FAST SPARSE RECOVERY FOR ANY RIP-1 MATRIX

Eric Price

UT Austin
ectprice@cs.utexas.edu

ABSTRACT

The Restricted Isometry Property (RIP) is a useful measure of which measurement matrices will work for sparse recovery. The RIP-1 is an L1 variant of the RIP that can be satisfied by sparse matrices, allowing for faster embedding and recovery. While L1 minimization is guaranteed to work for all matrices satisfying the RIP-1, faster iterative techniques were only known to work when the matrix is the adjacency of an expander graph. We show that Sequential Sparse Matching Pursuit (SSMP) works on all matrices satisfying the RIP-1, giving the first demonstration of near-linear recovery time for arbitrary RIP-1 matrices.

Index Terms— Sparse recovery, compressed sensing

1. INTRODUCTION

Compressed sensing is a popular framework for signal recovery developed over the past decade. The aim is to recover a $k$-sparse vector $x \in \mathbb{R}^n$ from a noisy measurement $b = Ax + \mu$, where $A \in \mathbb{R}^{m \times n}$ is a well-chosen “measurement matrix” with $m \ll n$ and $\mu$ is arbitrary noise. From this observation $b$, the goal is to recover an estimate $\hat{x}$ of $x$ with

$$\|\hat{x} - x\| = O(\|\mu\|).$$

Much of the compressed sensing literature is based on the Restricted Isometry Property (RIP). Unfortunately, in the parameter regime of interest the RIP only holds for dense matrices [1], which are inefficient to store and manipulate. This led to the introduction of the RIP-1, an $\ell_1$ variant of the RIP, which is achievable with sparse matrices and is sufficient for L1 minimization to achieve robust sparse recovery [2]:

**Definition 1.1.** A matrix $A \in \mathbb{R}^{m \times n}$ satisfies the $(k, \epsilon)$ RIP-1 if, for all $k$-sparse $x \in \mathbb{R}^n$,

$$(1 - \epsilon) \|Ax\|_1 \leq \|x\|_1 \leq \|Ax\|_1.$$

For binary matrices, the RIP-1 is essentially equivalent to $A$ being the adjacency matrix of an unbalanced bipartite expander [2]. As an example, a random binary matrix with $O(\frac{1}{\epsilon^2} \log(n/k))$ ones per column and $m = O(\frac{1}{\epsilon^2} k \log(n/k))$ will usually satisfy the $(k, \epsilon)$ RIP-1. However, the definition encompasses nonbinary matrices.

One interesting example of nonbinary RIP-1 matrices comes from randomly flipping the sign of each entry of an expander adjacency matrix. The result, which behaves like a random sparse $\{0, \pm 1\}$ matrix, is quite similar to a COUNTSKETCH matrix [3]. The random signs cause the noise to largely cancel itself out, leading to better performance than binary matrices like COUNT-MIN [4, 5].

In this paper, we show that Sequential Sparse Matching Pursuit (SSMP), an iterative algorithm introduced in [6], works on arbitrary RIP-1 matrices. Previously, SSMP was only known to work for expander adjacency matrices, as did all other fast iterative methods [7, 8, 9, 10, 11].

**Theorem 1.2.** Let $A \in \mathbb{R}^{m \times n}$ satisfy the $(ck, \frac{1}{10})$ RIP-1 for some (sufficiently large) constant $c$. Let $x \in \mathbb{R}^n$ be $k$-sparse, $\mu \in \mathbb{R}^m$, and let $\hat{x}$ be the result of running SSMP on $(A, Ax + \mu, k)$. Then

$$\|\hat{x} - x\|_1 = O(\|\mu\|_1).$$

(1)

Alternatively, one could give a guarantee for non-sparse inputs, e.g. for all $x \in \mathbb{R}^n$ SSMP gives $\hat{x}$ with

$$\|\hat{x} - x\|_1 = O(\min_{k\text{-sparse } x'} \|x' - x\|_1).$$
This follows from Theorem 1.2 and the triangle inequality, since we can set \( \mu = A(x' - x) \) and then \( \|\mu\|_1 \leq \sum_j \|A(x' - x)_j\|_1 \leq \frac{1}{1 - \epsilon} \|x' - x\|_1 \).

1.1. Related work.

The RIP-1 was introduced in [2], where it was shown to imply robust recovery using L1 minimization. This gives fast embedding times, since the measurement matrix can be sparse, but not fast recovery. Subsequently, a number of fast iterative methods have been proposed. Expander Matching Pursuit (EMP) [7], Sparse Matching Pursuit (SMP) [8], and Sequential Sparse Matching Pursuit (SSMP) [6] all use \( \tilde{O}(n) \) recovery time and get the robustness guarantee (1) (see also [12] for a survey). Another set of work applies to “exact” recovery with \( \mu = 0 \) [9, 10] or very small [11]. The phase transition of some of these methods is explored in [13]. However, all of these iterative methods use more than just the RIP-1: they expect the matrix to be the adjacency of an expander.

2. SSMP ALGORITHM

The SSMP algorithm is shown as Algorithm 2.1. It uses the following definition: for any vector \( x \in \mathbb{R}^n \), define \( H_k(x) \in \mathbb{R}^n \) to be the restriction of \( x \) to its \( k \) largest coefficients. In [6] it was shown how to maintain a data structure to implement the algorithm in \( O(n d(d + \log n)) \) time per inner loop, where \( d \) is the matrix column sparsity (i.e. typically \( d = O(\log \frac{n}{k}) \)). The method works for arbitrary RIP-1 matrices, not just expander adjacency matrices.

```plaintext
1: procedure SSMP(\( A, b, k \))
2: \( \tilde{x}^0 = 0 \)
3: for \( j \leftarrow 1, \ldots, T = O(\log \|x\|_1 / \|\mu\|_1) \) do
4: \( \tilde{x}^{j,0} \leftarrow \tilde{x}^{j-1} \)
5: for \( a \leftarrow 1, \ldots, r = (c - 1)k + 1 \) do
6: \( (i, z) \leftarrow \arg\min_{(i,z)} \|b - A(\tilde{x}^{j,a-1} + ze_i)\|_1 \)
7: \( \tilde{x}^{j,a} \leftarrow \tilde{x}^{j,a-1} + ze_i \)
8: end for
9: \( \tilde{x}^j \leftarrow H_k(\tilde{x}^{j,r}) \) \Comment{Restrict to \( k \) terms}
10: end for
11: return \( x' = \tilde{x}^T \)
12: end procedure
```

Algorithm 2.1: SSMP.

The SSMP algorithm iteratively refines its estimate \( \tilde{x}^j \) of the signal \( x \). The inner loop adds a single coordinate to the estimate, which we show decreases the residual error \( \|A(x - \tilde{x}^{j,a}) + \mu\|_1 \) by at least a \( 1 - \frac{1}{O(k + \epsilon)} \) factor, unless we have already converged to \( \|x - \tilde{x}^{j,a}\|_1 = O(\|\mu\|_1) \).

After \( r = O(k) \) rounds, we will have that

\[
\|A(x - \tilde{x}^{j,r}) + \mu\|_1 \leq \frac{1}{8} \|A(x - \tilde{x}^{j,0}) + \mu\|_1.
\]

At this point, having updated our estimate many times, it is starting to lose the sparsity we need to apply the RIP-1. Therefore in our outer loop, we resparsify \( \tilde{x}^{j,a} \) back down to \( k \) terms. This increases \( \|A(x - \tilde{x}^{j,a}) + \mu\|_1 \) by at most a \( 2 + O(\epsilon) \) factor, so we get

\[
\|A(x - \tilde{x}^{j+1,0}) + \mu\|_1 \leq \frac{1}{2} \|A(x - \tilde{x}^{j,0}) + \mu\|_1
\]

in every round, until it converges to \( O(\|\mu\|_1) \).

The tricky part, and the novel part of this paper, is showing progress in each step of the inner loop. The original argument relied on properties of expander matrices; here, we solely consider geometry and the RIP-1. We discuss it in Section 3. The other parts are straightforward, and covered in Section 4.

3. PROOF OF SEQUENTIAL PROGRESS

We start with a geometric lemma about the \( \ell_1 \) norm:

**Lemma 3.1.** Let \( x_1, \ldots, x_s, \mu \in \mathbb{R}^m \), and \( z = \mu + \sum_i x_i \). Suppose that \( \|\mu\|_1 < c \|z\|_1 \) and

\[
(1 - \epsilon)\sum_{i=1}^{s} \|x_i\|_1 \leq \sum_{i=1}^{s} x_i \|_{1'}
\]

for some constants \( 0 \leq c, \epsilon < 1/2 \). Then there exists an \( i \) such that \( \|z - x_i\|_1 \leq (1 - \frac{1}{8}(1 - 2\epsilon - 5\epsilon)) \|z\|_1 \).

Intuitively, the condition means the \( x_i \) form a chain that is nearly at its maximal length; it is nearly “taut.” Almost all the mass needs to be oriented toward the final vector \( z \); very little is “slack” that can be “wasted” by moving in superfluous directions. On average, the \( x_i \) are pointed in the right direction and fairly large; hence at least one \( x_i \) is both of these.
Proof. Define the “projection” operator \( p(a,b) \) of \( a \) onto \( b \) to be the coordinatewise nearest neighbor of \( a \) to the intervals \([0,b_i]\) for each coordinate \( i \). That is, for positive coordinates \( b_i \geq 0 \), we define

\[
p(a,b)_i = \begin{cases} 
0 & \text{if } a_i < 0 \\
 a_i & \text{if } 0 \leq a_i \leq b_i \\
b_i & \text{if } a_i > b_i
\end{cases}
\]

and analogously for negative coordinates (so \( p(a,b)_i = -p(-a,-b)_i \)). As a property of this operator, for all \( a,b \) we have \( \|b - p(a,b)\|_1 = \|b\|_1 - \|p(a,b)\|_1 \).

For simplicity of notation, let \( v_i = x_i \) for \( i \geq 1 \) and \( v_0 = \mu \), so \( z = \sum_{i=0}^s v_i \). Let \( u_i = p(v_i,z) \), and \( w_i = \|v_i - u_i\|_1 = \|v_i\|_1 - \|u_i\|_1 \). Then \( u_i \) is the part of \( v_i \) moving in the right direction, and \( w_i \) is the amount of mass “wasted” in the wrong direction. In particular,

\[
\|z - v_i\|_1 = \|z - u_i\|_1 + \|u_i - v_i\|_1 = \|z\|_1 - \|u_i\|_1 + w_i
\]

(2)

So we just want to show that some \( i \) has large \( \|u_i\|_1 - w_i \). First we will show that \( \|u_i\|_1 \) is large on average, then that \( w_i \) is small on average, and hence the difference is large for at least one \( i \). First, we claim

\[
\sum_{i=0}^s \|u_i\|_1 \geq \|z\|_1.
\]

(3)

We do this by showing that for any coordinate \( j \), \( \sum_{i=0}^s |(u_i)_j| \geq |z_j| \). WLOG suppose \( z_j \geq 0 \), so \( (u_i)_j \geq 0 \) for all \( i \). Then by the definition of projection, for each \( i \) either \( (u_i)_j \geq (v_i)_j \) or \( (u_i)_j = z_j \). If the latter ever happens, \( \sum_{i=0}^s (u_i)_j \geq \max_i (u_i)_j = z_j \); otherwise, \( \sum_{i=0}^s (u_i)_j \geq \sum_{i=0}^s (v_i)_j = z_j \).

Now, consider showing that the \( w_i \) are small. Intuitively, this is “wasted” mass that doesn’t help reach the goal: it’s in the wrong direction, or overshooting the mark. We don’t have enough slack to waste much mass, so \( \sum_{i=1}^s w_i \) must be small. In equations,

\[
\frac{1}{1 - \epsilon} \left\| \sum_{i=1}^s x_i \right\|_1 \geq \sum_{i=1}^s \|x_i\|_1 \\
= \sum_{i=1}^s (w_i + \|u_i\|_1) \\
\geq \|z\|_1 - \|u_0\|_1 + \sum_{i=1}^s w_i \\
\geq \left( \left\| \sum_{i=1}^s x_i \right\|_1 - \|\mu\|_1 \right) - \|u_0\|_1 + \sum_{i=1}^s w_i
\]

and hence

\[
\frac{\epsilon}{1 - \epsilon} \left( \sum_{i=1}^s x_i \right) \geq -\|\mu\|_1 + \sum_{i=1}^s w_i
\]

\[
\frac{\epsilon}{1 - \epsilon} (\|z\|_1 + \|\mu\|_1) \geq -\|\mu\|_1 + \sum_{i=1}^s w_i
\]

so

\[
\sum_{i=1}^s w_i \leq \left( 2 + \frac{\epsilon}{1 - \epsilon} \right) \|\mu\|_1 + \frac{\epsilon}{1 - \epsilon} \|z\|_1.
\]

(4)

Hence we have that the “non-wasted” mass \( u_i \) is large, and the “wasted” mass \( w_i \) is small. We just need to show that some particular \( i \) has large \( \|u_i\|_1 - w_i \), but this will be true on average.

Subtracting Equation 4 from Equation 3,

\[
\sum_{i=1}^s \|u_i\|_1 - w_i \geq (1 - \frac{\epsilon}{1 - \epsilon}) \|z\|_1.
\]

(5)

So for \( \epsilon \leq 1/2 \),

\[
\sum_{i=1}^s \|u_i\|_1 - w_i \geq (1 - 2\epsilon - 5c) \|z\|_1.
\]

Let \( j \) be such that \( \|u_j\|_1 - w_j \) is above the mean. Then by Equation 5 and Equation 2,

\[
\|u_j\|_1 - w_j \geq \frac{1}{s} (1 - 2\epsilon - 5c) \|z\|_1 \\
\|z - x_j\|_1 = \|z\|_1 - \|u_j\|_1 + w_j \\
\geq (1 - \frac{1}{s} (1 - 2\epsilon - 5c)) \|z\|_1
\]

as desired.

Now we can apply Lemma 3.1 to matrices satisfying the RIP:

**Lemma 3.2.** Suppose \( A \) satisfies an RIP-1 of order \((s, 1/10), s > 1\). If \( y \) is \( s \)-sparse, and \( \|w\|_1 \leq \frac{1}{s} \|y\|_1 \), then there exists a \( 1 \)-sparse \( z \) such that

\[
\|A(y - z) + w\|_1 \leq \epsilon \frac{1}{s} \|Ay + w\|_1.
\]
Proof. First, note that
\[ \|w\|_1 \leq \frac{1}{30(1 - \epsilon)} \|Ay\|_1 \leq \frac{1}{27} \|Ay\|_1 \leq \frac{1}{26} \|Ay + w\|_1 . \]

Split \( y = y_1 + y_2 + \ldots + y_s \), for orthogonal 1-sparse \( y_i \). Let \( v_i = Ay_i \). Let \( \epsilon = 1/10 \), so we have by the RIP-1 of order \((s, \epsilon)\) that
\[ (1 - \epsilon) \|v_i\|_1 \leq \|y_i\|_1 \leq \|v_i\|_1 \]
and
\[ \left\| \sum_{i=1}^s v_i \right\|_1 = \|Ay\|_1 \geq \|y\|_1 \geq (1 - \epsilon) \sum_{i=1}^s \|v_i\|_1 . \]

Hence we can apply Lemma 3.1: for any noise vector \( w \) with \( \|w\|_1 \leq c \|Ay + w\|_1 \), there exists a \( j \) with
\[ \|A(y - y_j) + w\|_1 \leq (1 - \frac{1}{s}(1 - 2\epsilon - 5\epsilon)) \|Ay + w\|_1 . \]

For \( \epsilon \leq 1/10 \) and \( c \leq 1/25 \), this gives
\[ \|A(y - y_j) + w\|_1 \leq (1 - \frac{3}{5s}) \|Ay + w\|_1 \leq e^{-\frac{1}{25}} \|Ay + w\|_1 \]
So \( z = y_j \) satisfies the desired result. \( \square \)

4. ANALYSIS OF SSMP

We now apply our results to analyze SSMP. Since \( \hat{x}^j,a \) is \((k + a)\)-sparse, \( \hat{x}^j,a - x \) is \((2k + a)\)-sparse. Hence we have the following corollary of Lemma 3.2:

Corollary 4.1. In SSMP, if \( A \) satisfies an RIP-1 of order \((\log n)^{1/2} \), and \( \|\mu\|_1 \leq \frac{1}{30} \|\hat{x}^j,a - x\|_1 \), then
\[ \|A\hat{x}^j,a+1 - b\|_1 \leq e^{-\frac{1}{2(k + a + 1)}} \|A\hat{x}^j,a - b\|_1 \]
for all \( j \) and \( a \).

If we telescope this, with \( H_n \approx \log n \) denoting the \( n \)th harmonic number \( \sum_{i=1}^n \frac{1}{i} \), we have
\[ \|A\hat{x}^{j,t} - b\|_1 \leq e^{-\frac{n}{2} (H_{2k+1} - H_{2k-1})} \|A\hat{x}^{j,0} - b\|_1 . \]

Setting \( t = ck + 1 \), since \( H_{(2+c)k} - H_{2k-1} > \log \frac{2+c}{2} \), we have
\[ \|A\hat{x}^{j,ck+1} - b\|_1 \leq \sqrt{\frac{2}{2 + c}} \|A\hat{x}^{j,0} - b\|_1 . \]

For \( c = 128 \), this gives
\[ \|A\hat{x}^{j,ck+1} - b\|_1 \leq \frac{1}{8} \|A\hat{x}^{j,0} - b\|_1 . \]

Because \( A \) satisfies the RIP-1 we know
\[ \|A(\hat{x}^{j,1,t} - x) - \mu\|_1 \geq \|A(\hat{x}^{j,1,t} - x)\|_1 - \|\mu\|_1 \geq (1 - \epsilon) \|\hat{x}^{j,1,t} - x\|_1 - \|\mu\|_1 \]
so since \( \epsilon < 1/2 \),
\[ \|\hat{x}^{j,1,t} - x\|_1 \leq 2 \|A\hat{x}^{j,1,t} - b\|_1 + 2 \|\mu\|_1 \leq \frac{1}{4} \|A\hat{x}^{j} - b\|_1 + 2 \|\mu\|_1 \leq \frac{1}{4} \|A(\hat{x} - x)\|_1 + \frac{9}{4} \|\mu\|_1 \leq \frac{1}{4} \|\hat{x} - x\|_1 + \frac{9}{4} \|\mu\|_1 . \]

Since \( x \) is \( k \)-sparse and \( \hat{x}^{j+1} = H_k(\hat{x}^{j+1,t}) \) is the nearest \( k \)-sparse vector to \( \hat{x}^{j+1,t} \), we have by the triangle inequality that
\[ \|\hat{x}^{j+1,t} - x\|_1 \leq 2 \|\hat{x}^{j+1,t} - x\|_1 \leq \frac{1}{2} \|\hat{x}^{j+1} - x\|_1 + \frac{9}{2} \|\mu\|_1 \leq \frac{1}{18} \|\hat{x}^{j+1} - x\|_1 . \]

Now, if \( \|\mu\|_1 \leq \frac{1}{18} \|\hat{x}^{j+1} - x\|_1 \), we have
\[ \|\hat{x}^{j+1} - x\|_1 \leq \frac{3}{4} \|\hat{x}^{j} - x\|_1 . \]
This means the error decreases to \( O(\|\mu\|_1) \) in \( T = O(\log \|\mu\|_1) \) iterations, after which it never grows larger then \( O(\|\mu\|_1) \). This gives Theorem 1.2.

5. CONCLUSION

We have shown that the SSMP algorithm gives deterministic \( \ell_1 \) sparse recovery for arbitrary RIP-1 matrices, not just those binary ones. This allows us to use matrices, such as the COUNTSKETCH matrix, that are also suitable for high-probability \( \ell_2 \) recovery. One interesting question is whether SSMP on such a matrix will have high-probability \( \ell_2 \) recovery. One can show that one of the early estimates \( x^{0,a} \) will be close to the Count-Sketch estimate, and hence a good \( \ell_2 \) estimate with high probability; however, we do not know how to show that the later adjustments do not combine into a bad \( \ell_2 \) result.
6. REFERENCES


