On Deterministic Sketching and Streaming for Sparse Recovery and Norm Estimation

Jelani Nelson[a], Huy L. Nguyễn[b], David P. Woodruff[c]

[a]*minilek@seas.harvard.edu*
[b]*hlnguyen@princeton.edu*
[c]*dpwoodru@us.ibm.com*

Abstract

We study classic streaming and sparse recovery problems using *deterministic* linear sketches, including ℓ_1/ℓ_1 and ℓ_∞/ℓ_1 sparse recovery problems (the latter also being known as ℓ_1-heavy hitters), norm estimation, and approximate inner product. We focus on devising a fixed matrix $A \in \mathbb{R}^{m\times n}$ and a deterministic recovery/estimation procedure which work for all possible input vectors simultaneously. Our results improve upon existing work, the following being our main contributions:

- A proof that ℓ_∞/ℓ_1 sparse recovery and inner product estimation are equivalent, and that incoherent matrices can be used to solve both problems. Our upper bound for the number of measurements is $m = O(\varepsilon^{-2}\min\{\log n, (\log n/\log(1/\varepsilon))^2\})$, which holds for any $0 < \varepsilon < 1/2$. We can also obtain fast sketching and recovery algorithms by making use of the Fast Johnson-Lindenstrauss transform. Both our running times and number of measurements improve upon previous work. We can also obtain better error guarantees than previous work in terms of a smaller tail of the input vector.

- A new lower bound for the number of linear measurements required to solve ℓ_1/ℓ_1 sparse recovery. We show $\Omega(k/\varepsilon^2 + k\log(n/k)/\varepsilon)$ measurements are required to recover an x' with $\|x - x'\|_1 \le (1+\varepsilon)\|x_{tail(k)}\|_1$, where $x_{tail(k)}$ is x projected onto all but its largest k coordinates in magnitude.

- A tight bound of $m = \Theta(\varepsilon^{-2}\log(\varepsilon^2 n))$ on the number of measurements required to solve deterministic norm estimation, i.e., to recover $\|x\|_2 \pm \varepsilon\|x\|_1$.

For all the problems we study, tight bounds are already known for the randomized complexity from previous work, except in the case of ℓ_1/ℓ_1 sparse recovery, where a nearly tight bound is known. Our work thus aims to study the deterministic complexities of these problems.

Keywords: streaming algorithms, sparse recovery, heavy hitters, norm estimation
15A03 MSC: ,
68Q25 MSC:

---

1. Introduction

In this work we provide new results for the point query problem as well as several other related problems: approximate inner product, ℓ_1/ℓ_1 sparse recovery, and deterministic norm estimation. For many of these problems efficient randomized sketching and streaming algorithms exist, and thus we are interested in understanding the *deterministic* complexities of these problems.

1.1. Applications

Here we give a motivating application of the point query problem; for a formal definition of the problem, see below. Consider k servers $S^1, \ldots, S^k$, each holding a database $D^1, \ldots, D^k$, respectively. The servers want to compute statistics of the union D of the k databases. For instance, the servers may want to know the frequency of a record or attribute-pair in D. It may be too expensive for the servers to communicate their individual databases to a centralized server, or to compute the frequency exactly. Hence, the servers wish to communicate a short summary or "sketch" of their databases to a centralized server, who can then combine the sketches to answer frequency queries about D.

We model the databases as vectors $x^i \in \mathbb{R}^n$. To compute a sketch of x^i, we compute Ax^i for a matrix A with m rows and n columns. Importantly, $m \ll n$, and so Ax^i is much easier to communicate than x^i. The servers compute $Ax^1, \ldots, Ax^k$, respectively, and transmit these to a centralized server. Since A is a linear map, the centralized server can compute Ax for $x = c_1x^1 + \ldots c_kx^k$ for any real numbers $c_1, \ldots, c_k$. Notice that the c_i are allowed to be both positive and negative, which is crucial for estimating the frequency of record or attribute-pairs in the difference of two datasets, which allows for tracking which items have experienced a sudden growth or decline in frequency. This is useful in network anomaly detection [1, 2, 3, 4, 5], and also for transactional data [6]. It is also useful for maintaining the set of frequent items over a changing database relation [6].

Associated with A is an output algorithm Out which given Ax, outputs a vector x' for which $\|x' - x\|_\infty \le \varepsilon \|x_{tail(k)}\|_1$ for some number k, where $x_{tail(k)}$ denotes the vector x with the top k entries in absolute value replaced with 0 (the other entries being unchanged). Thus x' approximates x well on every coordinate. We call the pair (A, Out) a solution to the point query problem. Given such a matrix A and an output algorithm Out, the centralized server can obtain an approximation to the value of every entry in x, which depending on the application, could be the frequency of an attribute-pair. It can also, e.g., extract the maximum frequencies of x, which are useful for obtaining the most frequent items. The centralized server obtains an entire histogram of values of coordinates in x, which is a useful low-memory representation of x. Notice that the communication is mk words, as opposed to nk if the servers were to transmit $x^1, \ldots, x^n$.

Our correctness guarantees hold for all input vectors simultaneously using one fixed (A, Out) pair, and thus it is stronger and should be contrasted with the guarantee that the algorithm succeeds given Ax with high probability for some fixed input x. For example, for the point query problem, the latter guarantee is achieved by the CountMin sketch [7] or CountSketch [8]. One of the reasons the randomized guarantee is less useful is because of *adaptive* queries. That is, suppose the centralized server computes x' and transmits information about x' to $S^1, \ldots, S^k$. Since x' could depend on A, if the servers were to then use the same matrix A to compute sketches $Ay^1, \ldots, Ay^k$ for databases $y^1, \ldots, y^k$ which depend on x', then A need not succeed, since it is not guaranteed to be correct with high probability for inputs y^i which depend on A.

1.2. Notation and Problem Definitions

Throughout this work $[n]$ denotes $\{1, \ldots, n\}$. For q a prime power, $\mathbb{F}_q$ denotes the finite field of size q. For $x \in \mathbb{R}^n$ and $S \subseteq [n]$, x_S denotes the vector with $(x_S)_i = x_i$ for $i \in S$, and $(x_S)_i = 0$ for $i \notin S$. The notation x_{-i} is shorthand for $x_{[n]\setminus\{i\}}$. For a matrix $A \in \mathbb{R}^{m \times n}$ and integer $i \in [n]$, A_i denotes the ith column of A. For matrices A and vectors x, A^T and x^T denote their transposes. For $x \in \mathbb{R}^n$ and integer $k \le n$, we let $head(x, k) \subseteq [n]$ denote the set of k largest coordinates in x in absolute value, and $tail(x, k) = [n] \backslash head(x, k)$. We often use $x_{head(k)}$ to denote $x_{head(x,k)}$, and similarly for the tail. For real numbers $a, b, \varepsilon \geq 0$, we use the notation $a = (1 \pm \varepsilon)b$ to convey that $a \in [(1-\varepsilon)b, (1+\varepsilon)b]$. A collection of vectors $\{C_1, \ldots, C_n\} \in [q]^t$ is called a *code* with *alphabet size* q and *block length* t, and we define $\Delta(C_i, C_j) = |\{k : (C_i)_k \neq (C_j)_k\}|$. The *relative distance* of the code is $\max_{i \neq j} \Delta(C_i, C_j)/t$.

We now define the problems that we study in this work. In all these problems there is some *error parameter* $0 < \varepsilon < 1/2$, and we want to design a fixed matrix $A \in \mathbb{R}^{m \times n}$ and deterministic algorithm *Out* for each problem satisfying the following.

Problem 1:. In the *ℓ_∞/ℓ_1 recovery problem*, also called the *point query problem*, $\forall x \in \mathbb{R}^n$, $x' = Out(Ax)$ satisfies $\|x - x'\|_\infty \leq \varepsilon\|x\|_1$. The pair (A, Out) furthermore satisfies the *k-tail guarantee* if actually $\|x - x'\|_\infty \leq \varepsilon\|x_{tail(k)}\|_1$.

Problem 2:. In the *inner product problem*, $\forall x, y \in \mathbb{R}^n$, $\alpha = Out(Ax, Ay)$ satisfies $|\alpha - \langle x, y\rangle| \leq \varepsilon\|x\|_1\|y\|_1$.

Problem 3:. In the *ℓ_1/ℓ_1 recovery problem with the k-tail guarantee*, $\forall x \in \mathbb{R}^n$, $x' = Out(Ax)$ satisfies $\|x - x'\|_1 \leq (1 + \varepsilon)\|x_{tail(k)}\|_1$.

Problem 4:. In the *ℓ_2 norm estimation problem*, $\forall x \in \mathbb{R}^n$, $\alpha = Out(Ax)$ satisfies $|\|x\|_2 - \alpha| \leq \varepsilon\|x\|_1$.

We note that for the first, second, and fourth problems above, our errors are additive and not relative. By additive error we mean the error has the form $\varepsilon \cdot Q$, where Q is a quantity depending on the problem definition, e.g., for the above four problems Q is $\|x_{tail(k)}\|_1$, $\|x\|_1\|y\|_1$, $\|x_{tail(k)}\|_1$, and $\|x\|_1$, respectively. A relative error for the first problem above would instead require that $|x'_i - x_i| \leq \varepsilon x_i$ for all $i \in [n]$. For the second and fourth problems, a relative error would be of the form $\varepsilon\langle x, y\rangle$ and $\varepsilon\|x\|_2$, respectively.

Relative error is impossible to achieve with a sublinear number of measurements. If A is a fixed matrix with $m < n$, then it has a non-trivial kernel. Since for all the problems above an *Out* procedure would have to output 0 when $Ax = 0$ to achieve bounded relative approximation, such a procedure would fail on any input vector in the kernel which is not the 0 vector.

For Problem 2 one could also ask to achieve additive error $\varepsilon\|x\|_p\|y\|_p$ for $p > 1$. For $y = e_i$ for a standard unit vector e_i, this would mean approximating x_i up to additive error $\varepsilon\|x\|_p$. This is not possible unless $m = \Omega(n^{2-2/p})$ for $1 < p \leq 2$ and $m = \Omega(n)$ for $p \geq 2$ [9].

For Problem 3, it is known that the analogous guarantee of returning x' for which $\|x - x'\|_2 \leq \varepsilon\|x_{tail(k)}\|_2$ is not possible unless $m = \Omega(n)$ [10].

1.3. Our Contributions and Related Work

We study the four problems stated above, where we have the deterministic guarantee that a single pair (A, Out) provides the desired guarantee for all input vectors simultaneously.

We first show that point query and inner product are equivalent up to changing ε by a constant factor. We then show that any "incoherent matrix" A can be used for these two problems to perform the linear measurements; that is, a matrix A whose columns have unit ℓ_2 norm and such that each pair of columns has dot product at most ε in magnitude. Such matrices can be obtained from the Johnson-Lindenstrauss (JL) lemma [11], almost pairwise independent sample spaces [12, 13], or error-correcting codes [14, 15], and they play a prominent role in compressed sensing [16, 17] and mathematical approximation theory [18]. The connection between point query and codes was implicit in [19], though a suboptimal code was used, and the observation that the more general class of incoherent matrices suffices is novel. This connection allows us to show that $m = O(\varepsilon^{-2}\min\{\log n, (\log n/\log(1/\varepsilon))^2\})$ measurements suffice, and where Out and the construction of A are completely deterministic.

The works [20, 21] have shown the lower bound that any incoherent matrix must have $m = \Omega(\varepsilon^{-2}\log n/\log(1/\varepsilon))$ when $\varepsilon = \Omega(1/\sqrt{n})$. Meanwhile the best known lower bound for point query is $m = \Omega(\varepsilon^{-2} + \varepsilon^{-1}\log(\varepsilon n))$ [22, 23, 24], and the previous best known upper bound was $m = O(\varepsilon^{-2}\log^2 n/(\log 1/\varepsilon + \log\log n))$ [19].

If the construction of A is allowed to be Las Vegas polynomial time, then we can use the Fast Johnson-Lindenstrauss transforms [25, 26, 27, 28] so that Ax can be computed quickly, e.g. in $O(n\log m)$ time as long as $m < n^{1/2-\gamma}$ [26], and with $m = O(\varepsilon^{-2}\log n)$. Our Out algorithm is equally fast. We also show that for point query, if we allow the measurement matrix A to be constructed by a polynomial Monte Carlo algorithm, then the $1/\varepsilon^2$-tail guarantee can be obtained essentially "for free", i.e. by keeping $m = O(\varepsilon^{-2}\log n)$. Previously the work [19] only showed how to obtain the $1/\varepsilon$-tail guarantee "for free" in this sense of not increasing m (though the m in [19] was worse).

We note that for randomized algorithms which succeed with high probability for any given input, it suffices to take $m = O(\varepsilon^{-1}\log n)$ by using the CountMin data structure [7], and this is optimal [29] (the lower bound in [29] is stated for the so-called heavy hitters problem, but also applies to the ℓ_∞/ℓ_1 recovery problem).

For the ℓ_1/ℓ_1 sparse recovery problem with the k-tail guarantee, we show a lower bound of $m = \Omega(k \log(\varepsilon n/k)/\varepsilon + k/\varepsilon^2)$. The best upper bound is $O(k \log(n/k)/\varepsilon^2)$ [30]. Our lower bound implies a separation for the complexity of this problem in the case that one must simply pick a random (A, Out) pair which works for some given input x with high probability (i.e. not for all x simultaneously), since [31] showed an $m = O(k \log n \log^3(1/\varepsilon)/\sqrt{\varepsilon})$ upper bound in this case. The first summand of our lower bound uses techniques used in [32, 31]. The second summand uses a generalization of an argument of Gluskin [24], which was later rediscovered by Ganguly [23], which showed the lower bound $m = \Omega(1/\varepsilon^2)$ for point query.

Finally, we show how to devise an appropriate (A, Out) for ℓ_2 norm estimation with $m = O(\varepsilon^{-2} \log(\varepsilon^2 n))$, which is optimal. The construction of A is randomized but then works for all x with high probability. The proof takes A according to known upper bounds on Gelfand widths, and the recovery procedure Out requires solving a simple convex program. As far as we are aware, this is the first work to investigate this problem in the deterministic setting. In the case that (A, Out) can be chosen randomly to work for any fixed x with high probability, one can use the AMS sketch [33] with $m = O(\varepsilon^{-2} \log(1/\delta))$ to succeed with probability $1 - \delta$ and to obtain the better guarantee $\varepsilon\|x\|_2$. The AMS sketch can also be used for the inner product problem to obtain error guarantee $\varepsilon\|x\|_2\|y\|_2$ with the same m.

2. Point Query and Inner Product Estimation

We first show that the problems of point query and inner product estimation are equivalent up to changing the error parameter ε by a constant factor.

Theorem 1. *Any solution (A, Out') to inner product estimation with error parameter ε yields a solution (A, Out) to the point query problem with error parameter ε. Also, a solution (A, Out) for point query with error ε yields a solution (A, Out') to inner product with error 12ε. The time complexities of Out and Out' are equal up to* $\mathrm{poly}(n)$ *factors.*

Proof. Let (A, Out') be a solution to the inner product problem such that $Out'(Ax, Ay) = \langle x, y\rangle \pm \varepsilon\|x\|_1\|y\|_1$. Then given $x \in \mathbb{R}^n$, to solve the point query problem we return the vector with $Out(Ax)_i = Out'(Ax, Ae_i)$, and our guarantees are immediate.

Now let (A, Out) be a solution to the point query problem. Then given $x, y \in \mathbb{R}^n$, let $x' = Out(Ax), y' = Out(Ay)$. Our estimate for the inner product is $Out'(Ax, Ay) = \left\langle x'_{head(1/\varepsilon)}, y'_{head(1/\varepsilon)} \right\rangle$. Observe the following: any coordinate i with $|x'_i| \geq 2\varepsilon\|x\|_1$ must have $|x_i| \geq \varepsilon\|x\|_1$, and thus there are at most $1/\varepsilon$ such coordinates. Also, any i with $|x_i| \geq 3\varepsilon\|x\|_1$ will have $|x'_i| \geq 2\varepsilon\|x\|_1$. Thus, $\{i : |x_i| \geq 3\varepsilon\|x\|_1\} \subseteq head(x', 1/\varepsilon)$, and similarly for x replaced with y. Now,

$$\begin{aligned}
&\left|\left\langle x'_{head(1/\varepsilon)}, y'_{head(1/\varepsilon)} \right\rangle - \langle x, y\rangle\right| \\
&\quad \leq \left|\left\langle x'_{head(1/\varepsilon)}, y'_{head(1/\varepsilon)} \right\rangle - \left\langle x_{head(x',1/\varepsilon)}, y_{head(y',1/\varepsilon)} \right\rangle\right| \\
&\qquad + \left|\left\langle x_{head(x',1/\varepsilon)}, y_{tail(y',1/\varepsilon)} \right\rangle\right| + \left|\left\langle x_{tail(x',1/\varepsilon)}, y_{head(y',1/\varepsilon)} \right\rangle\right| \\
&\qquad + \left|\left\langle x_{tail(x',1/\varepsilon)}, y_{tail(y',1/\varepsilon)} \right\rangle\right|
\end{aligned}$$

We can bound

$$\left|\left\langle x'_{head(1/\varepsilon)}, y'_{head(1/\varepsilon)} \right\rangle - \left\langle x_{head(x',1/\varepsilon)}, y_{head(y',1/\varepsilon)} \right\rangle\right|$$

by

$$\sum_{i \in head(x',1/\varepsilon)} \varepsilon\|y\|_1 x_i + \sum_{i \in head(x',1/\varepsilon)} \varepsilon\|x\|_1 y_i + \frac{1}{\varepsilon} \cdot \varepsilon^2\|x\|_1\|y\|_1 \leq 3\varepsilon\|x\|_1\|y\|_1.$$

We can also bound

$$\begin{aligned}
&\left|\left\langle x_{head(x',1/\varepsilon)}, y_{tail(y',1/\varepsilon)} \right\rangle\right| + \left|\left\langle x_{tail(x',1/\varepsilon)}, y_{head(y',1/\varepsilon)} \right\rangle\right| \\
&\qquad \leq \|x\|_1\|y_{tail(y',1/\varepsilon)}\|_\infty + \|x_{tail(x',1/\varepsilon)}\|_\infty\|y\|_1 \leq 6\varepsilon\|x\|_1\|y\|_1
\end{aligned}$$

Finally we have the bound

$$\left|\left\langle x_{tail(x',1/\varepsilon)}, y_{tail(y',1/\varepsilon)} \right\rangle\right| \leq \|x_{tail(x',1/\varepsilon)}\|_2\|y_{tail(y',1/\varepsilon)}\|_2. \tag{1}$$

Since $\|x_{tail(x',1/\varepsilon)}\|_\infty \leq 3\varepsilon\|x\|_1$ and $\|x_{tail(x',1/\varepsilon)}\|_1 \leq \|x\|_1$, we have that the value $\|x_{tail(x',1/\varepsilon)}\|_2$ is maximized when it has exactly $1/(3\varepsilon)$ coordinates each of value exactly $3\varepsilon\|x\|_1$, which yields ℓ_2 norm $\sqrt{3\varepsilon}\|x\|_1$, and similarly for x replaced with y. Thus the right hand side of Eq. (1) is bounded by $3\varepsilon\|x\|_1\|y\|_1$. Thus in summary, our total error in inner product estimation is $12\varepsilon\|x\|_1\|y\|_1$. $\square$

Since the two problems are equivalent up to changing ε by a constant factor, we focus on the point query problem. We first show that any *ε-incoherent matrix* A has a correct associated output procedure Out. By an ε-incoherent matrix, we mean an $m \times n$ matrix A for which all columns A_i of A have unit ℓ_2 norm, and for all $i \neq j$ we have $|\langle A_i, A_j \rangle| \leq \varepsilon$. We have the following lemma, which follows readily from the definition of ε-incoherence.

Lemma 2. *Any ε-incoherent matrix A has an associated* $\mathrm{poly}(mn)$*-time deterministic recovery procedure Out for which (A, Out) is a solution to the point query problem. In fact, for any $x \in \mathbb{R}^n$, given Ax and $i \in [n]$, the output x'_i satisfies $|x'_i - x_i| \leq \varepsilon \|x_{-i}\|_1$.*

Proof. Let $x \in \mathbb{R}^n$ be arbitrary. We define $Out(Ax) = A^T Ax$. Observe that for any $i \in [n]$, we have

$$x'_i = A_i^T Ax = \sum_{j=1}^{n} \langle A_i, A_j \rangle x_j = x_i \pm \varepsilon \|x_{-i}\|_1.$$

□

It is known that any ε-incoherent matrix has $m = \Omega((\log n)/(\varepsilon^2 \log 1/\varepsilon))$ [20, 21], and the JL lemma implies such matrices with $m = O((\log n)/\varepsilon^2)$ [11]. For example, there exist matrices in $\{-1/\sqrt{m}, 1/\sqrt{m}\}^{m \times n}$ satisfying this property [34], which can also be found in $\mathrm{poly}(n)$ time [35] (we note that [35] gives running time exponential in precision, but the proof holds if the precision is taken to be $O(\log(n/\varepsilon))$. It is also known that ε-incoherent matrices can be obtained from almost pairwise independent sample spaces [12, 13] or error-correcting codes (see [15, 36], which have several constructions), and thus these tools can also be used to solve the point query problem. The connection to codes was already implicit in [19], though the code used in that work is suboptimal, as we will show soon. Below we elaborate on what bounds these tools provide for ε-incoherent matrices, and what they imply for the point query problem.

ε-Incoherent matrices from JL:. The upside of the connection to the JL lemma is that we can obtain matrices A for the point query problem such that Ax can be computed quickly, via the Fast Johnson-Lindenstrauss Transform introduced by Ailon and Chazelle [25] or related subsequent works. The JL lemma states the following.

Theorem 3 (JL lemma)**.** *For any $x_1, \ldots, x_N \in \mathbb{R}^n$ and any $0 < \varepsilon < 1/2$, there exists $A \in \mathbb{R}^{m \times n}$ with $m = O(\varepsilon^{-2} \log N)$ such that for all $i, j \in [N]$ we have $\|Ax_i - Ax_j\|_2 = (1 \pm \varepsilon)\|x_i - x_j\|_2$.*

Consider the matrix A obtained from the JL lemma when the set of vectors is $\{0, e_1, \ldots, e_n\} \in \mathbb{R}^n$. Then columns A_i of A have ℓ_2 norm $1 \pm \varepsilon$, and furthermore for $i \neq j$ we have $|\langle A_i, A_j \rangle| = (\|A_i - A_j\|_2^2 - \|A\|_i^2 - \|A\|_j^2)/2 = ((1 \pm \varepsilon)^2 2 - (1 \pm \varepsilon) - (1 \pm \varepsilon))/2 \leq 2\varepsilon + \varepsilon^2/2$. By scaling each column to have ℓ_2 norm exactly 1, we still preserve that dot products between pairs of columns are $O(\varepsilon)$ in magnitude.

ε-incoherent matrices from almost pairwise independence:. Next we elaborate on the connection between ε-incoherent matrices and almost pairwise independence.

Definition 4. *An* ε-almost k-wise independent sample space *is a set $S \subseteq \{-1, 1\}^n$ satisfying the following. For any $T \subsetneq [n]$, $|T| = k$, the ℓ_1 distance between the uniform distribution over $\{-1, 1\}^k$ and the distribution of $x(T)$ when x is drawn uniformly at random from S is at most ε. Here $x(T) \in \{-1, 1\}^{|T|}$ is the bitstring x projected onto the coordinates in T.*

Note that if S is ε-almost k-wise independent, then for any $|T| = k$, $|\mathbb{E}_{x \in S} \prod_{i \in T} x_i| \leq \varepsilon$. Therefore if we choose $k = 2$ and form a $|S| \times n$ matrix where the rows of A are the elements of S, divided by a scale factor of $\sqrt{|S|}$, then A is ε-incoherent. Known constructions of almost pairwise independent sample spaces give $|S| = \mathrm{poly}(\varepsilon^{-1} \log n)$ [12, 37, 13]. We do not delve into the specific bounds on $|S|$ since they yield worse results than the JL-based construction above. The probabilistic method implies that such an S exists with $S = O(\varepsilon^{-2} \log n)$, matching the JL construction, but an explicit almost pairwise independent sample space with this size is currently not known.

ε-incoherent matrices from codes:. Finally we explain the connection between ε-incoherent matrices and codes. This connection is discussed in previous work [20, 14, 15] and not novel, but we elaborate on the connection for the sake of self-containment. Let $\mathcal{C} = \{C_1, \ldots, C_n\}$ be a code with alphabet size q, block length t, and relative distance $1 - \varepsilon$. The fact that such a code gives rise to a matrix $A \in \mathbb{R}^{m \times n}$ for point query with error parameter ε was implicit in [19], but we make it explicit here.

We let $m = qt$ and conceptually partition the rows of A arbitrarily into t sets each of size q. For the column A_i, let $(A_i)_{j,k}$ denote the entry of A_i in the kth coordinate of the jth block. We set $(A_i)_{j,k} = 1/\sqrt{t}$ if $(C_i)_j = k$, and $(A_i)_{j,k} = 0$ otherwise. Said differently, for $y = Ax$ we label the entries of y with double-indices $(i,j) \in [t] \times [q]$. We define deterministic hash functions $h_1, \ldots, h_t : [n] \to [q]$ by $h_i(j) = (C_j)_i$, and we set $y_{i,j} = \sum_{k:h_i(k)=j} x_k/\sqrt{t}$. Our procedure *Out* produces a vector x' with $x'_k = \sum_{i=1}^{t} y_{i,h_i(k)}$. Each column has exactly t non-zero entries of value $1/\sqrt{t}$, and thus has ℓ_2 norm 1. Furthermore, for $i \neq j$, $\langle A_i, A_j \rangle = (t - \Delta(C_i, C_j))/t \leq \varepsilon$.

The work [19] instantiated the above with the following *Chinese remainder code* [38, 39, 40]. Let $p_1 < \ldots < p_t$ be primes, and let $q = p_t$. We let $(C_i)_j = i \mod p_j$. To obtain n codewords with relative distance $1 - \varepsilon$, this construction required setting $t = O(\varepsilon^{-1} \log n/(\log(1/\varepsilon) + \log\log n))$ and $p_1, p_t = \Theta(\varepsilon^{-1} \log n) = O(t \log t)$. The proof uses that for $i, j \in [n]$, $|i - j|$ has at most $\log_{p_1} n$ prime factors greater than or equal to p_1, and thus C_i, C_j can have at most $\log_{p_1} n$ many equal coordinates. This yields $m = tq = O(\varepsilon^{-2} \log^2 n/(\log 1/\varepsilon + \log\log n))$.

We observe here that this bound is never optimal. A random code with $q = 2/\varepsilon$ and $t = O(\varepsilon^{-1} \log n)$ has the desired properties by applying the Chernoff bound on a pair of codewords, then a union bound over codewords (alternatively, such a code is promised by the Gilbert-Varshamov (GV) bound).

If ε is sufficiently small, a Reed-Solomon code performs even better. That is, we take a finite field $\mathbb{F}_q$ for $q = \Theta(\varepsilon^{-1} \log n/(\log\log n + \log(1/\varepsilon)))$ and $q = t$, and each C_i corresponds to a distinct degree-d polynomial p_i over $\mathbb{F}_q$ for $d = \Theta(\log n/(\log\log n + \log(1/\varepsilon)))$ (note there are at least $q^d > n$ such polynomials). We set $(C_i)_j = p_i(j)$. The relative distance is as desired since $p_i - p_j$ has at most d roots over $\mathbb{F}_q$ and thus can be 0 at most $d \leq \varepsilon t$ times. This yields $qt = O(\varepsilon^{-2}(\log n/(\log\log n + \log(1/\varepsilon))^2)$, which surpasses the GV bound for $\varepsilon < 2^{-\Omega(\sqrt{\log n})}$, and is always better than the Chinese remainder code. We note that this construction of a binary matrix based on Reed-Solomon codes is identical to one used by Kautz and Singleton in the different context of group testing [41].

In Table 1 we elaborate on what known constructions of codes and JL matrices provide for us in terms of point query. In the case of running time for the Reed-Solomon construction, we use that degree-d polynomials can be evaluated on $d + 1$ points in a total of $O(d \log^2 d \log\log d)$ field operations over $\mathbb{F}_q$ [43, Ch. 10]. In the case of [26], the constant $\gamma > 0$ can be chosen arbitrarily, and the constant in the big-Oh depends on $1/\gamma$. We note that except in the case of Reed-Solomon codes, the construction of A is random-

Time	m	Details	Explicit?
$O((n\log n)/\varepsilon^2)$	$O(\varepsilon^{-2}\log n)$	$A \in \{-1/\sqrt{m}, 1/\sqrt{m}\}^{m\times n}$ [34, 35]	yes
$O((n\log n)/\varepsilon)$	$O(\varepsilon^{-2}\log n)$	sparse JL [42], GV code	no
$O(nd\log^2 d\log\log d/\varepsilon)$	$O(d^2/\varepsilon^2)$	Reed-Solomon code	yes
$O_\gamma(n\log m + m^{2+\gamma})$	$O(\varepsilon^{-2}\log n)$	FFT-based JL [26]	no
$O(n\log n)$	$O(\varepsilon^{-2}\log^5 n)$	FFT-based JL [27, 28]	no

Table 1: Implications for point query from JL matrices and codes. Time indicates the running time to compute Ax given x. In the case of Reed-Solomon, $d = O(\log n/(\log\log n + \log(1/\varepsilon)))$. We say the construction is "explicit" if A can be computed in deterministic time poly(n); otherwise we only provide a polynomial time Las Vegas algorithm to construct A.

ized (though once A is generated, incoherence can be verified in polynomial time, thus providing a poly(n)-time Las Vegas algorithm).

Note that Lemma 2 did not just give us error $\varepsilon\|x\|_1$, but actually gave us $|x_i - x'_i| \le \varepsilon\|x_{-i}\|_1$, which is stronger. We now show that an even stronger guarantee is possible. We will show that in fact it is possible to obtain $\|x - x'\|_\infty \le \varepsilon\|x_{tail(1/\varepsilon^2)}\|_1$ while increasing m by only an additive $O(\varepsilon^{-2}\log(\varepsilon^2 n))$, which is less than our original m except potentially in the Reed-Solomon construction. The idea is to, in parallel, recover a good approximation of $x_{head(1/\varepsilon^2)}$ with error proportional to $\|x_{tail(1/\varepsilon^2)}\|_1$ via compressed sensing, then to subtract from Ax before running our recovery procedure. We now give details.

We in parallel run a *k-sparse recovery* algorithm which has the following guarantee: there is a pair (B, Out') such that for any $x \in \mathbb{R}^n$, we have that $x' = Out'(Bx) \in \mathbb{R}^n$ satisfies $\|x' - x\|_2 \le O(1/\sqrt{k})\|x_{tail(k)}\|_1$. Such a matrix B can be taken to have the *restricted isometry property of order* k (k-RIP), i.e. that it preserves the ℓ_2 norm up to a small multiplicative constant factor for all k-sparse vectors in $\mathbb{R}^n$.[1] It is known [44] that any such x' also satisfies the guarantee that $\|x'_{head(k)} - x\|_1 \le O(1)\|x_{tail(k)}\|_1$, where $x'_{head(k)}$ is the vector which agrees with the value of x' on the top k coordinates in magnitude,

[1] Unfortunately currently the only known constructions of k-RIP constructions with the values of m we discuss are Monte Carlo, forcing our algorithms in this section with the k-tail guarantee to only be Monte Carlo polynomial time when constructing the measurement matrix.

and is 0 on the remaining coordinates. Moreover, it is also known [45] that if B satisfies the JL lemma for a particular set of $N = (en/k)^{O(k)}$ points in $\mathbb{R}^n$, then B will be k-RIP. The associated output procedure Out' takes Bx and outputs $\text{argmin}_{z|Bx=Bz}\|z\|_1$ by solving a linear program [46]. All the JL matrices in Table 1 provide this guarantee with $O(k\log(en/k))$ rows, except for the last row which satisfies k-RIP with $O(k\log(en/k)\log^2 k\log(k\log n))$ rows [47].

Theorem 5. *Let A be an ε-incoherent matrix, and let B be k-RIP. Then there is an output procedure Out which for any $x \in \mathbb{R}^n$, given only Ax, Bx, outputs a vector x' with $\|x' - x\|_\infty \leq \varepsilon\|x_{tail(k)}\|_1$.*

Proof. Given Bx, we first run the k-sparse recovery algorithm to obtain a vector y with $\|x-y\|_1 = O(1)\|x_{tail(k)}\|_1$. We then construct our output vector x' coordinate by coordinate. To construct x'_i, we replace y_i with 0, obtaining the vector z^i. Then we compute $A(x - z^i)$ and run the point query output procedure associated with A and index i. The guarantee is that the output w^i of the point query algorithm satisfies $|w^i_i - (x - z^i)_i| \leq \varepsilon\|(x - z^i)_{-i}\|_1$, where

$$\|(x - z^i)_{-i}\|_1 = \|(x - y)_{-i}\|_1 \leq \|x - y\|_1 = O(1)\|x_{tail(k)}\|_1,$$

and so $|(w^i + z^i)_i - x_i| = O(\varepsilon)\|x_{tail(k)}\|_1$. If we define our output vector by $x'_i = w^i_i + z^i_i$ and rescale ε by a constant factor, this proves the theorem. □

Theorem 5 may seem similar to the work of Krahmer and Ward [28], which tells us that from a k-RIP matrix we can get a JL matrix. Below, we will set $k = 1/\varepsilon^2$ in Theorem 5, so [28] would tell us that this matrix preserves the norms, up to a constant factor, of a fixed set of $\exp(\varepsilon^{-2})$ points. This is not the same conclusion of Theorem 5, which states that for every vector x, Out outputs a vector x' with the ℓ_∞/ℓ_1 guarantee.

By setting $k = 1/\varepsilon^2$ in Theorem 5 and stacking the rows of a k-RIP and ε-incoherent matrix each with $O((\log n)/\varepsilon^2)$ rows (here, by stacking the rows of two matrices A and B, we mean forming the matrix C whose rows are the union of the rows of A and of B) we obtain the following corollary, which says that by increasing the number of measurements $m = O(\varepsilon^{-2}\log n)$ by only a constant factor, we can obtain a stronger tail guarantee.

Corollary 6. *There is an $m \times n$ matrix A and associated output procedure Out which for any $x \in \mathbb{R}^n$, given Ax, outputs a vector x' with $\|x' - x\|_\infty \leq \varepsilon\|x_{tail(1/\varepsilon^2)}\|_1$. Here $m = O((\log n)/\varepsilon^2)$.*

Of course, again by using various choices of ε-incoherent matrices and k-RIP matrices, we can trade off the number of linear measurements for various tradeoffs in the running time and tail guarantee. It is also possible to obtain a tail-error guarantee for inner product. While this is implied black-box by reducing from point query with the k-tail guarantee, by performing the argument from scratch we can obtain a better error guarantee involving mixed ℓ_1 and ℓ_2 norms.

Theorem 7. *Suppose $1/\varepsilon^2 < n/2$. There is an (A, Out) with $A \in \mathbb{R}^{m\times n}$ for $m = O(\varepsilon^{-2}\log n)$ such that for any $x, y \in \mathbb{R}^n$, $Out(Ax, Ay)$ gives an output which is $\langle x, y\rangle \pm \varepsilon(\|x\|_2\|y_{tail(1/\varepsilon^2)}\|_1 + \|x_{tail(1/\varepsilon^2)}\|_1\|y\|_2) + \varepsilon^2\|x_{tail(1/\varepsilon^2)}\|_1\|y_{tail(1/\varepsilon^2)}\|_1$.*

Proof. Using the ℓ_2/ℓ_1 sparse recovery mentioned in Section 2, we can recover x', y' such that $\|x - x'\|_2 \le \varepsilon\|x_{tail(1/\varepsilon^2)}\|_1$, and similarly for $y - y'$. The number of measurements is the number of measurements required for $1/\varepsilon^2$-RIP, which is $O(\varepsilon^{-2}\log(\varepsilon^2 n))$. Our estimation procedure Out simply outputs $\langle x', y'\rangle$. Then,

$$\begin{aligned}
|\langle x, y\rangle - \langle x', y'\rangle| &= \left|\sum_i x_i(y_i - y_i') + y_i'(x_i - x_i')\right| \\
&\le \left|\sum_i x_i(y_i - y_i')\right| + |y_i'(x_i - x_i')| \\
&\le \|x\|_2\|y - y'\|_2 + \|y'\|_2\|x - x'\|_2 \\
&\le \|x\|_2\|y - y'\|_2 + (\|y - y'\|_2 + \|y\|_2)\|x - x'\|_2
\end{aligned}$$

The theorem then follows by our bounds on $\|x - x'\|_2$ and $\|y - y'\|_2$. □

Note that again A, Out in Theorem 7 can be taken to be applied efficiently by using RIP matrices based on the Fast Johnson-Lindenstrauss Transform.

3. Lower Bound for ℓ_∞/ℓ_1 Recovery

Here we provide a lower bound for the point query problem addressed in Section 2.

Theorem 8. *Let $0 < \varepsilon < \varepsilon_0$ for some universal constant $\varepsilon_0 < 1$. Suppose $1/\varepsilon^2 < n/2$, and A is an $m \times n$ matrix for which given Ax it is always possible to produce a vector x' such that $\|x - x'\|_\infty \le \varepsilon\|x_{tail(k)}\|_1$. Then $m = \Omega(k\log(n/k)/\log k + \varepsilon^{-2} + \varepsilon^{-1}\log n)$.*

Proof. The lower bound of $\Omega(\varepsilon^{-2})$ for any k is already proven in [23].

The lower bound of $\Omega(k \log(n/k)/\log k + \varepsilon^{-1} \log n)$ follows from a standard volume argument. For completeness, we give the argument below. Let $B_1(x, r)$ denote the ℓ_1 ball centered at x of radius r. We use the following lemma by Gilbert-Varshamov (see e.g. [32]).

Lemma 9 ([32, Lemma 3.1])**.** *For any $q, k \in \mathbb{Z}^+, \varepsilon \in \mathbb{R}^+$ with $\varepsilon < 1 - 1/q$, there exists a set $S \subset \{0,1\}^{qk}$ of binary vectors with exactly k ones, such that S has minimum Hamming distance $2\varepsilon k$ and*

$$\log |S| > (1 - H_q(\varepsilon)) k \log q$$

where H_q is the q-ary entropy function $H_q(x) = -x \log_q \frac{x}{q-1} - (1-x)\log_q(1-x)$.

Assume $\varepsilon < 1/200$. Consider a set S of n dimensional binary vectors in $\mathbb{R}^n$ with exactly $1/(5\varepsilon)$ ones such that minimum Hamming distance between any two vectors in S is at least $1/(10\varepsilon)$. By the above lemma, we can get $\log |S| = \Omega(\varepsilon^{-1} \log(\varepsilon n))$. For any $x \in S$, and $z \in B_1(x, 1/(200\varepsilon))$, we have $\|z_{tail(k)}\|_1 \leq \|z\|_1 \leq 1/(5\varepsilon) + 1/(200\varepsilon) = 41/(200\varepsilon)$, $z \in B_1(0, 41/(200\varepsilon))$, and there are at most $4/(200\varepsilon)$ coordinates that are ones in x and smaller than $3/4$ in z, and at most $4/(200\varepsilon)$ coordinates that are zeros in x and at least $1/4$ in z. If z' is a good approximation of z, then $\|z' - z\|_\infty \leq 41/200 < 1/4$ so the indices of the coordinates of z' at least $1/2$ differ from those of x at most $8/(200\varepsilon) < 1/(20\varepsilon)$ places. Thus, for any two different vectors $x, y \in S$ and $z \in B_1(x, 1/(200\varepsilon)), t \in B_1(y, 1/(200\varepsilon))$, the outputs for inputs z and t are different and hence, we must have $Az \neq At$. Notice that for the mapping $x \to Ax$, the image of $B_1(x, 1/(200\varepsilon))$ is the translated version of the image of $B_1(0, 41/(200\varepsilon))$ scaled down in every dimension by a factor of 41. For x's in S, the images of $B(x, 1/(200\varepsilon))$ are disjoint subsets of the image of $B(0, 41/(200\varepsilon))$. By comparing their volumes, we have $41^m \geq |S|$, implying $m = \Omega(\varepsilon^{-1} \log(\varepsilon n))$.

Next, consider the set S' of all vectors in $\mathbb{R}^n$ with exactly k coordinates equal to $1/k$ and the rest equal to 0. For any $x \in S'$, and $z \in B_1(x, 1/(3k))$, we have $\|z_{tail(k)}\|_1 \leq 1/(3k)$ and $z \in B_1(0, 1 + 1/(3k))$ centered at the origin. Therefore, if z' is a good approximation of z, the indices of the largest k coordinates of z' are exactly the same as those of x. Thus, for any two different vectors $x, y \in S'$ and $z \in B_1(x, 1/(3k), t \in B_1(y, 1/(3k))$, the outputs for inputs z and t are different and hence, we must have $Az \neq At$. Notice that for the mapping $x \to Ax$, the image of $B_1(x, 1/(3k))$ is the translated version of the image of $B_1(0, 1 + 1/(3k))$ scaled down in every dimension by a factor of $3k+1$. For x's in S', the images of $B(x, 1/(3k))$ are disjoint subsets of the image of $B(0, 1 + 1/(3k))$. By comparing their volumes, we have $(3k+1)^m \geq |S'| \geq (n/k)^k$, implying $m = \Omega(k \log(n/k)/\log k)$.

□

4. Lower Bounds for ℓ_1/ℓ_1 recovery

Recall in the ℓ_1/ℓ_1-recovery problem, we would like to design a matrix $A \in \mathbb{R}^{m\times n}$ such that for any $x \in \mathbb{R}^n$, given Ax we can recover $x' \in \mathbb{R}^n$ such that $\|x - x'\|_1 \leq (1+\varepsilon)\|x_{tail(k)}\|_1$. We now show two lower bounds.

Theorem 10. *Let $0 < \varepsilon < 1/16$ be arbitrary, and k be an integer. Suppose $k/\varepsilon^2 < (n-1)/2$. Then any matrix $A \in \mathbb{R}^{m\times n}$ which allows ℓ_1/ℓ_1-recovery with the k-tail guarantee with error ε must have $m \geq \min\{n/2, (1/16)k/\varepsilon^2\}$.*

Proof. Without loss of generality we may assume that the rows of A are orthonormal. This is because first we can discard rows of A until the rows remaining form a basis for the rowspace of A. Call this new matrix with potentially fewer rows A'. Note that any dot products of rows of A with x that the recovery algorithm uses can be obtained by taking linear combinations of entries of $A'x$. Next, we can then find a matrix $T \in \mathbb{R}^{m\times m}$ so that TA' has orthonormal rows, and given $TA'x$ we can recover $A'x$ in post-processing by left-multiplication with T^{-1}.

We henceforth assume that the rows of A are orthonormal. Since $A\cdot 0 = 0$, and our recovery procedure must in particular be accurate for $x = 0$, the recovery procedure must output $x' = 0$ for any $x \in ker(A)$. We consider $x = (I - A^T A)y$ for $y = \sum_{i=1}^k \sigma_i e_{\pi(i)}$. Here π is a random permutation on n elements, and $\sigma_1, \ldots, \sigma_k$ are independent and uniform random variables in $\{-1, 1\}$. Since $x \in ker(A)$, which follows since $AA^T = I$ by orthonormality of the rows of A, the recovery algorithm will output $x' = 0$. Nevertheless, we will show that unless $m \geq \min\{n/2, (1/16)k/\varepsilon^2\}$, we will have $\|x\|_1 > (1+\varepsilon)\|x_{tail(k)}\|_1$ with positive probability so that by the probabilistic method there exists $x \in ker(A)$ for which $x' = 0$ is not a valid output.

If $m \geq n/2$ we are done. Otherwise, since $\|x\|_1 = \|x_{head(k)}\|_1 + \|x_{tail(k)}\|_1$, it is equivalent to show that $\|x_{head}(k)\|_1 > \varepsilon\|x_{tail}(k)\|_1$ with positive probability. We first have

$$\begin{aligned}
\mathbb{E}\,\|x_{tail}(k)\|_1 &\le \mathbb{E}\,\|x\|_1 \\
&\le \mathbb{E}\,\|y\|_1 + \mathbb{E}\,\|A^T Ay\|_1 \\
&\le k + \sqrt{n}\cdot\left(\mathbb{E}\,\|A^T Ay\|_2^2\right)^{1/2} && (2)\\
&= k + \sqrt{n}\cdot\left(\mathbb{E}\,y^T A^T A A^T Ay\right)^{1/2} \\
&= k + \sqrt{n}\cdot\left(\mathbb{E}\,y^T A^T Ay\right)^{1/2} && (3)\\
&= k + \sqrt{n}\cdot\left(\mathbb{E}\left\langle \sum_{j=1}^{k}\sigma_j A_{\pi(j)}, \sum_{j=1}^{k}\sigma_j A_{\pi(j)}\right\rangle\right)^{1/2} \\
&= k + \sqrt{n}\cdot\left(\sum_{j=1}^{k}\mathbb{E}\,\|A_{\pi(j)}\|_2^2\right)^{1/2} \\
&= k + \sqrt{kn}\cdot(\mathbb{E}\,\|A_{\pi(1)}\|_2^2)^{1/2} \\
&= k + \sqrt{km}. && (4)
\end{aligned}$$

Eq. (2) uses Cauchy-Schwarz. Eq. (3) follows since A has orthonormal rows, so that $AA^T = I$. Eq. (4) uses that the sum of squared entries over all columns equals the sum of squared entries over rows, which is m since the rows have unit norm.

We now turn to lower bounding $\|x_{head(k)}\|_1$. Define $\eta_{i,j} = \sigma_j/\sigma_i$ so that for fixed i the $\eta_{i,j}$ are independent and uniform ± 1 random variables (except for $\eta_{i,i}$, which is 1). We have

$$\begin{aligned}
\|x_{head(k)}\|_1 &\ge \|x_{\pi([k])}\|_1 \\
&= \sum_{i=1}^{k}\left|e_{\pi(i)}^T y - e_{\pi(i)}^T A^T y\right| \\
&= \sum_{i=1}^{k}\left|1 - \sum_{j=1}^{k}\eta_{i,j}\left\langle A_{\pi(i)}, A_{\pi(j)}\right\rangle\right| && (5)
\end{aligned}$$

Now, for fixed $i \in [k]$ we have

$$\mathbb{E}\left|\sum_{j=1}^{k}\eta_{i,j}\left\langle A_{\pi(i)}, A_{\pi(j)}\right\rangle\right| \le \left(\mathbb{E}\left(\sum_{j=1}^{k}\eta_{i,j}\left\langle A_{\pi(i)}, A_{\pi(j)}\right\rangle\right)^2\right)^{1/2}$$

$$
\begin{aligned}
&= \sqrt{k} \cdot \left(\mathbb{E} \left\langle A_{\pi(1)}, A_{\pi(2)} \right\rangle^2\right)^{1/2} \\
&< \sqrt{\frac{k}{n(n-1)}} \cdot \|A^T A\|_F \\
&= \sqrt{\frac{k}{n(n-1)}} \cdot \|A\|_F \qquad (6) \\
&= \sqrt{\frac{mk}{n(n-1)}} \\
&< \frac{1}{8} \qquad (7)
\end{aligned}
$$

Eq. (6) follows since $\|A^T A\|_F^2 = \text{trace}(A^T A A^T A) = \text{trace}(A^T A) = \|A\|_F^2$. Here $\|\cdot\|_F$ denotes the Frobenius norm, i.e. $\|B\|_F = \sqrt{\sum_{i,j} B_{i,j}^2}$.

Putting things together, by Eq. (4), for $m < (1/16)k/\varepsilon^2$ a random vector x has $\|x_{tail(k)}\|_1 \leq 2k + 2\sqrt{km} \leq 4\sqrt{km}$ with probability strictly larger than 1/2 by Markov's inequality. Also, call an $i \in [k]$ *bad* if $|x_{\pi(i)}| \leq 1/2$. Combining Eq. (5) with Eq. (7) and using a Markov bound we have that the expected number of bad indices $i \in [k]$ is less than $k/4$. Thus the probability that a random x has more than $k/2$ bad indices is less than 1/2 by Markov's inequality. Thus by a union bound, with probability strictly larger than $1 - (1/2) - (1/2) = 0$, a random x taken as described simultaneously has $\|x_{tail(k)}\|_1 \leq 4\sqrt{km}$ and less than $k/2$ bad indices, the latter of which implies that $\|x_{head(k)}\|_1 > k/2$. Thus there exists a vector in $x \in \ker(A)$ for which $\|x_{head(k)}\|_1 > \varepsilon\|x_{tail(k)}\|_1$ when $m < (1/16)k/\varepsilon^2$, and we thus must have $m \geq (1/16)k/\varepsilon^2$. □

We now give another lower bound via a different approach. As in [32, 31], we use 2-party communication complexity to prove an $\Omega((k/\varepsilon)\log(\varepsilon n/k))$ bound on the number of rows of any ℓ_1/ℓ_1 sparse recovery scheme. The main difference from prior work is that we use deterministic communication complexity and a different communication problem.

We give a brief overview of the concepts from communication complexity that we need, referring the reader to [48] for further details. Formally, in the 1-way deterministic 2-party communication complexity model, there are two parties, Alice and Bob, holding inputs $x, y \in \{0,1\}^r$, respectively. The goal is to compute a Boolean function $f(x, y)$. A single message $m(x)$ is sent from Alice to Bob, who then outputs $g(m(x), y)$ for a Boolean function g. The protocol is correct if $g(m(x), y) = f(x, y)$ for all inputs x and y. The 1-way deterministic communication complexity of f, denoted $D^{1-way}(f)$, is the minimum over all correct protocols, of the maximum message length $|m(x)|$ over all inputs x.

We use the $EQ(x, y) : \{0,1\}^r \times \{0,1\}^r \to \{0,1\}$ function, which is 1 if $x = y$ and 0 otherwise. It is known [48] that $D^{1-way}(EQ) = r$. We show how to use a pair (A, Out) with the property that for all vectors z, the output z' of $Out(Az)$ satisfies $\|z - z'\|_1 \le (1+\varepsilon)\|z_{tail(k)}\|_1$, to construct a correct protocol for EQ on strings $x, y \in \{0,1\}^r$ for $r = \Theta((k/\varepsilon)\log n \log(\varepsilon n/k))$. We then show how this implies the number of rows of A is $\Omega((k/\varepsilon)\log(\varepsilon n/k))$.

We can assume the rows of A are orthonormal as in the beginning of the proof of Theorem 10. Let A' be the matrix where we round each entry of A to $b = O(\log n)$ bits per entry. We use the following Lemma of [32].

Lemma 11. *(Lemma 5.1 of [32]) Consider any $m \times n$ matrix A with orthonormal rows. Let A' be the result of rounding A to b bits per entry. Then for any $v \in \mathbb{R}^n$ there exists an $s \in \mathbb{R}^n$ with $A'v = A(v - s)$ and $\|s\|_1 \le n^2 2^{-b}\|v\|_1$.*

Theorem 12. *Any matrix A which allows ℓ_1/ℓ_1-recovery with the k-tail guarantee with error ε satisfies $m = \Omega((k/\varepsilon)\log(\varepsilon n/k))$.*

Proof. Let S be the set of all strings in $\{0, c\varepsilon/k\}^n$ containing exactly $k/(c\varepsilon)$ entries equal to $c\varepsilon/k$, for an absolute constant $c > 0$ specified below. Observe that $\log|S| = \Theta((k/\varepsilon)\log(\varepsilon n/k))$.

In the $EQ(x, y)$ problem, Alice is given a string x of length $r = \log n \cdot \log|S|$. Alice splits x into $\log n$ contiguous chunks $x^1, \ldots, x^{\log n}$, each containing $r/\log n$ bits. She uses x^i as an index to choose an element of S. She sets

$$u = \sum_{i=1}^{\log n} 2^i x^i,$$

and transmits $A'u$ to Bob.

Bob is given a string y of length r in the $EQ(x, y)$ problem. He performs the same procedure as Alice, namely, he splits y into $\log n$ contiguous chunks $y^1, \ldots, y^{\log n}$, each containing $r/\log n$ bits. He uses y^i as an index to choose an element of S. He sets

$$v = \sum_{i=1}^{\log n} 2^i y^i.$$

Given $A'u$, he outputs $A'(u-v)$, which by applying Lemma 11 once to Au and once to Av, is equal to $A(u-v-s)$ for an s with $\|s\|_1 \leq n^2 2^{-b}(\|u\|_1+\|v\|_1) \leq 1/n$, where the last inequality follows for sufficiently large $b = O(\log n)$. If $A'(u-v) = 0$, he outputs that x and y are equal, otherwise he outputs that x and y are not equal.

Observe that if $x = y$, then $u = v$, and so Bob outputs the correct answer. Next, we consider $x \neq y$, and show that $A'(u-v) \neq 0$. To do this, it suffices to show that $\|(u-v-s)_{head(k)}\|_1 > \varepsilon\|u-v-s\|_1$, as then $Out(A(u-v-s))$ could not output 0, which would also mean that $A'(u-v) \neq 0$.

To show that $\|(u-v-s)_{head(k)}\|_1 > \varepsilon\|u-v-s\|_1$, first observe that $\|s\|_1 \leq 1/n$, so by the triangle inequality, it is enough to show that $\|(u-v)_{head(k)}\|_1 > 2\varepsilon\|u-v\|_1$.

Let $z^1 = u - v$. Let $i \in [\log n]$ be the largest index of a chunk for which $x^i \neq y^i$, and let j_1 be such that $|z^1_{j_1}| = \|z^1\|_\infty$. Then $|z^1_{j_1}| = c\varepsilon \cdot 2^i/k$, while

$$\|z^1\|_1 \leq 2 \cdot 2 + 2 \cdot 4 + 2 \cdot 8 + \cdots + 2 \cdot 2^i < 2 \cdot 2^{i+1} = 2^{i+2}.$$

Let z^2 be z^1 with coordinate j_1 removed. Repeating this argument on z^2, we again find a coordinate j_2 with $|z^2_{j_2}| \geq \frac{c\varepsilon}{4k} \cdot \|z^2\|_1$. It follows by induction that after k steps, and for $\varepsilon > 0$ less than an absolute constant $\varepsilon_0 > 0$,

$$\|(u-v)_{tail(k)}\|_1 \leq \left(1 - \frac{c\varepsilon}{4k}\right)^k \|u-v\|_1 \leq (1 - c\varepsilon)\,\|u-v\|_1,$$

and so

$$\|(u-v)_{head(k)}\|_1 > c\varepsilon\|u-v\|_1.$$

Setting $c = 2$, we have that $\|(u-v)_{head(k)}\|_1 > 2\varepsilon\|u-v\|_1$, as desired.

Finally, observe the communication of this protocol is the number of rows of A times $O(\log n)$, since this is the number of bits required to specify $m(x) = A'u$. It follows by the communication lower bound for EQ, that the number of rows of A is $\Omega(r/\log n) = \Omega((k/\varepsilon)\log(\varepsilon n/k))$. This proves our theorem. □

5. Deterministic Norm Estimation and the Gelfand Width

Theorem 13. *For $1 \leq p < q \leq \infty$, let m be the minimum number such that there is an $n-m$ dimensional subspace S of $\mathbb{R}^n$ satisfying $\sup_{v \in S} \frac{\|v\|_q}{\|v\|_p} \leq \varepsilon$. Then there is an $m \times n$ matrix A and associated output procedure Out which for any $x \in \mathbb{R}^n$, given Ax, outputs an estimate of $\|v\|_q$ with additive error at* *most $\varepsilon\|v\|_p$. Moreover, any matrix A with fewer rows will fail to perform the same task.*

Proof. Consider a matrix A whose kernel is such a subspace. For any sketch z, we need to return a number in the range $[\|x\|_q - \varepsilon\|x\|_p, \|x\|_q + \varepsilon\|x\|_p]$ for any x satisfying $Ax = z$. Assume for contradiction that it is not possible. Then there exist x and y such that $Ax = Ay$ but $\|x\|_q - \varepsilon\|x\|_p > \|y\|_q + \varepsilon\|y\|_p$. However, since $x - y$ is in the kernel of A,

$$\|x\|_q - \|y\|_q \leq \|x - y\|_q \leq \varepsilon\|x - y\|_p \leq \varepsilon(\|x\|_p + \|y\|_p)$$

Thus, we have a contradiction. The above argument also shows that given the sketch z, the output procedure can return $\min_{x:Ax=z} \|x\|_q + \varepsilon\|x\|_p$. This is a convex optimization problem that can be solved using the ellipsoid algorithm. Below we give the details of the algorithm for finding a $1 + \varepsilon$ approximation of OPT, where OPT is equal to $\min_{x:Ax=z} \|x\|_q + \varepsilon\|x\|_p$.

Let $y = A^T(AA^T)^{-1}z$. Then $Ay = z = Ax$, y is the projection of x on the space spanned by the rows of A, and thus y is the vector of minimum ℓ_2 norm satisfying $Ay = z$. We have for any x satisfying $Ax = z$,

$$\begin{aligned} n^{-1/2}\|y\|_2 \leq n^{-1/2}\|x\|_2 \leq \|x\|_q \leq OPT = \min_{x:Ax=z} \|x\|_q + \varepsilon\|x\|_p \\ \leq \|y\|_q + \varepsilon\|y\|_p \leq (1 + \varepsilon)\sqrt{n}\|y\|_2 \end{aligned} \tag{8}$$

The value $\|y\|_2$ can be computed from the sketch z, and we use this value to find OPT using binary search. Specifically, in each step we use the ellipsoid algorithm to solve the feasibility problem $\|x\|_q + \varepsilon\|x\|_p \leq M$ on the affine subspace $Ax = z$. Recall that when solving feasibility problems, the ellipsoid algorithm takes time polynomial in the dimension, the running time of a separation oracle, and the logarithm of the ratio of volumes of an initial ellipsoid containing a feasible point and the volume of the intersection of that ellipsoid with the feasible set. Let x^* be the optimal solution of the minimization problem. If $M \geq (1 + \varepsilon)OPT$ then by the triangle inequality every point in the ℓ_2 ball centered at x^* of radius $\frac{\varepsilon n^{-1}\|y\|_2}{1+\varepsilon}$ is feasible. Furthermore, by Eq. (8) the set of feasible solutions is contained in the intersection of the ℓ_2 ball about the origin of radius $(1 + \varepsilon)n\|y\|_2$ and the affine subspace (or equivalently, the ℓ_2 ball about y of radius $\sqrt{(1 + \varepsilon)^2 n^2 - 1}\|y\|_2$ and the affine subspace). Thus, the ellipsoid algorithm runs in time polynomial in n and $\log(1/\varepsilon)$ assuming a polynomial time separation oracle.

Now we describe the separation oracle. Consider a point x such that $\|x\|_q + \varepsilon\|x\|_p > M$. We want to find a hyperplane separating x and $\{y | \|y\|_q + \varepsilon\|y\|_p \le M\}$. Without loss of generality assume that $x_i \ge 0$ for all i. Define $f_{x,p,i}$ as follows:

$$f_{x,p,i} = \begin{cases} \|x\|_p^{1-p} x_i^{p-1} & \text{if } p < \infty \\ 1/k & \text{if } p = \infty \text{ and } x_i = \max_j x_j \text{ and } k = |\{t | x_t = \max_j x_j\}| \\ 0 & \text{if } p = \infty \text{ and } x_i < \max_j x_j \end{cases}.$$

The hyperplane we consider is $h \cdot y = h \cdot x$ where $h_i = f_{x,q,i} + \varepsilon f_{x,p,i}$.

Lemma 14. *If $h \cdot y \ge h \cdot x$ then $\|y\|_q + \varepsilon\|y\|_p \ge \|x\|_q + \varepsilon\|y\|_p$.*

Proof. For any y, consider y' such that $y'_i = |y_i|$. We have $\|y'\|_q + \varepsilon\|y'\|_p = \|y\|_q + \varepsilon\|y\|_p$ and $h \cdot y' \ge h \cdot y$. Thus, we only need to prove the claim for y such that $y_i \ge 0 \ \forall i$.

If $p < \infty$ then by Hölder's inequality,

$$\|y\|_p \cdot \|x\|_p^{p-1} = \|y\|_p \cdot \|(x_i^{p-1})_i\|_{p/(p-1)} \ge \sum_i y_i x_i^{p-1}.$$

If $p = \infty$ then $\|y\|_\infty \ge \sum_{i: x_i = \max_j x_j} y_i / k$.

In either case, $\|y\|_p \ge \sum_i y_i f_{x,p,i}$, and the same inequality holds for p replaced with q. Thus,

$$\|y\|_q + \varepsilon\|y\|_p \ge y \cdot h \ge x \cdot h = \|x\|_q + \varepsilon\|x\|_p.$$

□

By the above lemma, h separates x and the set of feasible solutions. This concludes the description of the algorithm.

For the lower bound, consider a matrix A with fewer than m rows. Then in the kernel of A, there exists v such that $\|v\|_q > \varepsilon\|v\|_p$. Both v and the zero vector give the same sketch (a zero vector). However, by the stated requirement, we need to output 0 for the zero vector but some positive number for v. Thus, no matrix A with fewer than m rows can solve the problem. □

The subspace S of highest dimension of $\mathbb{R}^n$ satisfying $\sup_{v \in S} \frac{\|v\|_q}{\|v\|_p} \le \varepsilon$ is related to the Gelfand width, a well-studied notion in functional analysis.

Definition 15. *Fix $p < q$. The Gelfand width of order m of ℓ_p and ℓ_q unit balls in $\mathbb{R}^n$ is defined as*

$$\inf_{\text{subspace } A: codim(A)=m} \sup_{v \in A} \frac{\|v\|_q}{\|v\|_p}$$

Using known bounds for the Gelfand width for $p = 1$ and $q = 2$, we get the following corollary.

Corollary 16. *Assume that $1/\varepsilon^2 < n/2$. There is an $m \times n$ matrix A and associated output procedure Out which for any $x \in \mathbb{R}^n$, given Ax, outputs an estimate e such that $\|x\|_2 - \varepsilon\|x\|_1 \leq e \leq \|x\|_2 + \varepsilon\|x\|_1$. Here $m = O(\varepsilon^{-2}\log(\varepsilon^2 n))$ and this bound for m is tight.*

Proof. The corollary follows from the following bound on the Gelfand width by Foucart et al. [22] and Garnaev and Gluskin [49]:

$$\inf_{\text{subspace } A: \text{codim}(A)=m} \sup_{v \in A} \frac{\|v\|_2}{\|v\|_1} = \Theta\left(\sqrt{\frac{1 + \log(n/m)}{m}}\right)$$

□

Acknowledgments

We thank Raghu Meka for answering several questions about almost k-wise independent sample spaces. We thank an anonymous reviewer for pointing out the connection between incoherent matrices and ε-biased spaces, which are used to construct almost k-wise independent sample spaces.

References

[1] D. Barbará, N. Wu, S. Jajodia, in: Proceedings of the 1st SIAM International Conference on Data Mining.

[2] E. D. Demaine, A. López-Ortiz, J. I. Munro, in: ESA, pp. 348–360.

[3] A. C. Gilbert, Y. Kotidis, S. Muthukrishnan, M. J. Strauss, Quicksand: Quick summary and analysis of network data, DIMACS Technical Report 2001-43, 2001.

[4] R. M. Karp, S. Shenker, C. H. Papadimitriou, ACM Trans. Database Syst. 28 (2003) 51–55.

[5] J. Misra, D. Gries, Sci. Comput. Program. 2 (1982) 143–152.

[6] G. Cormode, S. Muthukrishnan, ACM Trans. Database Syst. 30 (2005) 249–278.

[7] G. Cormode, S. Muthukrishnan, J. Algorithms 55 (2005) 58–75.

[8] M. Charikar, K. Chen, M. Farach-Colton, Theor. Comput. Sci. 312 (2004) 3–15.

[9] S. Ganguly, in: COCOA, pp. 301–312.

[10] A. Cohen, W. Dahmen, R. A. DeVore, J. Amer. Math. Soc. 22 (2009) 211–231.

[11] W. B. Johnson, J. Lindenstrauss, Contemporary Mathematics 26 (1984) 189–206.

[12] N. Alon, O. Goldreich, J. Håstad, R. Peralta, Random Struct. Algorithms 3 (1992) 289–304.

[13] J. Naor, M. Naor, SIAM J. Comput. 22 (1993) 838–856.

[14] N. Alon, Discrete Mathematics 273 (2003) 31–53.

[15] W. U. Bajwa, R. Calderbank, D. G. Mixon, Appl. Comput. Harmon. Anal. 33 (2012) 58–78.

[16] D. L. Donoho, X. Huo, IEEE Trans. Inform. Th. 47 (2001) 2558–2567.

[17] S. G. Mallat, Z. Zhang, IEEE Trans. Signal Process. 41 (1993) 3397–3415.

[18] A. C. Gilbert, S. Muthukrishnan, M. Strauss, in: SODA, pp. 243–252.

[19] S. Ganguly, A. Majumder, in: ESCAPE, pp. 48–59.

[20] N. Alon, Combinatorics, Probability & Computing 18 (2009) 3–15.

[21] V. I. Levenshtein, Problemy Kibernet (1983) 43–110.

[22] S. Foucart, A. Pajor, H. Rauhut, T. Ullrich, Journal of Complexity 26 (2010) 629–640.

[23] S. Ganguly, in: CSR, pp. 204–215. Full version at http://www.cse.iitk.ac.in/users/sganguly/csr-full.pdf.

[24] E. D. Gluskin, Vestn. Leningr. Univ. Math. 14 (1982) 163–170.

[25] N. Ailon, B. Chazelle, SIAM J. Comput. 39 (2009) 302–322.

[26] N. Ailon, E. Liberty, Discrete & Computational Geometry 42 (2009) 615–630.

[27] N. Ailon, E. Liberty, in: Proceedings of the 22nd Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 185–191.

[28] F. Krahmer, R. Ward, SIAM J. Math. Anal. 43 (2011) 1269–1281.

[29] H. Jowhari, M. Saglam, G. Tardos, in: PODS, pp. 49–58.

[30] P. Indyk, M. Ružić, in: FOCS, pp. 199–207.

[31] E. Price, D. P. Woodruff, in: FOCS, pp. 295–304.

[32] K. D. Ba, P. Indyk, E. Price, D. P. Woodruff, in: SODA, pp. 1190–1197.

[33] N. Alon, Y. Matias, M. Szegedy, JCSS 58 (1999) 137–147.

[34] D. Achlioptas, J. Comput. Syst. Sci. 66 (2003) 671–687.

[35] D. Sivakumar, in: STOC, pp. 619–626.

[36] A. R. Calderbank, S. D. Howard, S. Jafarpour, J. Sel. Topics Signal Processing 4 (2010) 358–374.

[37] A. Ben-Aroya, A. Ta-Shma, in: FOCS, pp. 191–197.

[38] H. Krishna, B. Krishna, K.-Y. Lin, J.-D. Sun, Computational Number Theory and Digital Signal Processing: Fast Algorithms and Error Control Techniques, CRC, Boca Raton, FL, 1994.

[39] M. A. Soderstrand, W. K. Jenkins, G. A. Jullien, F. J. Taylor, Residue Number System Arithmetic: Modern Applications in Digital Signal Processing, IEEE Press, New York, 1986.

[40] R. W. Watson, C. W. Hastings, Proc. IEEE 4 (1966) 1920–1931.

[41] W. H. Kautz, R. C. Singleton, IEEE Trans. Inf. Theory 10 (1964) 363–377.

[42] D. M. Kane, J. Nelson, in: SODA, pp. 1195–1206.

[43] J. von zur Gathen, J. Gerhard, Modern Computer Algebra, Cambridge University Press, 1999.

[44] A. C. Gilbert, M. J. Strauss, J. A. Tropp, R. Vershynin, in: STOC, pp. 237–246.

[45] R. Baraniuk, M. A. Davenport, R. DeVore, M. Wakin, Constructive Approximation 28 (2008) 253–263.

[46] E. Candès, J. Romberg, T. Tao, IEEE Trans. Information Theory 52 (2006) 489–509.

[47] M. Rudelson, R. Vershynin, Communications on Pure and Applied Mathematics 61 (2008) 1025–1045.

[48] E. Kushilevitz, N. Nisan, Communication complexity, Cambridge University Press, 1997.

[49] A. Y. Garnaev, E. D. Gluskin, Soviet Mathematics Doklady 30 (1984) 200–203.