SPARSE APPROXIMATION VIA ITERATIVE THRESHOLDING

Kyle K. Herrity, Anna C. Gilbert, and Joel A. Tropp

University of Michigan
Department of Mathematics
Ann Arbor, MI 48109

ABSTRACT

The well-known *shrinkage* technique is still relevant for contemporary signal processing problems over redundant dictionaries. We present theoretical and empirical analyses for two iterative algorithms for sparse approximation that use shrinkage. The GENERAL IT algorithm amounts to a Landweber iteration with nonlinear shrinkage at each iteration step. The BLOCK IT algorithm arises in morphological components analysis. A sufficient condition for which General IT exactly recovers a sparse signal is presented, in which the cumulative coherence function naturally arises. This analysis extends previous results concerning the Orthogonal Matching Pursuit (OMP) and Basis Pursuit (BP) algorithms to IT algorithms.

1. INTRODUCTION

Sparse approximation problems have been studied for nearly a century, and they arise in many arenas, from compression and analysis of audio, image, and video signals, to machine learning, denoising, and regularization. In each of these applications, a target signal is approximated by a linear combination of elementary signals, drawn from a (fixed) large and linearly dependent collection of signals called a *dictionary*. The crucial element in sparse approximation is that we seek a good approximation using as few elementary signals as possible—a *sparse* approximation. Much recent attention has been devoted to proving algorithms compute optimal sparse approximations, despite the fact that the general sparse approximation problem is NP-hard for an arbitrary redundant dictionary. The geometry of the overcomplete dictionary under consideration plays an important role in the success of the algorithm.

In 1994, Donoho and Johnstone introduced a denoising technique known as *shrinkage*, which is optimal for estimating signals that are sparse with respect to an orthonormal basis and contaminated with Gaussian white noise. Although the assumption that the dictionary be unitary is crucial to the optimality of shrinkage, it has been employed quite succesfully in practice in the case where the dictionary is redundant [2].

We define two iterative thresholding algorithms (GENERAL IT and BLOCK IT) which incorporate coefficient shrinkage or thresholding at each iteration. GENERAL IT is a Landweber iteration with nonlinear shrinkage at each step and is motivated by the analysis of [1]. BLOCK IT is used when our redundant dictionary is a union of orthonormal bases (e.g., morphological components [3]). The BLOCK IT algorithm thresholds in each basis sequentially and, as such, is the more practical algorithm. Each substep of a full iteration involves a single unitary transform (possibly with a fast implementation) and we need only work with residual and coefficient vectors, which are equal in length to the original signal. GENERAL IT requires that we manipulate a larger coefficient vector and perform two matrix–vector multiplications much larger than the single unitary transforms. This algorithm does, however, take into account the interactions among the vectors in the dictionary.

We provide a sufficient condition for which guarantees that GENERAL IT recovers exactly sparse signals. This sufficient condition matches the sufficient geometric conditions for the Orthogonal Matching Pursuit (OMP) and Basis Pursuit (BP) algorithms. We also provide analysis of the fixed points of the BLOCK IT algorithm. In the following section we make rigorous the concepts that arise in sparse approximation problems and define two iterative thresholding algorithms. We then provide the theoretical analysis of these algorithms, followed by a discussion of the main result of the article: a sufficient condition guaranteeing the recovery of exactly sparse signals. The paper concludes with a study of the empirical performance of each algorithm.

2. PRELIMINARIES

We work in the complex inner-product space $\mathbb{C}^d$, which is called the *signal space*. The objective is to find an efficient representation of the signal by selecting atoms, or column vectors, from a dictionary. A *dictionary* $\mathscr{D}$ in $\mathbb{C}^d$ is a finite collection of unit-norm (column) vectors $\{\phi_\omega\}$, called atoms, that span the signal space. We write $\mathscr{D} = \{\phi_\omega | \omega \in \Omega\}$ and we form the $d \times N$ dictionary matrix Φ whose columns are the atoms of $\mathscr{D}$. Usually, the number of atoms, N, in the dictionary is much larger than d. We use the symbol * for the complex, conjugate transpose of vectors and matrices.

A fundamental metric associated with a dictionary is the

1-4244-0469-X/06/$20.00 ©2006 IEEE

coherence μ of the dictionary. It is an indicator of how correlated two atoms from the dictionary are to each other and is calculated by

$$\mu = \max_{j \neq k} |\langle \phi_{\omega_j}, \phi_{\omega_k} \rangle| = \max_{j \neq k} |\left(\Phi^* \Phi\right)_{jk}|$$

A less pessimistic measure of the correlation between atoms, and a generalization of the coherence is the *cumulative coherence*. The cumulative coherence function $\mu_1(m)$ is defined for positive integers m by

$$\mu_1(m) = \max_{|\Lambda| = m} \max_{\Psi} \sum_{\lambda \in \Lambda} |\langle \psi, \phi_\lambda \rangle|,$$

where the vector ψ ranges over the atoms indexed by $\Omega \setminus \Lambda$.

A *representation* of a signal s in Φ is a column vector (or coefficient vector) c such that $s = \Phi c$. Since $N > d$, the atoms of Φ are linearly dependent, and so the hope is we can find a sparse representation, i.e., a representation with few nonzero coefficients. We calculate the sparsity of the coefficient vector via the l_0 quasi-norm $\|\cdot\|_0$ as

$$\|c\|_0 = |\{\omega \in \Omega \mid c_\omega \neq 0\}|$$

2.1. Spare approximation problems

The SPARSE problem is to construct the best approximation of a signal with a linear combination of m atoms or fewer from the dictionary.

$$\min_{c \in \mathbb{C}^\Omega} \|s - \Phi c\|_2 \quad \text{subject to} \quad \|c\|_0 \leq m \qquad (\text{SPARSE})$$

This problem is primarily studied in applications where one has a generative model of the input signal or in resource-constrained settings where one has limited storage. The EXACT-SPARSE problem is to recover an exact superposition

$$s = \Phi \alpha = \sum_{k=1}^{m} \alpha_k \phi_k \qquad (\text{EXACT-SPARSE})$$

of m atoms from a redundant dictionary, where $\alpha \in \mathbb{C}^\Omega$.

Though natural signals rarely happen to be linear combinations of atoms, the EXACT-SPARSE problem may lend some insight into some of the more challenging sparse approximation problems, such as SPARSE.

2.2. Iterative thresholding (IT) algorithms

Let us first define the (nonlinear) hard and soft thresholding operators, H_θ and S_θ.

$$H_\theta(x) = \begin{cases} x & |x| \geq \frac{\theta}{2} \\ 0 & |x| < \frac{\theta}{2} \end{cases} \quad S_\theta(x) = \begin{cases} x + \frac{\theta}{2} & x \leq \frac{-\theta}{2} \\ 0 & |x| < \frac{\theta}{2} \\ x - \frac{\theta}{2} & x \geq \frac{\theta}{2} \end{cases}$$

If $x \in \mathbb{C}$, then we use a complex thresholding operator, which is defined as $H_\theta\left(re^{\mathrm{i}\omega}\right) = H_\theta(r)e^{\mathrm{i}\omega}$, and similarly for S_θ. Let us continue with a formal description of the algorithms.

Algorithm 1 (GENERAL IT)

1. *Initialize the coefficient vector* $c^0 = 0$*, the approximation vector* $a^0 = \Phi c^0 = 0$*, and the iteration counter* $t = 1$.
2. *Update the coefficient vector as*

$$c^t \longleftarrow H_\theta\left(c^{t-1} + \Phi^*\left(s - \Phi c^{t-1}\right)\right)$$

3. *Calculate the new approximation vector as* $a^t = \Phi c^t$.
4. *When the stopping criterion is met, output the vector of coefficients* c^T*, the approximation vector* a^T*, and the residual* $r^T = s - a^T$.

In the case where $\Phi = [\Phi_1, \Phi_2, \ldots, \Phi_J]$ is a union of J orthonormal dictionaries, a second algorithm, which we will refer to as BLOCK IT, has been proposed in [2, 3]. This algorithm is similar to GENERAL IT, except that we analyze the residual of the signal in the first basis, threshold the result, compute the new residual, and repeat in each of the other bases. We describe the algorithm for $J = 2$, but the definition may easily be extended for $J > 2$.

Algorithm 2 (BLOCK IT)

1. *Initialize the coefficient vector* $c = [c_1, c_2] = 0$*, and the residual* $r^0 = s$*, and the iteration counter* $t = 1$.
2. *Update the coefficient vector as*
   (1) $c_1^t = H_\theta\left(\Phi_1^* r^{t-1}\right)$
   (2) $r^{t-\frac{1}{2}} = s - \Phi_1 c_1^t$
   (3) $c_2^t = H_\theta\left(\Phi_2^* r^{t-\frac{1}{2}}\right)$
   (4) $r^t = s - \Phi_2 c_2^t$
3. *Calculate the new approximation vector as* $a_t = \Phi c_t$.
4. *When the stopping criterion is met, output the vector of coefficients* $c = [c_1, c_2]$.

3. MAIN RESULTS

We will develop a condition which guarantees that GENERAL IT recovers the m-term representation of exactly sparse signals—resembling the condition in [4]—when the coefficient vector α contains only ones and zeros. The results in [4] demonstrate that the critical problem in sparse approximation is to find the support set of the coefficients—not to find the values of the coefficients themselves. For this reason, we focus on this type of coefficient vector. Denote the set of atoms participating in s as Λ. From the dictionary synthesis matrix, we form the $d \times m$ matrix Φ_Λ whose columns are the atoms listed in Λ. We also define a second matrix Ψ_Λ whose columns are the $(N-m)$ atoms indexed by $\Omega \setminus \Lambda$.

Theorem 3.1. *Let $s = \Phi_\Lambda \mathbf{1}$, where $|\Lambda| = m$. Then* GENERAL IT *(hard) recovers the m-term representation of s up to any prescribed error tolerance if*

$$\mu_1(m-1) + \mu_1(m) < 1$$

Proof. Notice that after rearranging terms,

$$\Phi^* s = \left[\begin{array}{c} \Phi_\Lambda^* \Phi_\Lambda \mathbf{1} \\ \hline \Psi_\Lambda^* \Phi_\Lambda \mathbf{1} \end{array} \right].$$

The upper entries in the vector of coefficients capture the coherence of the vectors in the Λ among themselves and the lower entries capture the coherence of these vectors with the rest of the dictionary. We can bound the coherence of the vectors in Λ among themselves as

$$\left| \sum_{j \neq i} \langle \phi_{\omega_i}, \phi_{\omega_j} \rangle \right| \leq \sum_{j \neq i} |\langle \phi_{\omega_i}, \phi_{\omega_j} \rangle| \leq \mu_1(m-1)$$

for $\omega_i, \omega_j \in \Lambda$. Similary, we can bound the expression

$$\left| \sum_{\substack{\omega_k \in \Lambda \\ \omega_l \notin \Lambda}} \langle \phi_{\omega_k}, \phi_{\omega_l} \rangle \right| \leq \sum_{\substack{\omega_k \in \Lambda \\ \omega_l \notin \Lambda}} |\langle \phi_{\omega_k}, \phi_{\omega_l} \rangle| \leq \mu_1(m).$$

These two bounds provide upper and lower estimates for each entry in $\Phi^* s$. Each entry v_i in $\Phi_\Lambda^* \Phi_\Lambda \mathbf{1}$ satisfies

$$1 - \mu_1(m-1) \leq v_i \leq 1 + \mu_1(m-1)$$

and each entry v_i' in $\Psi_\Lambda^* \Phi_\Lambda \mathbf{1}$ satisfies

$$-\mu_1(m) \leq v_i' \leq \mu_1(m).$$

Because we threshold $\Phi^* s$ to obtain the first coefficient vector $c_1 = H_\theta(\Phi^* s)$, we would like to retain those coefficients indexed by Λ. This is where our bound on m is enforced. Since $\mu_1(m) < 1 - \mu_1(m-1)$ by hypothesis, if we choose $\theta \in \big(\mu_1(m), 1 - \mu_1(m-1)\big)$, when we threshold,

$$c_1 = \left[\begin{array}{c} \Phi_\Lambda^* \Phi_\Lambda \mathbf{1} \\ \hline 0 \end{array} \right]$$

GENERAL IT has thus recovered the index set, Λ, in one iteration. In general, we find that at the nth step in the iteration, those coefficients v_i indexed by Λ are bounded below and above by $1 - (\mu_1(m-1))^n$ and $1 + (\mu_1(m-1))^n$, respectively, while those coefficients not in Λ are zero. Because $\mu_1(m-1) < 1$, we can recover both the index set and the coefficients up to any prescribed tolerance. ■

The next result gives us necessary and sufficient conditions on a signal and a vector of coefficients for that vector of coefficients to be a fixed point of the BLOCK IT algorithm.

Theorem 3.2. *Let s be an arbitrary signal of length d. Let $\Phi = [\Phi_1 \mid \Phi_2]$ be a union of two orthonormal bases. A necessary and sufficient condition for the vector $c = [c_1, c_2]$ to be a fixed point of* BLOCK IT *(hard) is*

$$||\Phi^*(s - \Phi c)||_\infty < \frac{\theta}{2} \tag{1}$$

$$\operatorname{supp}(\epsilon_i) \cap \operatorname{supp}(c_i) = \emptyset \tag{2}$$

$$c_{\min} > \frac{\theta}{2} \tag{3}$$

where $\epsilon_i = \Phi_i^*(s - \Phi c)$.

Proof. First, assume $c = [c_1, c_2]$ is a fixed point. Then

$$c_1 = H_\theta \left[\Phi_1^* s - (\Phi_1^* \Phi_2) c_2 \right], \; c_2 = H_\theta \left[\Phi_2^* s - (\Phi_2^* \Phi_1) c_1 \right]$$

We can write

$$c_1 = \Phi_1^* s - (\Phi_1^* \Phi_2) c_2 - \epsilon_1 \tag{4}$$

$$c_2 = \Phi_2^* s - (\Phi_2^* \Phi_1) c_1 - \epsilon_2 \tag{5}$$

where we must have $||\epsilon_i||_\infty < \frac{\theta}{2}$ for $i = 1, 2$. Substituting Equation (4) into Equation (5), and vice versa, we arrive at $\Phi_1 \epsilon_1 = \Phi_2 \epsilon_2$, and $s - \Phi_1 c_1 = \Phi_2 c_2 + \Phi_1 \epsilon_1$. So we have

$$s - \Phi c = s - \Phi_1 c_1 - \Phi_2 c_2 = \Phi_1 \epsilon_1 = \Phi_2 \epsilon_2$$

Thus we arrive at

$$\Phi_1^*(s - \Phi c) = \epsilon_1 \qquad \text{and} \qquad \Phi_2^*(s - \Phi c) = \epsilon_2$$

Together these give rise to condition (1).

Now suppose the conditions hold. Since $\Phi_1^*(s - \Phi c) = \epsilon_1$ and $\Phi_2^*(s - \Phi c) = \epsilon_2$, we see that $\Phi_1 \epsilon_1 = \Phi_2 \epsilon_2$, and $||\epsilon_i|| < \frac{\theta}{2}$ for $i = 1, 2$. Therefore

$$s - \Phi_1 c_1 = \Phi_2 c_2 + \Phi_1 \epsilon_1 \tag{6}$$

By multiplying Equation (6) through by Φ_2^* and thresholding, and, similarly, multiplying through by Φ_1^* and thresholding, we obtain the result. ■

Observe that if $s = \Phi\alpha$ is an exactly-sparse signal, then α is a fixed point of BLOCK IT. This result does not, however, guarantee that the algorithm converges, upon input s, to the coefficient vector α.

4. EXPERIMENTAL RESULTS

To evaluate the performance of these algorithms, we have tested them with exactly sparse input signals,

$$s = \Phi\alpha = \sum_{k=1}^{m} \alpha_k \phi_k$$

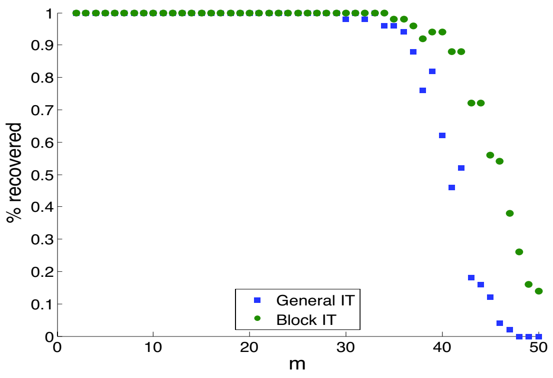

Fig. 1. GENERAL IT (hard) vs. BLOCK IT (hard), $\alpha_k = \mathbf{1}$

where the coefficients α_k are chosen from iid normal distributions or $\alpha_k = \mathbf{1}$. We experimented with the Dirac–Fourier dictionary, which is the collection of d complex exponentials and d impulses. We set $d = 128$. That is $\phi_\omega[t] = (1/\sqrt{d})\mathrm{e}^{2\pi \mathrm{i} t\omega/d}$ for $\omega = 1, \dots, d$, and $\phi_\lambda[t] = \delta_\lambda[t]$ for $\lambda = 1, \dots, d$. Note that this dictionary has coherence $\mu = \frac{1}{\sqrt{d}}$, so the most pessimistic estimate for the maximum number of terms in an exactly sparse signal we can recover is $m = 6$. In our first experiment, we generated sparse signals of the type in Theorem 3.1 with coefficients identically zero or one and ran 50 independent trials for each m-term representation. We checked to see whether the GENERAL IT algorithm recovered the signal within 15 iterations by examining all possible values of θ (in increments of $\frac{1}{100}$) from 0 to 2. Our empirical evidence suggests that GENERAL IT performs better than the theoretical expectations; the algorithm has no problem recovering m-term representations up to $m \approx 30$. See Figure 1.

In another experiment, the α_k were drawn from a normal distribution $N(0, 1)$, and we found, on average over fifty trials, that GENERAL IT (soft) recovered the vectors in Λ better than GENERAL IT (hard). When the coefficients are unrestricted, perhaps soft thresholding is the better choice for unions of othornormal bases; see Figure 2. Figures 1 and 3 show the relative performance of GENERAL IT and BLOCK IT.

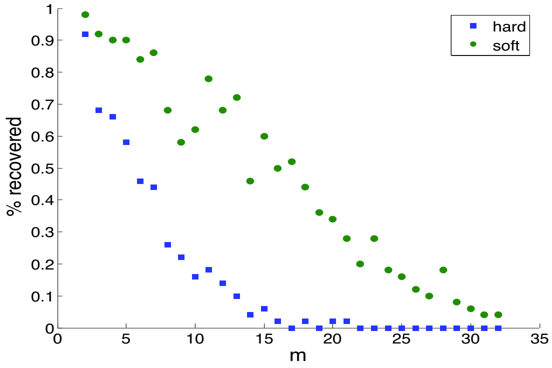

Fig. 2. GENERAL IT (hard) vs. GENERAL IT (soft), $\alpha_k \sim N(0, 1)$

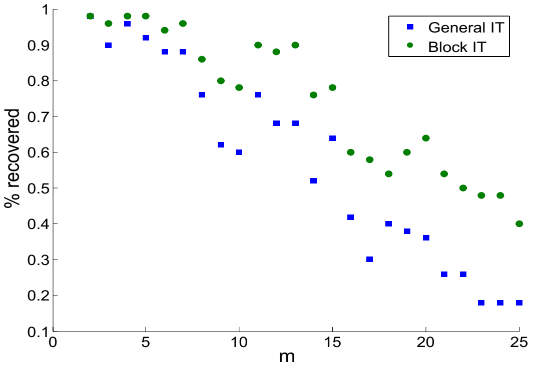

Fig. 3. GENERAL IT (soft) vs. BLOCK IT (soft), $\alpha_k \sim N(0, 1)$

5. CONCLUSION

In this paper, we provide a sufficient condition under which the GENERAL IT algorithm recovers exactly sparse signals when the coefficient vector $\alpha_k = \mathbf{1}$. The definition of cumulative coherence is a natural consequence of the proof of this condition and bolsters our evidence that the cumulative coherence is a critical geometric property of redundant dictionaries for sparse approximation. We also present a necessary and sufficient condition for fixed points of the BLOCK IT algorithm. Our experimental results suggest that BLOCK IT is the better algorithm for unions of orthonormal bases although both algorithms perform equivalently when recovering the index set alone. Little of this analysis suggests how to set the threshold θ. Preliminary theoretical results suggest a relationship which involves the coherence μ of the dictionary and the minimum and maximum values of the coefficients.

6. REFERENCES

[1] I. Daubechies, M. Defrise, and C. Demol, “An iterative thresholding algorithm for linear inverse problems with a sparsity constraint”, *Comm. in Pure and Appl. Math*, vol. 57, 1413–1457, 2004.

[2] M. Elad, “Why simple shrinkage is still relevant for redundant representations?” Technical Report, The Technion, Jan. 2005.

[3] J. L. Starck, Y. Moudden, J. Bobin, M. Elad, D. L. Donoho, “Morphological component analysis,” in proceedings of *SPIE Conference on Wavelets*, vol. 5914, July, 2005.

[4] J. A. Tropp, “Greed is good: Algorithmic results for sparse approximation.” *IEEE Trans. Inform. Theory*, vol. 50, no. 10, pp. 2231-2242, Oct. 2004.