnull components of $x$ were drawn from the gaussian $\mathcal{N}(0, 1)$ distribution. The results for the two-stage $l_1$ method are represented by the "+ in a circle" sign.

[3] Cohen, A., Dahmen, W. and DeVore R., Compressed sensing and best $k$-term approximation,

[4] Lemaréchal, C. and Oustry, F., SDP relaxations in combinatorial optimization from a Lagrangian viewpoint, Advances in convex analysis and global optimization (Pythagorion, 2000), Nonconvex Optim. Appl. vol. 54, Kluwer Acad. Publ., pp. 119–134.

[5] Chrétien, S. An alternating $\ell_1$ relaxation for the compressed sensing problem, http://arxiv.org/abs/0809.0660

[6] Hiriart Urruty, J.-B. and Lemaréchal, C., Convex analysis and minimization algorithms II: Advanced theory and bundle methods, Springer-Verlag, 1993, 306, Grundlehren der Mathematischen Wissenschaften.

[7] Candès, E. The restricted isometry property and its implications for compressed sensing, Compte Rendus de l’Académie des Sciences, Paris, Serie I, 346 589-592.

[8] Candès, E., Wakin, M. and Boyd S., Enhancing Sparsity by Reweighted $l_1$ Minimization, Journal of Fourier Analysis and Applications, 2008, 14, pp. 877–905.