# On the Power of Adaptivity in Sparse Recovery

Piotr Indyk Eric Price David P. Woodruff

August 18, 2011

**Abstract**

The goal of (stable) sparse recovery is to recover a $k$-sparse approximation $x^*$ of a vector $x$ from linear measurements of $x$. Specifically, the goal is to recover $x^*$ such that

$$\|x - x^*\|_p \leq C \min_{k\text{-sparse } x'} \|x - x'\|_q$$

for some constant $C$ and norm parameters $p$ and $q$. It is known that, for $p = q = 1$ or $p = q = 2$, this task can be accomplished using $m = O(k \log(n/k))$ *non-adaptive* measurements [3] and that this bound is tight [9, 12, 28].

In this paper we show that if one is allowed to perform measurements that are *adaptive* , then the number of measurements can be considerably reduced. Specifically, for $C = 1 + \epsilon$ and $p = q = 2$ we show

- A scheme with $m = O(\frac{1}{\epsilon} k \log\log(n\epsilon/k))$ measurements that uses $O(\log^* k \cdot \log\log(n\epsilon/k))$ rounds. This is a significant improvement over the best possible non-adaptive bound.
- A scheme with $m = O(\frac{1}{\epsilon} k \log(k/\epsilon) + k \log(n/k))$ measurements that uses *two* rounds. This improves over the best possible non-adaptive bound.

To the best of our knowledge, these are the first results of this type.

# 1 Introduction

In recent years, a new "linear" approach for obtaining a succinct approximate representation of $n$-dimensional vectors (or signals) has been discovered. For any signal $x$, the representation is equal to $Ax$, where $A$ is an $m \times n$ matrix, or possibly a random variable chosen from some distribution over such matrices. The vector $Ax$ is often referred to as the *measurement vector* or *linear sketch* of $x$. Although $m$ is typically much smaller than $n$, the sketch $Ax$ often contains plenty of useful information about the signal $x$.

A particularly useful and well-studied problem is that of *stable sparse recovery*. We say that a vector $x'$ is $k$-sparse if it has at most $k$ non-zero coordinates. The sparse recovery problem is typically defined as follows: for some norm parameters $p$ and $q$ and an approximation factor $C > 0$, given $Ax$, recover an "approximation" vector $x^*$ such that

$$\|x - x^*\|_p \leq C \min_{k\text{-sparse } x'} \left\|x - x'\right\|_q \tag{1}$$

(this inequality is often referred to as $\ell_p/\ell_q$ *guarantee*). If the matrix $A$ is random, then Equation (1) should hold for each $x$ with some probability (say, 2/3). Sparse recovery has a tremendous number of applications in areas such as compressive sensing of signals [3, 10], genetic data acquisition and analysis [29, 2] and data stream algorithms[1] [27, 21]; the latter includes applications to network monitoring and data analysis.

It is known [3] that there exist matrices $A$ and associated recovery algorithms that produce approximations $x^*$ satisfying Equation (1) with $p = q = 1$, constant approximation factor $C$, and sketch length

$$m = O(k \log(n/k)) \tag{2}$$

A similar bound, albeit using random matrices $A$, was later obtained for $p = q = 2$ [16] (building on [5, 6, 7]). Specifically, for $C = 1 + \epsilon$, they provide a distribution over matrices $A$ with

$$m = O(\frac{1}{\epsilon} k \log(n/k)) \tag{3}$$

rows, together with an associated recovery algorithm.

It is also known that the bound in Equation (2) is asymptotically optimal for some constant $C$ and $p = q = 1$, see [9] and [12] (building on [13, 17, 24]). The bound of [9] also extends to the randomized case and $p = q = 2$. For $C = 1 + \epsilon$, a lower bound of $m = \Omega(\frac{1}{\epsilon} k \log(n/k))$ was recently shown [28] for the randomized case and $p = q = 2$, improving upon the earlier work of [9] and showing the dependence on $\epsilon$ is optimal. The necessity of the "extra" logarithmic factor multiplying $k$ is quite unfortunate: the sketch length determines the "compression rate", and for large $n$ any logarithmic factor can worsen that rate tenfold.

In this paper we show that this extra factor can be greatly reduced if we allow the measurement process to be *adaptive*. In the adaptive case, the measurements are chosen in rounds, and the choice of the measurements in each round depends on the outcome of the measurements in the previous rounds. The adaptive measurement model has received a fair amount of attention in the recent years [22, 4, 20, 19, 25, 1], see also [8]. In particular [19] showed that adaptivity helps reducing the approximation error in the presence of random noise. However, no asymptotic improvement to the number of measurements needed for sparse recovery (as in Equation (1)) was previously known.

**Results** In this paper we show that adaptivity can lead to very significant improvements in the number of measurements over the bounds in Equations (2) and (3). We consider randomized sparse recovery with $\ell_2/\ell_2$ guarantee, and show two results:

[1] In streaming applications, a data stream is modeled as a sequence of linear operations on an (implicit) vector x. Example operations include increments or decrements of $x$'s coordinates. Since such operations can be directly performed on the linear sketch $Ax$, one can maintain the sketch using only $O(m)$ words.

1. A scheme with $m = O(\frac{1}{\epsilon} k \log\log(n\epsilon/k))$ measurements and an approximation factor $C = 1+\epsilon$. For low values of $k$ this provides an *exponential* improvement over the best possible non-adaptive bound. The scheme uses $O(\log^* k \cdot \log\log(n\epsilon/k))$ rounds.

2. A scheme with $m = O(\frac{1}{\epsilon} k \log(k/\epsilon) + k \log(n/k))$ and an approximation factor $C = 1 + \epsilon$. For low values of $k$ and $\epsilon$ this offers a significant improvement over the best possible non-adaptive bound, since the dependence on $n$ and $\epsilon$ is "split" between two terms. The scheme uses only two rounds.

**Implications** Our new bounds lead to potentially significant improvements to efficiency of sparse recovery schemes in a number of application domains. Naturally, not *all* applications support adaptive measurements. For example, network monitoring requires the measurements to be performed simultaneously, since we cannot ask the network to "re-run" the packets all over again. However, a surprising number of applications are capable of supporting adaptivity. For example:

- Streaming algorithms for data analysis: since each measurement round can be implemented by one pass over the data, adaptive schemes simply correspond to multiple-pass streaming algorithms (see [26] for some examples of such algorithms).

- Compressed sensing of signals: several architectures for compressive sensing, e.g., the single-pixel camera of [11], already perform the measurements in a sequential manner. In such cases the measurements can be made adaptive[2]. Other architectures supporting adaptivity are under development [8].

- Genetic data analysis and acqusition: as above.

Therefore, it seems likely that the results in this paper will be applicable in a wide variety of scenarios.

**Techniques** On a high-level, both of our schemes follow the same two-step process. First, we reduce the problem of finding the best $k$-sparse approximation to the problem of finding the best 1-sparse approximation (using relatively standard techniques). This is followed by solving the latter (simpler) problem.

The first scheme starts by "isolating" most of of the large coefficients by randomly sampling $\approx \epsilon/k$ fraction of the coordinates; this mostly follows the approach of [16] (cf. [15]). The crux of the algorithm is in the identification of the isolated coefficients. Note that in order to accomplish this using $O(\log\log n)$ measurements (as opposed to $O(\log n)$ achieved by the "standard" binary search algorithm) we need to "extract" significantly more than one bit of information per measurements. To achieve this, we proceed as follows. First, observe that if the given vector (say, $z$) is *exactly* 1-sparse, then one can extract the position of the non-zero entry (say $z_j$) from *two* measurements $a(z) = \sum_i z_i$, and $b(z) = \sum_i i z_i$, since $b(z)/a(z) = j$. A similar algorithm works even if $z$ contains some "very small" non-zero entries: we just round $b(z)/a(z)$ to the nearest integer. This algorithm is a special case of a general algorithm that achieves $O(\log n/\log SNR)$ measurements to identify a single coordinate $x_j$ among $n$ coordinates, where $SNR = x_j^2/\|x_{[n]\setminus j}\|^2$ (SNR stands for signal-to-noise ratio). This is optimal as a function of $n$ and the SNR [9].

A natural approach would then be to partition $[n]$ into two sets $\{1, \ldots, n/2\}$ and $\{n/2+1, \ldots n\}$, find the heavier of the two sets, and recurse. This would take $O(\log n)$ rounds. The key observation is that not only do we recurse on a smaller-sized set of coordinates, but the SNR has also increased since $x_j^2$ has remained the same but the squared norm of the tail has dropped by a constant factor. Therefore in the next round we can afford to partition our set into more than two sets, since as long as we keep the ratio of

[2]We note that, in any realistic sensing system, minimizing the number of measurements is only one of several considerations. Other factors include: minimizing the computation time, minimizing the amount of communication needed to transfer the measurement matrices to the sensor, satisfying constraints on the measurement matrix imposed by the hardware etc. A detailed cost analysis covering all of these factors is architecture-specific, and beyond the scope of this paper.

$\log(\#$ of sets $)$ and $\log SNR$ constant, we only need $O(1)$ measurements per round. This ultimately leads to a scheme that finishes after $O(\log\log n)$ rounds.

In the second scheme, we start by hashing the coordinates into a universe of size polynomial in $k$ and $1/\epsilon$, in a way that approximately preserves the top coefficients without introducing spurious ones, and in such a way that the mass of the tail of the vector does not increase significantly by hashing. This idea is inspired by techniques in the data stream literature for estimating moments [23, 30] (cf. [5, 7, 14]). Here, though, we need stronger error bounds. This enables us to identify the positions of those coefficients (in the hashed space) using only $O(\frac{1}{\epsilon}k\log(k/\epsilon))$ measurements. Once this is done, for each large coefficient $i$ in the hash space, we identify the actual large coefficient in the preimage of $i$. This can be achieved using the number of measurements that does not depend on $\epsilon$.

## 2 Preliminaries

We start from a few definitions. Let $x$ be an $n$-dimensional vector.

**Definition 2.1.** *Define*

$$H_k(x) = \arg\max_{\substack{S\in[n]\\ |S|=k}} \|x_S\|_2$$

*to be the largest $k$ coefficients in $x$.*

**Definition 2.2.** *For any vector $x$, we define the "heavy hitters" to be those elements that are both (i) in the top $k$ and (ii) large relative to the mass outside the top $k$. We define*

$$H_{k,\epsilon}(x) = \{j \in H_k(x) \mid x_j^2 \geq \epsilon \left\| x_{\overline{H_k(x)}} \right\|_2^2\}$$

**Definition 2.3.** *Define the error*

$$\mathrm{Err}^2(x,k) = \left\| x_{\overline{H_k(x)}} \right\|_2^2$$

For the sake of clarity, the analysis of the algorithm in section 4 assumes that the entries of $x$ are sorted by the absolute value (i.e., we have $|x_1| \geq |x_2| \geq \ldots \geq |x_n|$). In this case, the set $H_k(x)$ is equal to $[k]$; this allows us to simplify the notation and avoid double subscripts. The algorithms themselves are invariant under the permutation of the coordinates of $x$.

**Running times of the recovery algorithms** In the non-adaptive model, the running time of the recovery algorithm is well-defined: it is the number of operations performed by a procedure that takes $Ax$ as its input and produces an approximation $x^*$ to $x$. The time needed to generate the measurement vectors $A$, or to encode the vector $x$ using $A$, is not included. In the adaptive case, the distinction between the matrix generation, encoding and recovery procedures does not exist, since new measurements are generated based on the values of the prior ones. Moreover, the running time of the measurement generation procedure heavily depends on the representation of the matrix. If we suppose that we may output the matrix in sparse form and receive encodings in time bounded by the number of non-zero entries in the matrix, our algorithms run in $n\log^{O(1)} n$ time.

## 3 Full adaptivity

This section shows how to perform $k$-sparse recovery with $O(k\log\log(n/k))$ measurements. The core of our algorithm is a method for performing 1-sparse recovery with $O(\log\log n)$ measurements. We then extend this to $k$-sparse recovery via repeated subsampling.

### 3.1 1-sparse recovery

This section discusses recovery of 1-sparse vectors with $O(\log\log n)$ adaptive measurements. First, in Lemma 3.1 we show that if the heavy hitter $x_j$ is $\Omega(n)$ times larger than the $\ell_2$ error ($x_j$ is "$\Omega(n)$-heavy"), we can find it with two non-adaptive measurements. This corresponds to non-adaptive 1-sparse recovery with approximation factor $C = \Theta(n)$; achieving this with $O(1)$ measurements is unsurprising, because the lower bound [9] is $\Omega(\log_{1+C} n)$.

Lemma 3.1 is not directly very useful, since $x_j$ is unlikely to be that large. However, if $x_j$ is $D$ times larger than everything else, we can partition the coordinates of $x$ into $D$ random blocks of size $N/D$ and perform dimensionality reduction on each block. The result will in expectation be a vector of size $D$ where the block containing $j$ is $D$ times larger than anything else. The first lemma applies, so we can recover the block containing $j$, which has a $1/\sqrt{D}$ fraction of the $\ell_2$ noise. Lemma 3.2 gives this result.

We then have that with two non-adaptive measurements of a $D$-heavy hitter we can restrict to a subset where it is an $\Omega(D^{3/2})$-heavy hitter. Iterating $\log\log n$ times gives the result, as shown in Lemma 3.3.

**Lemma 3.1.** *Suppose there exists a $j$ with $|x_j| \geq C\frac{n}{\sqrt{\delta}}\left\|x_{[n]\setminus\{j\}}\right\|_2$ for some constant $C$. Then two non-adaptive measurements suffice to recover $j$ with probability $1-\delta$.*

*Proof.* Let $s\colon [n] \to \{\pm 1\}$ be chosen from a 2-wise independent hash family. Perform the measurements $a(x) = \sum s(i)x_i$ and $b(x) = \sum (n+i)s(i)x_i$. For recovery, output the closest integer to $b/a - n$.

Let $z = x_{[n]\setminus\{j\}}$. Then $\mathbb{E}[a(z)^2] = \|z\|_2^2$ and $\mathbb{E}[b(z)^2] \leq 4n^2\|z\|_2^2$. Hence with probability at least $1-2\delta$, we have both

$$|a(z)| \leq \sqrt{1/\delta}\,\|z\|_2$$
$$|b(z)| \leq 2n\sqrt{1/\delta}\,\|z\|_2$$

Thus

$$\begin{aligned}\frac{b(x)}{a(x)} =& \frac{s(j)(n+j)x_j + b(z)}{s(j)x_j + a(z)}\\ \left|\frac{b(x)}{a(x)} - (n+j)\right| =& \left|\frac{b(z) - (n+j)a(z)}{s(j)x_j + a(z)}\right|\\ \leq& \frac{|b(z)| + (n+j)\,|a(z)|}{||x_j| - |a(z)||}\\ \leq& \frac{4n\sqrt{1/\delta}\,\|z\|_2}{||x_j| - |a(z)||}\end{aligned}$$

Suppose $|x_j| > (8n+1)\sqrt{1/\delta}\,\|z\|_2$. Then

$$\begin{aligned}\left|\frac{b(x)}{a(x)} - (n+j)\right| <& \frac{4n\sqrt{1/\delta}\,\|z\|_2}{8n\sqrt{1/\delta}\,\|z\|_2}\\ =& 1/2\end{aligned}$$

so $\hat{\imath} = j$. □

**Lemma 3.2.** *Suppose there exists a $j$ with $|x_j| \geq C\frac{B^2}{\delta^2}\left\|x_{[n]\setminus\{j\}}\right\|_2$ for some constant $C$ and parameters $B$ and $\delta$. Then with two non-adaptive measurements, with probability $1-\delta$ we can find a set $S \subset [n]$ such that $j \in S$ and $\left\|x_{S\setminus\{j\}}\right\|_2 \leq \left\|x_{[n]\setminus\{j\}}\right\|_2/B$ and $|S| \leq 1 + n/B^2$.*

*Proof.* Let $D = B^2/\delta$, and let $h\colon [n] \to [D]$ and $s\colon [n] \to \{\pm 1\}$ be chosen from pairwise independent hash families. Then define $S_p = \{i \in [n] \mid h(i) = p\}$. Define the matrix $A \in \mathbb{R}^{D\times n}$ by $A_{h(i),i} = s(i)$ and $A_{p,i} = 0$ elsewhere. Then

$$(Az)_p = \sum_{i\in S_p} s(i) z_i.$$

Let $p^* = h(j)$ and $y = x_{[n]\setminus\{j\}}$. We have that

$$\begin{aligned}
\mathbb{E}[|S_{p^*}|] =& 1 + (n-1)/D \\
\mathbb{E}[(Ay)^2_{p^*}] = \mathbb{E}[\left\|y_{S_{p^*}}\right\|_2^2] =& \|y\|_2^2 / D \\
\mathbb{E}[\|Ay\|_2^2] =& \|y\|_2^2
\end{aligned}$$

Hence by Chebyshev's inequality, with probability at least $1 - 4\delta$ all of the following hold:

$$|S_{p^*}| \le 1 + (n-1)/(D\delta) \le 1 + n/B^2 \tag{4}$$

$$\left\|y_{S_{p^*}}\right\|_2 \le \|y\|_2 / \sqrt{D\delta} \tag{5}$$

$$|(Ay)_{p^*}| \le \|y\|_2 / \sqrt{D\delta} \tag{6}$$

$$\|Ay\|_2 \le \|y\|_2 / \sqrt{\delta}. \tag{7}$$

The combination of (6) and (7) imply

$$\begin{aligned}
|(Ax)_{p^*}| \ge |x_j| - |(Ay)_{p^*}| \ge& (CD/\delta - 1/\sqrt{D\delta}) \|y\|_2 \\
\ge& (CD/\delta - 1/\sqrt{D\delta})\sqrt{\delta} \|Ay\|_2 \ge \frac{CD}{2\sqrt{\delta}} \|Ay\|_2
\end{aligned}$$

and hence

$$|(Ax)_{p^*}| \ge \frac{CD}{2\sqrt{\delta}} \left\|(Ax)_{[D]\setminus p^*}\right\|_2.$$

As long as $C/2$ is larger than the constant in Lemma 3.1, this means two non-adaptive measurements suffice to recover $p^*$ with probability $1 - \delta$. We then output the set $S_{p^*}$, which by (5) has

$$\begin{aligned}
\left\|x_{S_{p^*}\setminus\{j\}}\right\|_2 =& \left\|y_{S_{p^*}}\right\|_2 \le \|y\|_2 / \sqrt{D\delta} \\
=& \left\|x_{[n]\setminus\{j\}}\right\|_2 / \sqrt{D\delta} = \left\|x_{[n]\setminus\{j\}}\right\|_2 / B
\end{aligned}$$

as desired. The overall failure probability is $1 - 5\delta$; rescaling $\delta$ and $C$ gives the result. □

**Lemma 3.3.** *Suppose there exists a $j$ with $|x_j| \ge C \left\|x_{[n]\setminus\{j\}}\right\|_2$ for some constant $C$. Then $O(\log\log n)$ adaptive measurements suffice to recover $j$ with probability $1/2$.*

*Proof.* Let $C'$ be the constant from Lemma 3.2. Define $B_0 = 2$ and $B_i = B_{i-1}^{3/2}$ for $i \ge 1$. Define $\delta_i = 2^{-i}/4$ for $i \ge 0$. Suppose $C \ge 16C'B_0^2/\delta_0^2$.

Define $r = O(\log\log n)$ so $B_r \ge n$. Starting with $S_0 = [n]$, our algorithm iteratively applies Lemma 3.2 with parameters $B = 4B_i$ and $\delta = \delta_i$ to $x_{S_i}$ to identify a set $S_{i+1} \subset S_i$ with $j \in S_{i+1}$, ending when $i = r$.

We prove by induction that Lemma 3.2 applies at the $i$th iteration. We chose $C$ to match the base case. For the inductive step, suppose $\left\|x_{S_i\setminus\{j\}}\right\|_2 \le |x_j| / (C' 16 \frac{B_i^2}{\delta_i^2})$. Then by Lemma 3.2,

$$\left\|x_{S_{i+1}\setminus\{j\}}\right\|_2 \le |x_j| / (C' 64 \frac{B_i^3}{\delta_i^2}) = |x_j| / (C' 16 \frac{B_{i+1}^2}{\delta_{i+1}^2})$$

```
procedure NonAdaptiveShrink(x, D)                ▷ Find smaller set S containing heavy coordinate x_j
    For i ∈ [n], s_1(i) ← {±1}, h(i) ← [D]
    For i ∈ [D], s_2(i) ← {±1}
    a ← Σ s_1(i)s_2(h(i))x_i                     ▷ Observation
    b ← Σ s_1(i)s_2(h(i))x_i(D + h(i))           ▷ Observation
    p* ← Round(b/a − D).
    return {j* | h(j*) = p*}.
end procedure
procedure AdaptiveOneSparseRec(x)                ▷ Recover heavy coordinate x_j
    S ← [n]
    B ← 2, δ ← 1/4
    while |S| > 1 do
        S ← NonAdaptiveShrink(x_S, 4B^2/δ)
        B ← B^{3/2}, δ ← δ/2.
    end while
    return S[0]
end procedure
```

**Algorithm 3.1:** Adaptive 1-sparse recovery

so the lemma applies in the next iteration as well, as desired.

After $r$ iterations, we have $S_r \leq 1 + n/B_r^2 < 2$, so we have uniquely identified $j \in S_r$. The probability that any iteration fails is at most $\sum \delta_i < 2\delta_0 = 1/2$. □

## 3.2 $k$-sparse recovery

Given a 1-sparse recovery algorithm using $m$ measurements, one can use subsampling to build a $k$-sparse recovery algorithm using $O(km)$ measurements and achieving constant success probability. Our method for doing so is quite similar to one used in [16]. The main difference is that, in order to identify one large coefficient among a subset of coordinates, we use the adaptive algorithm from the previous section as opposed to error-correcting codes.

For intuition, straightforward subsampling at rate $1/k$ will, with constant probability, recover (say) 90% of the heavy hitters using $O(km)$ measurements. This reduces the problem to $k/10$-sparse recovery: we can subsample at rate $10/k$ and recover 90% of the remainder with $O(km/10)$ measurements, and repeat $\log k$ times. The number of measurements decreases geometrically, for $O(km)$ total measurements. Naively doing this would multiply the failure probability and the approximation error by $\log k$; however, we can make the number of measurements decay less quickly than the sparsity. This allows the failure probability and approximation ratios to also decay exponentially so their total remains constant.

To determine the number of rounds, note that the initial set of $O(km)$ measurements can be done in parallel for each subsampling, so only $O(m)$ rounds are necessary to get the first 90% of heavy hitters. Repeating $\log k$ times would require $O(m \log k)$ rounds. However, we can actually make the sparsity in subsequent iterations decay super-exponentially, in fact as a power tower. This give $O(m \log^* k)$ rounds.

**Theorem 3.4.** *There exists an adaptive $(1+\epsilon)$-approximate $k$-sparse recovery scheme with $O(\frac{1}{\epsilon} k \log \frac{1}{\delta} \log\log(n\epsilon/k))$ measurements and success probability $1-\delta$. It uses $O(\log^* k \log\log(n\epsilon))$ rounds.*

To prove this, we start from the following lemma:

**Lemma 3.5.** *We can perform $O(\log\log(n/k))$ adaptive measurements and recover an $\hat{\imath}$ such that, for any $j \in H_{k,1/k}(x)$ we have $\Pr[\hat{\imath} = j] = \Omega(1/k)$.*

*Proof.* Let $S = H_k(x)$. Let $T \subset [n]$ contain each element independently with probability $p = 1/(4C^2k)$, where $C$ is the constant in Lemma 3.3. Let $j \in H_{k,1/k}(x)$. Then we have

$$\mathbb{E}[\left\|x_{T\setminus S}\right\|_2^2] = p\left\|x_{\overline{S}}\right\|_2^2$$

so

$$\left\|x_{T\setminus S}\right\|_2 \leq \sqrt{4p}\left\|x_{\overline{S}}\right\|_2 = \frac{1}{C\sqrt{k}}\left\|x_{\overline{S}}\right\|_2 \leq |x_j|/C$$

with probability at least $3/4$. Furthermore we have $\mathbb{E}[|T \setminus S|] < pn$ so $|T \setminus S| < n/k$ with probability at least $1 - 1/(4C^2) > 3/4$. By the union bound, both these events occur with probability at least $1/2$.

Independently of this, we have

$$\Pr[T \cap S = \{j\}] = p(1-p)^{k-1} > p/e$$

so all these events hold with probability at least $p/(2e)$. Assuming this,

$$\left\|x_{T\setminus\{j\}}\right\|_2 \leq |x_j|/C$$

and $|T| \leq 1 + n/k$. But then Lemma 3.3 applies, and $O(\log\log|T|) = O(\log\log(n/k))$ measurements can recover $j$ from a sketch of $x_T$ with probability $1/2$. This is independent of the previous probability, for a total success chance of $p/(4e) = \Omega(1/k)$. □

**Lemma 3.6.** *With $O(\frac{1}{\epsilon}k\log\frac{1}{f\delta}\log\log(n\epsilon/k))$ adaptive measurements, we can recover $T$ with $|T| \leq k$ and*

$$\mathrm{Err}^2(x_{\overline{T}}, fk) \leq (1+\epsilon)\,\mathrm{Err}^2(x,k)$$

*with probability at least $1-\delta$. The number of rounds required is $O(\log\log(n\epsilon/k))$.*

*Proof.* Repeat Lemma 3.5 $m = O(\frac{1}{\epsilon}k\log\frac{1}{f\delta})$ times in parallel with parameters $n$ and $k/\epsilon$ to get coordinates $T' = \{t_1, t_2, \ldots, t_m\}$. For each $j \in H_{k,\epsilon/k}(x) \subseteq H_{k/\epsilon,\epsilon/k}(x)$ and $i \in [m]$, the lemma implies $\Pr[j = t_i] \geq \epsilon/(Ck)$ for some constant $C$. Then $\Pr[j \notin T'] \leq (1-\epsilon/(Ck))^m \leq e^{-\epsilon m/(Ck)} \leq f\delta$ for appropriate $m$. Thus

$$\begin{aligned}\mathbb{E}[\left|H_{k,\epsilon/k}(x)\setminus T'\right|] \leq& f\delta\left|H_{k,\epsilon/k}(x)\right| \leq f\delta k\\ \Pr\left[\left|H_{k,\epsilon/k}(x)\setminus T'\right| \geq fk\right] \leq& \delta.\end{aligned}$$

Now, observe $x_{T'}$ directly and set $T \subseteq T'$ to be the locations of the largest $k$ values. Then, since $H_{k,\epsilon/k}(x) \subseteq H_k(x)$, $\left|H_{k,\epsilon/k}(x)\setminus T\right| = \left|H_{k,\epsilon/k}(x)\setminus T'\right| \leq fk$ with probability at least $1-\delta$.

Suppose this occurs, and let $y = x_{\overline{T}}$. Then

$$\begin{aligned}\mathrm{Err}^2(y, fk) =& \min_{|S|\leq fk}\left\|y_{\overline{S}}\right\|_2^2\\ \leq& \left\|y_{\overline{H_{k,\epsilon/k}(x)\setminus T}}\right\|_2^2\\ =& \left\|x_{\overline{H_{k,\epsilon/k}(x)}}\right\|_2^2\\ =& \left\|x_{\overline{H_k(x)}}\right\|_2^2 + \left\|x_{H_k(x)\setminus H_{k,\epsilon/k}(x)}\right\|_2^2\\ \leq& \left\|x_{\overline{H_k(x)}}\right\|_2^2 + k\left\|x_{H_k(x)\setminus H_{k,\epsilon/k}(x)}\right\|_\infty^2\\ \leq&(1+\epsilon)\left\|x_{\overline{H_k(x)}}\right\|_2^2\\ =&(1+\epsilon)\,\mathrm{Err}^2(x,k)\end{aligned}$$

as desired. □

```
procedure AdaptiveKSparseRec(x, k, ε, δ)                    ▷ Recover approximation x̂ of x
    R_0 ← [n]
    δ_0 ← δ/2, ε_0 ← ε/e, f_0 ← 1/32, k_0 ← k.
    J ← {}
    for i ← 0, ..., O(log* k) do                            ▷ While k_i ≥ 1
        for t ← 0, ..., Θ((1/ε_i) k_i log (1/δ_i)) do
            S_t ← Subsample(R_i, Θ(ε_i/k_i))
            J.add(AdaptiveOneSparseRec(x_{S_t}))
        end for
        R_{i+1} ← [n] \ J
        δ_{i+1} ← δ_i/8
        ε_{i+1} ← ε_i/2
        f_{i+1} ← 1/2^{1/(4^{i+1} f_i)}
        k_{i+1} ← k_i f_i
    end for
    x̂ ← x_J                                                 ▷ Direct observation
    return x̂
end procedure
```

**Algorithm 3.2:** Adaptive $k$-sparse recovery

**Theorem 3.7.** *We can perform $O(\frac{1}{\epsilon}k\log\frac{1}{\delta}\log\log(n\epsilon/k))$ adaptive measurements and recover a set $T$ of size at most $2k$ with*

$$\left\|x_{\overline{T}}\right\|_2 \le (1+\epsilon)\left\|x_{\overline{H_k(x)}}\right\|_2.$$

*with probability $1-\delta$. The number of rounds required is $O(\log^* k\log\log(n\epsilon))$.*

*Proof.* Define $\delta_i = \frac{\delta}{2\cdot 2^i}$ and $\epsilon_i = \frac{\epsilon}{e\cdot 2^i}$. Let $f_0 = 1/32$ and $f_i = 2^{-1/(4^i f_{i-1})}$ for $i > 0$, and define $k_i = k\prod_{j<i} f_j$. Let $R_0 = [n]$.

Let $r = O(\log^* k)$ such that $f_{r-1} < 1/k$. This is possible since $\alpha_i = 1/(4^{i+1}f_i)$ satisfies the recurrence $\alpha_0 = 8$ and $\alpha_i = 2^{\alpha_{i-1}-2i-2} > 2^{\alpha_{i-1}/2}$. Thus $\alpha_{r-1} > k$ for $r = O(\log^* k)$ and then $f_{r-1} < 1/\alpha_{r-1} < 1/k$.

For each round $i = 0,\dots,r-1$, the algorithm runs Lemma 3.6 on $x_{R_i}$ with parameters $\epsilon_i$, $k_i$, $f_i$, and $\delta_i$ to get $T_i$. It sets $R_{i+1} = R_i \setminus T_i$ and repeats. At the end, it outputs $T = \cup T_i$.

The total number of measurements is

$$\begin{aligned}
&O(\sum \frac{1}{\epsilon_i}k_i\log\frac{1}{f_i\delta_i}\log\log(n\epsilon_i/k_i))\\
\le &O(\sum \frac{2^i(k_i/k)\log(1/f_i)}{\epsilon}k(i+\log\frac{1}{\delta})\\
&\qquad\cdot\log(\log(k/k_i)+\log(n\epsilon/k)))\\
\le &O(\frac{1}{\epsilon}k\log\frac{1}{\delta}\log\log(n\epsilon/k)\\
&\qquad\cdot\sum 2^i(k_i/k)\log(1/f_i)(i+1)\log\log(k/k_i))
\end{aligned}$$

using the very crude bounds $i+\log(1/\delta) \le (i+1)\log(1/\delta)$ and $\log(a+b) \le 2\log a\log b$ for $a,b \ge e$.

But then

$$\begin{aligned}
&\sum 2^i (k_i/k)\log(1/f_i)(i+1)\log\log(k/k_i)\\
\leq &\sum 2^i (i+1) f_i \log(1/f_i)\log\log(1/f_i)\\
\leq &\sum 2^i (i+1) O(\sqrt{f_i})\\
= &O(1)
\end{aligned}$$

since $f_i < O(1/16^i)$, giving $O(\frac{1}{\epsilon}k\log\frac{1}{\delta}\log\log(n\epsilon/k)$ total measurements. The probability that any of the iterations fail is at most $\sum \delta_i < \delta$. The result has size $|T| \leq \sum k_i \leq 2k$. All that remains is the approximation ratio $\left\|x_{\overline{T}}\right\|_2 = \|x_{R_r}\|_2$.

For each $i$, we have

$$\begin{aligned}
\mathrm{Err}^2(x_{R_{i+1}}, k_{i+1}) = &\mathrm{Err}^2(x_{R_i \setminus T_i}, f_i k_i)\\
\leq &(1+\epsilon_i)\,\mathrm{Err}^2(x_{R_i}, k_i).
\end{aligned}$$

Furthermore, $k_r < k f_{r-1} < 1$. Hence

$$\begin{aligned}
\|x_{R_r}\|_2^2 = \mathrm{Err}^2(x_{R_r}, k_r) &\leq \left(\prod_{i=0}^{r-1}(1+\epsilon_i)\right)\mathrm{Err}^2(x_{R_0}, k_0)\\
&= \left(\prod_{i=0}^{r-1}(1+\epsilon_i)\right)\mathrm{Err}^2(x, k)
\end{aligned}$$

But $\prod_{i=0}^{r-1}(1+\epsilon_i) < e^{\sum \epsilon_i} < e$, so

$$\prod_{i=0}^{r-1}(1+\epsilon_i) < 1 + \sum e\epsilon_i \leq 1 + 2\epsilon$$

and hence

$$\left\|x_{\overline{T}}\right\|_2 = \|x_{R_r}\|_2 \leq (1+\epsilon)\left\|x_{\overline{H_k(x)}}\right\|_2$$

as desired. □

Once we find the support $T$, we can observe $x_T$ directly with $O(k)$ measurements to get a $(1+\epsilon)$-approximate $k$-sparse recovery scheme, proving Theorem 3.4

# 4 Two-round adaptivity

The algorithms in this section are invariant under permutation. Therefore, for simplicity of notation, the analysis assumes our vectors $x$ is sorted: $|x_1| \geq \ldots \geq |x_n| = 0$.

We are given a 1-round $k$-sparse recovery algorithm for $n$-dimensional vectors $x$ using $m(k, \epsilon, n, \delta)$ measurements with the guarantee that its output $\hat{x}$ satisfies $\|\hat{x} - x\|_p \leq (1+\epsilon)\cdot\|x_{\overline{[k]}}\|_p$ for a $p \in \{1, 2\}$ with probability at least $1-\delta$. Moreover, suppose its output $\hat{x}$ has support on a set of size $s(k, \epsilon, n, \delta)$. We show the following black box two-round transformation.

**Theorem 4.1.** *Assume $s(k, \epsilon, n, \delta) = O(k)$. Then there is a 2-round sparse recovery algorithm for $n$-dimensional vectors $x$, which, in the first round uses $m(k, \epsilon/5, \mathrm{poly}(k/\epsilon), 1/100)$ measurements and in the second uses $O(k \cdot m(1, 1, n, \Theta(1/k)))$ measurements. It succeeds with constant probability.*

**Corollary 4.2.** *For $p = 2$, there is a 2-round sparse recovery algorithm for $n$-dimensional vectors $x$ such that the total number of measurements is $O(\frac{1}{\epsilon}k\log(k/\epsilon) + k\log(n/k))$.*

*Proof of Corollary 4.2.* In the first round it suffices to use CountSketch with $s(k, \epsilon, n, 1/100) = 2k$, which holds for any $\epsilon > 0$ [28]. We also have that $m(k, \epsilon/5, \mathrm{poly}(k/\epsilon), 1/100) = O(\frac{1}{\epsilon}k\log(k/\epsilon))$. Using [5, 7, 14], in the second round we can set $m(1, 1, n, \Theta(1/k)) = O(\log n)$. The bound follows by observing that $\frac{1}{\epsilon}k\log(k/\epsilon) + k\log(n) = O(\frac{1}{\epsilon}k\log(k/\epsilon) + k\log(n/k))$. □

*Proof of Theorem 4.1.* In the first round we perform a dimensionality reduction of the $n$-dimensional input vector $x$ to a $\mathrm{poly}(k/\epsilon)$-dimensional input vector $y$. We then apply the black box sparse recovery algorithm on the reduced vector $y$, obtaining a list of $s(k, \epsilon/5, \mathrm{poly}(k/\epsilon), 1/100)$ coordinates, and show for each coordinate in the list, if we choose the largest preimage for it in $x$, then this list of coordinates can be used to provide a $1 + \epsilon$ approximation for $x$. In the second round we then identify which heavy coordinates in $x$ map to those found in the first round, for which it suffices to invoke the black box algorithm with only a constant approximation. We place the estimated values of the heavy coordinates obtained in the first pass in the locations of the heavy coordinates obtained in the second pass.

Let $N = \mathrm{poly}(k/\epsilon)$ be determined below. Let $h : [n] \to [N]$ and $\sigma : [n] \to \{-1, 1\}$ be $\Theta(\log N)$-wise independent random functions. Define the vector $y$ by $y_i = \sum_{j \mid h(j)=i} \sigma(j)x_j$. Let $Y(i)$ be the vector $x$ restricted to coordinates $j \in [n]$ for which $h(j) = i$. Because the algorithm is invariant under permutation of coordinates of $y$, we may assume for simplicity of notation that $y$ is sorted: $|y_1| \geq \ldots \geq |y_N| = 0$.

We note that such a dimensionality reduction is often used in the streaming literature. For example, the sketch of [30] for $\ell_2$-norm estimation utilizes such a mapping. A "multishot" version (that uses several functions $h$) has been used before in the context of sparse recovery [5, 7] (see [14] for an overview). Here, however, we need to analyze a "single-shot" version.

Let $p \in \{1, 2\}$, and consider sparse recovery with the $\ell_p/\ell_p$ guarantee. We can assume that $\|x\|_p = 1$. We need two facts concerning concentration of measure.

**Fact 4.3.** *(see, e.g., Lemma 2 of [23]) Let $X_1, \ldots, X_n$ be such that $X_i$ has expectation $\mu_i$ and variance $v_i^2$, and $X_i \leq K$ almost surely. Then if the $X_i$ are $\ell$-wise independent for an even integer $\ell \geq 2$,*

$$\Pr\left[\left|\sum_{i=1}^{n} X_i - \mu\right| \geq \lambda\right] \leq 2^{O(\ell)}\left(\left(v\sqrt{\ell}/\lambda\right)^{\ell} + (K\ell/\lambda)^{\ell}\right)$$

*where $\mu = \sum_i \mu_i$ and $v^2 = \sum_i v_i^2$.*

**Fact 4.4.** *(Khintchine inequality) ([18]) For $t \geq 2$, a vector $z$ and a $t$-wise independent random sign vector $\sigma$ of the same number of dimensions, $\mathbf{E}[|\langle z, \sigma\rangle|^t] \leq \|z\|_2^t(\sqrt{t})^t$.*

We start with a probabilistic lemma. Let $Z(j)$ denote the vector $Y(j)$ with the coordinate $m(j)$ of largest magnitude removed.

**Lemma 4.5.** *Let $r = O\left(\|x_{\overline{[k]}}\|_p \cdot \frac{\log N}{N^{1/6}}\right)$ and $N$ be sufficiently large. Then with probability $\geq 99/100$,*

1. *$\forall j \in [N]$, $\|Z(j)\|_p \leq r$.*
2. *$\forall i \in [N^{1/3}]$, $|\sigma(i) \cdot y_{h(i)} - x_i| \leq r$,*
3. *$\|y_{\overline{[k]}}\|_p \leq (1 + O(1/\sqrt{N})) \cdot \|x_{\overline{[k]}}\|_p + O(kr)$,*
4. *$\forall j \in [N]$, if $h^{-1}(j) \cap [N^{1/3}] = \emptyset$, then $|y_j| \leq r$,*
5. *$\forall j \in [N]$, $\|Y(j)\|_0 = O(n/N + \log N)$.*

*Proof.* We start by defining events $\mathcal{E}$, $\mathcal{F}$ and $\mathcal{G}$ that will be helpful in the analysis, and showing that all of them are satisfied simultaneously with constant probability.

**Event $\mathcal{E}$:** Let $\mathcal{E}$ be the event that $h(1), h(2), \ldots, h(N^{1/3})$ are distinct. Then $\Pr_h[\mathcal{E}] \geq 1 - 1/N^{1/3}$.

**Event $\mathcal{F}$:** Fix $i \in [N]$. Let $Z'$ denote the vector $Y(h(i))$ with the coordinate $i$ removed. Applying Fact 4.4 with $t = \Theta(\log N)$,

$$\begin{aligned}
& \Pr_\sigma[|\sigma(i)y_{h(i)} - x_i| \geq 2\sqrt{t} \cdot \|Z(h(i))\|_2] \\
\leq\ & \Pr_\sigma[|\sigma(i)y_{h(i)} - x_i| \geq 2\sqrt{t} \cdot \left\|Z'\right\|_2] \\
\leq\ & \Pr_\sigma[|\sigma(i)y_{h(i)} - x_i|^t \geq 2^t(\sqrt{t})^t \cdot \|Z'\|_2^t] \\
\leq\ & \Pr_\sigma[|\sigma(i)y_{h(i)} - x_i|^t \geq 2^t\, \mathbb{E}[\left|\langle \sigma, Z'\rangle\right|^t]] \\
=\ & \Pr_\sigma[|\sigma(i)y_{h(i)} - x_i|^t \geq 2^t\, \mathbb{E}[|\sigma(i)y_{h(i)} - x_i|^t] \leq 1/N^{4/3}.
\end{aligned}$$

Let $\mathcal{F}$ be the event that for all $i \in [N]$, $|\sigma(i)y_{h(i)} - x_i| \leq 2\sqrt{t} \cdot \|Z(h(i))\|_2$, so $\Pr_\sigma[\mathcal{F}] \geq 1 - 1/N$.

**Event $\mathcal{G}$:** Fix $j \in [N]$ and for each $i \in \{N^{1/3}+1, \ldots, n\}$, let $X_i = |x_i|^p \mathbf{1}_{h(i)=j}$ (i.e., $X_i = |x_i|^p$ if $h(i) = j$). We apply Lemma 4.3 to the $X_i$. In the notation of that lemma, $\mu_i = |x_i|^p/N$ and $v_i^2 \leq |x_i|^{2p}/N$, and so $\mu = \|x_{\overline{[N^{1/3}]}}\|_p^p/N$ and $v^2 \leq \|x_{\overline{[N^{1/3}]}}\|_{2p}^{2p}/N$. Also, $K = |x_{N^{1/3}+1}|^p$. Function $h$ is $\Theta(\log N)$-wise independent, so by Fact 4.3,

$$\begin{aligned}
& \Pr\left[\left|\sum_i X_i - \frac{\|x_{\overline{[N^{1/3}]}}\|_p^p}{N}\right| \geq \lambda\right] \\
\leq & 2^{O(\ell)}\left(\left(\|x_{\overline{[N^{1/3}]}}\|_{2p}^p \sqrt{\ell}/(\lambda\sqrt{N})\right)^\ell + \left(|x_{N^{1/3}+1}|^p \ell/\lambda\right)^\ell\right)
\end{aligned}$$

for any $\lambda > 0$ and an $\ell = \Theta(\log N)$. For $\ell$ large enough, there is a

$$\lambda = \Theta(\|x_{\overline{[N^{1/3}]}}\|_{2p}^p \sqrt{(\log N)/N} + |x_{N^{1/3}+1}|^p \cdot \log N)$$

for which this probability is $\leq N^{-2}$. Let $\mathcal{G}$ be the event that for all $j \in [N]$, $\|Z(j)\|_p^p \leq C\left(\frac{\|x_{\overline{[N^{1/3}]}}\|_p^p}{N} + \lambda\right)$ for some universal constant $C > 0$. Then $\Pr[\mathcal{G} \mid \mathcal{E}] \geq 1 - 1/N$.

By a union bound, $\Pr[\mathcal{E} \wedge \mathcal{F} \wedge \mathcal{G}] \geq 999/1000$ for $N$ sufficiently large.

We know proceed to proving the five conditions in the lemma statement. In the analysis we assume that the event $\mathcal{E} \wedge \mathcal{F} \wedge \mathcal{G}$ holds (i.e., we condition on that event).

**First Condition:** This condition follows from the occurrence of $\mathcal{G}$, and using that $\|x_{\overline{[N^{1/3}]}}\|_{2p} \leq \|x_{\overline{[N^{1/3}]}}\|_p$, and $\|x_{\overline{[N^{1/3}]}}\|_p \leq \|x_{\overline{[k]}}\|_p$, as well as $(N^{1/3} - k + 1)|x_{N^{1/3}+1}|^p \leq \|x_{\overline{[k]}}\|_p^p$. One just needs to make these substitutions into the variable $\lambda$ defining $\mathcal{G}$ and show the value $r$ serves as an upper bound (in fact, there is a lot of room to spare, e.g., $r/\log N$ is also an upper bound).

**Second Condition:** This condition follows from the joint occurrence of $\mathcal{E}$, $\mathcal{F}$, and $\mathcal{G}$.

**Third Condition:** For the third condition, let $y'$ denote the restriction of $y$ to coordinates in the set $[N] \setminus \{h(1), h(2), ..., h(k)\}$. For $p = 1$ and for any choice of $h$ and $\sigma$, $\|y'\|_1 \leq \|x_{\overline{[k]}}\|_1$. For $p = 2$, the vector $y$ is the sketch of [30] for $\ell_2$-estimation. By their analysis, with probability $\geq 999/1000$, $\|y'\|_2^2 \leq (1 + O(1/\sqrt{N}))\|x'\|_2^2$, where $x'$ is the vector whose support is $[n] \setminus \cup_{i=1}^k h^{-1}(i) \subseteq [n] \setminus [k]$. We assume this occurs and add $1/1000$ to our error probability. Hence, $\|y'\|_2^2 \leq (1 + O(1/\sqrt{N}))\|x_{\overline{[k]}}\|_2^2$.

We relate $\|y'\|_p^p$ to $\|y_{\overline{[k]}}\|_p^p$. Consider any $j = h(i)$ for an $i \in [k]$ for which $j$ is not among the top $k$ coordinates of $y$. Call such a $j$ *lost*. By the first condition of the lemma, $|\sigma(i)y_j - x_i| \leq r$. Since $j$ is not among the top $k$ coordinates of $y$, there is a coordinate $j'$ among the top $k$ coordinates of $y$ for which $j' \notin h([k])$ and $|y_{j'}| \geq |y_j| \geq |x_i| - r$. We call such a $j'$ a *substitute*. We can bijectively map substitutes to lost coordinates. It follows that $\|y_{\overline{[k]}}\|_p^p \leq \|y'\|_p^p + O(kr) \leq (1 + O(1/\sqrt{N}))\|x_{\overline{[k]}}\|_p^p + O(kr)$.

**Fourth Condition:** This follows from the joint occurrence of $\mathcal{E}, \mathcal{F}$, and $\mathcal{G}$, and using that $|x_{m(j)}|^p \leq \|x_{\overline{[k]}}\|_p^p/(N^{1/3} - k + 1)$ since $m(j) \notin [N^{1/3}]$.

**Fifth Condition:** For the fifth condition, fix $j \in [N]$. We apply Fact 4.3 where the $X_i$ are indicator variables for the event $h(i) = j$. Then $\mathbf{E}[X_i] = 1/N$ and $\mathbf{Var}[X_i] < 1/N$. In the notation of Fact 4.3, $\mu = n/N$, $v^2 < n/N$, and $K = 1$. Setting $\ell = \Theta(\log N)$ and $\lambda = \Theta(\log N + \sqrt{(n \log N)/N})$, we have by a union bound that for all $j \in [N]$, $\|Y(j)\|_0 \leq \frac{n}{N} + \Theta(\log N + \sqrt{(n \log N)/N}) = O(n/N + \log N)$, with probability at least $1 - 1/N$.

By a union bound, all events jointly occur with probability at least $99/100$, which completes the proof. □

**Event $\mathcal{H}$:** Let $\mathcal{H}$ be the event that the algorithm returns a vector $\hat{y}$ with $\|\hat{y} - y\|_p \leq (1 + \epsilon/5)\|y_{\overline{[k]}}\|_p$. Then $\Pr[\mathcal{H}] \geq 99/100$. Let $S$ be the support of $\hat{y}$, so $|S| = s(k, \epsilon/5, N, 1/100)$. We condition on $\mathcal{H}$.

In the second round we run the algorithm on $Y(j)$ for each $j \in S$, each using $m(1, 1, \|Y(j)\|_0, \Theta(1/k)))$-measurements. Using the fifth condition of Lemma 4.5, we have that $\|Y(j)\|_0 = O(\epsilon n/k + \log(k)/\epsilon)$ for $N = \mathrm{poly}(k/\epsilon)$ sufficiently large.

For each invocation on a vector $Y(j)$ corresponding to a $j \in S$, the algorithm takes the largest (in magnitude) coordinate $\mathrm{HH}(j)$ in the output vector, breaking ties arbitrarily. We output the vector $\hat{x}$ with support equal to $T = \{\mathrm{HH}(j) \mid j \in S\}$. We assign the value $\sigma(x_j)\hat{y}_j$ to $\mathrm{HH}(j)$. We have

$$\begin{aligned} \|x - \hat{x}\|_p^p =& \|(x - \hat{x})_T\|_p^p + \|(x - \hat{x})_{[n]\setminus T}\|_p^p \\ =& \|(x - \hat{x})_T\|_p^p + \|x_{[n]\setminus T}\|_p^p. \end{aligned} \tag{8}$$

The rest of the analysis is devoted to bounding the RHS of equation 8.

**Lemma 4.6.** *For $N = \mathrm{poly}(k/\epsilon)$ sufficiently large, conditioned on the events of Lemma 4.5 and $\mathcal{H}$,*

$$\|x_{[n]\setminus T}\|_p^p \leq (1 + \epsilon/3)\|x_{\overline{[k]}}\|_p^p.$$

*Proof.* If $[k] \setminus T = \emptyset$, the lemma follows by definition. Otherwise, if $i \in ([k] \setminus T)$, then $i \in [k]$, and so by the second condition of Lemma 4.5, $|x_i| \leq |y_{h(i)}| + r$. We also use the third condition of Lemma 4.5 to

obtain $\|y_{\overline{[k]}}\|_p \leq (1 + O(1/\sqrt{N})) \cdot \|x_{\overline{[k]}}\|_p + O(kr)$. By the triangle inequality,

$$\begin{aligned}\left(\sum_{i\in[k]\setminus T} |x_i|^p\right)^{1/p} &\leq k^{1/p} r + \left(\sum_{i \,\in\, [k]\setminus T} |y_{h(i)}|^p\right)^{1/p} \\ &\leq k^{1/p} r + \left(\sum_{i \,\in\, [N]\setminus S} |y_i|^p\right)^{1/p} \\ &\leq k^{1/p} r + (1+\epsilon/5) \cdot \|y_{\overline{[k]}}\|_p.\end{aligned}$$

The lemma follows using that $r = O(\|x_{\overline{[k]}}\|_2 \cdot (\log N)/N^{1/6})$ and $N = \mathrm{poly}(k/\epsilon)$ is sufficiently large. □

We bound $\|(x - \hat{x})_T\|_p^p$ using Lemma 4.5, $|S| \leq \mathrm{poly}(k/\epsilon)$, and that $N = \mathrm{poly}(k/\epsilon)$ is sufficiently large.

$$\begin{aligned}&\|(x-\hat{x})_T\|_p \\ \leq& \left(\sum_{j\in S} |x_{HH(j)} - \sigma(HH(j)) \cdot \hat{y}_j|^p\right)^{1/p} \\ \leq& \left(\sum_{j\in S} (|y_j - \hat{y}_j| + |\sigma(HH(j)) \cdot x_{HH(j)} - y_j|)^p\right)^{1/p} \\ \leq& \left(\sum_{j\in S} |y_j - \hat{y}_j|^p\right)^{1/p} \\ &+ \left(\sum_{j\in S} |\sigma(HH(j)) \cdot x_{HH(j)} - y_j|^p\right)^{1/p} \\ \leq& (1+\epsilon/5)\|y_{\overline{[k]}}\|_p \\ &+ \left(\sum_{j\in S} |\sigma(HH(j)) \cdot x_{HH(j)} - y_j|^p\right)^{1/p} \\ \leq& (1+\epsilon/5)(1+O(1/\sqrt{N}))\|x_{\overline{[k]}}\|_p + O(kr) \\ &+ \left(\sum_{j\in S} |\sigma(HH((j)) \cdot x_{HH(j)} - y_j|^p\right)^{1/p} \\ \leq& (1+\epsilon/4)\|x_{\overline{[k]}}\|_p \\ &+ \left(\sum_{j\in S} |\sigma(HH(j)) \cdot x_{HH(j)} - y_j|^p\right)^{1/p}\end{aligned}$$

**Event $\mathcal{I}$:** We condition on the event $\mathcal{I}$ that all second round invocations succeed. Note that $\Pr[\mathcal{I}] \geq 99/100$.

We need the following lemma concerning 1-sparse recovery algorithms.

**Lemma 4.7.** *Let $w$ be a vector of real numbers. Suppose $|w_1|^p > \frac{9}{10} \cdot \|w\|_p^p$. Then for any vector $\hat{w}$ for which $\|w - \hat{w}\|_p^p \leq 2 \cdot \|w_{\overline{[1]}}\|_p^p$, we have $|\hat{w}_1|^p > \frac{3}{5} \cdot \|w\|_p^p$. Moreover, for all $j > 1$, $|\hat{w}_j|^p < \frac{3}{5} \cdot \|w\|_p^p$.*

*Proof.* $\|w - \hat{w}\|_p^p \geq |w_1 - \hat{w}_1|^p$, so if $|\hat{w}_1|^p < \frac{3}{5} \cdot \|w\|_p^p$, then $\|w - \hat{w}\|_p^p > \left(\frac{9}{10} - \frac{3}{5}\right) \|w\|_p^p = \frac{3}{10} \cdot \|w\|_p^p$. On the other hand, $\|w_{\overline{[1]}}\|_p^p < \frac{1}{10} \cdot \|w\|_p^p$. This contradicts that $\|w - \hat{w}\|_p^p \leq 2 \cdot \|w_{\overline{[1]}}\|_p^p$. For the second part, for $j > 1$ we have $|w_j|^p < \frac{1}{10} \cdot \|w\|_p^p$. Now, $\|w - \hat{w}\|_p^p \geq |w_j - \hat{w}_j|^p$, so if $|\hat{w}_j|^p \geq \frac{3}{5} \cdot \|w\|_p^p$, then $\|w - \hat{w}\|_p^p > \left(\frac{3}{5} - \frac{1}{10}\right) \|w\|_p^p = \frac{1}{2} \cdot \|w\|_p^p$. But since $\|w_{\overline{[1]}}\|_p^p < \frac{1}{10} \cdot \|w\|_p^p$, this contradicts that $\|w - \hat{w}\|_p^p \leq 2 \cdot \|w_{\overline{[1]}}\|_p^p$. □

It remains to bound $\sum_{j \in S} |\sigma(HH(j)) \cdot x_{HH(j)} - y_j|^p$. We show for every $j \in S$, $|\sigma(HH(j)) \cdot x_{HH(j)} - y_j|^p$ is small.

Recall that $m(j)$ is the coordinate of $Y(j)$ with the largest magnitude. There are two cases.

**Case 1: $m(j) \notin [N^{1/3}]$.** In this case observe that $HH(j) \notin [N^{1/3}]$ either, and $h^{-1}(j) \cap [N^{1/3}] = \emptyset$. It follows by the fourth condition of Lemma 4.5 that $|y_j| \leq r$. Notice that $|x_{HH(j)}|^p \leq |x_{m(j)}|^p \leq \frac{\|x_{\overline{[k]}}\|_p^p}{N^{1/3} - k}$. Bounding $|\sigma(HH(j)) \cdot x_{HH(j)} - y_j|$ by $|x_{HH(j)}| + |y_j|$, it follows for $N = \text{poly}(k/\epsilon)$ large enough that $|\sigma(HH(j)) \cdot x_{HH(j)} - y_j|^p \leq \epsilon/4 \cdot \|x_{\overline{[k]}}\|_p / |S|)$.

**Case 2: $m(j) \in [N^{1/3}]$.** If $HH(j) = m(j)$, then $|\sigma(HH(j)) \cdot x_{HH(j)} - y_j| \leq r$ by the second condition of Lemma 4.5, and therefore

$$|\sigma(HH(j)) \cdot x_{HH(j)} - y_j|^p \leq r^p \leq \epsilon/4 \cdot \|x_{\overline{[k]}}\|_p / |S|$$

for $N = \text{poly}(k/\epsilon)$ large enough.

Otherwise, $HH(j) \neq m(j)$. From condition 2 of Lemma 4.5 and $m(j) \in [N^{1/3}]$, it follows that

$$\begin{aligned}
&|\sigma(HH(j)))x_{HH(j)} - y_j| \\
\leq &|\sigma(HH(j))x_{HH(j)} - \sigma(m(j))x_{m(j)}| + |\sigma(m(j))x_{m(j)} - y_j| \\
\leq &|x_{HH(j)}| + |x_{m(j)}| + r
\end{aligned}$$

Notice that $|x_{HH(j)}| + |x_{m(j)}| \leq 2|x_{m(j)}|$ since $m(j)$ is the coordinate of largest magnitude. Now, conditioned on $\mathcal{I}$, Lemma 4.7 implies that $|x_{m(j)}|^p \leq \frac{9}{10} \cdot \|Y(j)\|_p^p$, or equivalently, $|x_{m(j)}| \leq 10^{1/p} \cdot \|Z(j)\|_p$. Finally, by the first condition of Lemma 4.5, we have $\|Z(j)\|_p = O(r)$, and so $|\sigma(HH(j))x_{HH(j)} - y_j|^p = O(r^p)$, which as argued above, is small enough for $N = \text{poly}(k/\epsilon)$ sufficiently large.

The proof of our theorem follows by a union bound over the events that we defined. □

## Acknowledgements

This material is based upon work supported by the Space and Naval Warfare Systems Center Pacific under Contract No. N66001-11-C-4092, David and Lucille Packard Fellowship, MADALGO (Center for Massive Data Algorithmics, funded by the Danish National Research Association) and NSF grant CCF-1012042. E. Price is supported in part by an NSF Graduate Research Fellowship.

## References


[1] A. Aldroubi, H. Wang, and K. Zarringhalam, "Sequential adaptive compressed sampling via huffman codes," *Preprint*, 2008.

[2] A. Bruex, A. Gilbert, R. Kainkaryam, J. Schiefelbein, and P. Woolf, "Poolmc: Smart pooling of mRNA samples in microarray experiments," *BMC Bioinformatics*, 2010.

[3] E. J. Candès, J. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Comm. Pure Appl. Math.*, vol. 59, no. 8, pp. 1208–1223, 2006.

[4] R. Castro, J. Haupt, R. Nowak, and G. Raz, "Finding needles in noisy haystacks," *Proc. IEEE Conf. Acoustics, Speech, and Signal Proc.*, p. 51335136, 2008.

[5] M. Charikar, K. Chen, and M. Farach-Colton, "Finding frequent items in data streams," *ICALP*, 2002.

[6] G. Cormode and S. Muthukrishnan, "Improved data stream summaries: The count-min sketch and its applications," *LATIN*, 2004.

[7] ——, "Combinatorial algorithms for Compressed Sensing," in *Proc. 40th Ann. Conf. Information Sciences and Systems*, Princeton, Mar. 2006.

[8] Defense Sciences Office, "Knowledge enhanced compressive measurement," *Broad Agency Announcement*, vol. DARPA-BAA-10-38, 2010.

[9] K. Do Ba, P. Indyk, E. Price, and D. Woodruff, "Lower bounds for sparse recovery," *SODA*, 2010.

[10] D. L. Donoho, "Compressed Sensing," *IEEE Trans. Info. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

[11] M. Duarte, M. Davenport, D. Takhar, J. Laska, T. Sun, K. Kelly, and R. Baraniuk, "Single-pixel imaging via compressive sampling," *IEEE Signal Processing Magazine*, 2008.

[12] S. Foucart, A. Pajor, H. Rauhut, and T. Ullrich, "The gelfand widths of lp-balls for $0 < p \leq 1$," *J. Complexity*, 2010.

[13] A. Y. Garnaev and E. D. Gluskin, "On widths of the euclidean ball," *Sov. Math., Dokl.*, p. 200204, 1984.

[14] A. Gilbert and P. Indyk, "Sparse recovery using sparse matrices," *Proceedings of IEEE*, 2010.

[15] A. C. Gilbert, S. Guha, P. Indyk, Y. Kotidis, S. Muthukrishnan, and M. J. Strauss, "Fast, small-space algorithms for approximate histogram maintenance," in *ACM Symposium on Theoretical Computer Science*, 2002.

[16] A. C. Gilbert, Y. Li, E. Porat, and M. J. Strauss, "Approximate sparse recovery: optimizing time and measurements," in *STOC*, 2010, pp. 475–484.

[17] E. D. Gluskin, "Norms of random matrices and widths of finite-dimensional sets," *Math. USSR-Sb.*, vol. 48, p. 173182, 1984.

[18] U. Haagerup, "The best constants in the Khintchine inequality," *Studia Math.*, vol. 70, no. 3, pp. 231–283, 1982.

[19] J. Haupt, R. Baraniuk, R. Castro, and R. Nowak, "Compressive distilled sensing," *Asilomar*, 2009.

[20] J. Haupt, R. Castro, and R. Nowak, "Adaptive sensing for sparse signal recovery," *Proc. IEEE 13th Digital Sig. Proc./5th Sig. Proc. Education Workshop*, p. 702707, 2009.

[21] P. Indyk, "Sketching, streaming and sublinear-space algorithms," *Graduate course notes, available at http://stellar.mit.edu/S/course/6/fa07/6.895/*, 2007.

[22] S. Ji, Y. Xue, and L. Carin, "Bayesian compressive sensing," *IEEE Trans. Signal Processing*, vol. 56(6), p. 23462356, 2008.

[23] D. M. Kane, J. Nelson, E. Porat, and D. P. Woodruff, "Fast moment estimation in data streams in optimal space," *CoRR*, vol. abs/1007.4191, 2010.

[24] B. S. Kashin, "Diameters of some finite-dimensional sets and classes of smooth functions." *Math. USSR, Izv.,*, vol. 11, p. 317333, 1977.

[25] D. M. Malioutov, S. Sanghavi, and A. S. Willsky, "Compressed sensing with sequential observations," *ICASSP*, 2008.

[26] A. McGregor, "Graph mining on streams," *Encyclopedia of Database Systems*, p. 12711275, 2009.

[27] S. Muthukrishnan, "Data streams: Algorithms and applications)," *Foundations and Trends in Theoretical Computer Science*, 2005.

[28] E. Price and D. Woodruff, "(1+eps)-approximate sparse recovery," *FOCS*, 2011.

[29] N. Shental, A. Amir, and O. Zuk, "Identification of rare alleles and their carriers using compressed se(que)nsing," *Nucleic Acids Research*, vol. 38(19), pp. 1–22, 2010.

[30] M. Thorup and Y. Zhang, "Tabulation based 4-universal hashing with applications to second moment estimation," in *Proceedings of the 15th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2004, pp. 615–624.