A Simple Proof of a New Set Disjointness with Applications to Data Streams

Akshay Kamath
University of Texas at Austin, TX, USA

Eric Price
University of Texas at Austin, TX, USA

David P. Woodruff
Carnegie Mellon University, Pittsburgh, PA, USA

Abstract

The multiplayer promise set disjointness is one of the most widely used problems from communication complexity in applications. In this problem there are k players with subsets $S^1, \ldots, S^k$, each drawn from $\{1, 2, \ldots, n\}$, and we are promised that either the sets are (1) pairwise disjoint, or (2) there is a unique element j occurring in all the sets, which are otherwise pairwise disjoint. The total communication of solving this problem with constant probability in the blackboard model is $\Omega(n/k)$.

We observe for most applications, it instead suffices to look at what we call the "mostly" set disjointness problem, which changes case (2) to say there is a unique element j occurring in at least half of the sets, and the sets are otherwise disjoint. This change gives us a much simpler proof of an $\Omega(n/k)$ randomized total communication lower bound, avoiding Hellinger distance and Poincare inequalities. Our proof also gives strong lower bounds for high probability protocols, which are much larger than what is possible for the set disjointness problem. Using this we show several new results for data streams:

1. for ℓ_2-Heavy Hitters, any $O(1)$-pass streaming algorithm in the insertion-only model for detecting if an ε-ℓ_2-heavy hitter exists requires $\min(\frac{1}{\varepsilon^2} \log \frac{\varepsilon^2 n}{\delta}, \frac{1}{\varepsilon} n^{1/2})$ bits of memory, which is optimal up to a $\log n$ factor. For deterministic algorithms and constant ε, this gives an $\Omega(n^{1/2})$ lower bound, improving the prior $\Omega(\log n)$ lower bound. We also obtain lower bounds for Zipfian distributions.
2. for ℓ_p-Estimation, $p > 2$, we show an $O(1)$-pass $\Omega(n^{1-2/p} \log(1/\delta))$ bit lower bound for outputting an $O(1)$- approximation with probability $1 - \delta$, in the insertion-only model. This is optimal, and the best previous lower bound was $\Omega(n^{1-2/p} + \log(1/\delta))$.
3. for low rank approximation of a sparse matrix in $\mathbb{R}^{d\times n}$, if we see the rows of a matrix one at a time in the row-order model, each row having $O(1)$ non-zero entries, any deterministic algorithm requires $\Omega(\sqrt{d})$ memory to output an $O(1)$-approximate rank-1 approximation.

Finally, we consider strict and general turnstile streaming models, and show separations between sketching lower bounds and non-sketching upper bounds for the heavy hitters problem.

2012 ACM Subject Classification Theory of computation → Lower bounds and information complexity

Keywords and phrases Streaming algorithms, heavy hitters, communication complexity, information complexity

Digital Object Identifier 10.4230/LIPIcs.CCC.2021.37

Funding *Eric Price*: NSF grant CCF-1751040 (CAREER).
David P. Woodruff: NSF grant No. CCF-1815840 and a Simons Investigator Award.

Acknowledgements The authors would like to thank the anonymous reviewers of a previous version of this paper for helpful suggestions that significantly improved the presentation.

1 Introduction

Communication complexity is a common technique for establishing lower bounds on the resources required of problems, such as the memory required of a streaming algorithm. The multiplayer promise set disjointness is one of the most widely used problems from communication complexity in applications, not only in data streams [3, 5, 18, 39, 48, 49, 19], but also in compressed sensing [67], distributed functional monitoring [77, 78], distributed learning [32, 52, 11], matrix-vector query models [71], voting [60, 61], and so on. We shall restrict ourselves to the study of set disjointness in the number-in-hand communication model, described below, which covers all of the above applications. Set disjointness is also well-studied in the number-on-forehead communication model, see, e.g., [38, 72, 7, 56, 21, 6, 69, 70], though we will not discuss that model here.

In the number-in-hand multiplayer promise set disjointness problem there are k players with subsets $S^1, \ldots, S^k$, each drawn from $\{1, 2, \ldots, n\}$, and we are promised that either:

1. the S^i are pairwise disjoint, or
2. there is a unique element j occurring in all the sets, which are otherwise pairwise disjoint.

The promise set disjointness problem was posed by Alon, Matias, and Szegedy [3], who showed an $\Omega(n/k^4)$ total communication bound in the blackboard communication model, where each player's message can be seen by all other players. This total communication bound was then improved to $\Omega(n/k^2)$ by Bar-Yossef, Jayram, Kumar, and Sivakumar [5], who further improved this bound to $\Omega(n/k^{1+\gamma})$ for an arbitrarily small constant $\gamma > 0$ in the one-way model of communication. These bounds were further improved by Chakrabarti, Khot, and Sun to $\Omega(n/(k \log k))$ in the general communication model and an optimal $\Omega(n/k)$ bound for 1-way communication. The optimal $\Omega(n/k)$ total communication bound for general communication was finally obtained in [39, 49].

To illustrate a simple example of how this problem can be used, consider the *streaming model*. The streaming model is one of the most important models for processing massive datasets. One can model a stream as a list of integers $i_1, \ldots, i_m \in [n] = \{1, 2, \ldots, n\}$, where each item $i \in [n]$ has a frequency x_i which denotes its number of occurrences in the stream. We refer the reader to [4, 66] for further background on the streaming model of computation.

An important problem in this model is computing the p-th frequency moment $F_p = \sum_{j=1}^{n} x_j^p$. To reduce from the promise set disjointness problem, the first player runs a streaming algorithm on the items in its set, passes the state of the algorithm to the next player, and so on. The total communication is $k \cdot s$, where s is the amount of memory of the streaming algorithm. Observe that in the first case of the promise we have $F_p \leq n$, while in the second case we have $F_p \geq k^p$. Setting $k = (2n)^{1/p}$ therefore implies an algorithm estimating F_p up to a factor better than 2 can solve promise set disjointness and therefore $k \cdot s = (2n)^{1/p} s = \Omega(n/(2n)^{1/p})$, that is, $s = \Omega(n^{1-2/p})$. For $p > 2$, this is known to be best possible up to a constant factor [14].

Notice that nothing substantial would change in this reduction if one were to change the second case in the promise to instead say: (2) there is a unique element j occurring in at least half of the sets, and the sets are otherwise disjoint. Indeed, in the above reduction, in one case we have $F_p \geq (k/2)^p$, while in the second case we have $F_p \leq n$. This recovers the same $\Omega(n^{1-2/p})$ lower bound, up to a constant factor. We call this new problem "mostly" set disjointness (MostlyDISJ).

While it is seemingly inconsequential to consider MostlyDISJ instead of promise set disjointness, there are some peculiarities about this problem that one cannot help but wonder about. In the promise set disjointness problem, there is a deterministic protocol solving the problem with $O(n/k \log k + k)$ bits of communication – we walk through the players one at a time, and each indicates if its set size is smaller than n/k. Eventually we must reach such a player, and when we do, that player posts its set to the blackboard. We then ask one other player to confirm an intersection. Notice that there always must exist a player with a set of size at most n/k by the pigeonhole principle. On the other hand, for the MostlyDISJ problem, it does not seem so easy to achieve a deterministic protocol with $O(n/k \log k + k)$ bits of communication. Indeed, in the worst case we could have up to $k/2$ players posting their entire set to the blackboard, and still be unsure if we are in Case (1) or Case (2).

More generally, is there a gap in the dependence on the error probability of algorithms for promise set disjointness versus MostlyDISJ? Even if one's main interest is in constant error probability protocols, is there anything that can be learned from this new problem?

1.1 Our Results

We more generally define MostlyDISJ so that in Case (2), there is an item occurring in $l = \Theta(k)$ of the sets, though it is still convenient to think of $l = k/2$. Our main theorem is that MostlyDISJ requires $\Omega(n)$ communication to solve deterministically, or even with failure probability e^{-k}.

▶ **Theorem 1.** MostlyDISJ *with n elements, k players, and $l = ck$ for an absolute constant $c \in (0, 1)$ requires $\Omega(\min(n, n\frac{\log(1/\delta)}{k}))$ bits of communication for failure probability δ.*

This result does not have any restriction on the order of communication, and is in the "blackboard model" where each message is visible to all other players. We note that as $c \to 1$, our lower bound goes to 0, but for any absolute constant $c \in (0, 1)$, we achieve the stated $\Omega(\min(n, n\frac{\log(1/\delta)}{k}))$ lower bound. We did not explicitly compute our lower bound as a function of c, as $c \to 1$.

Notice that for constant δ, Theorem 1 recovers the $\Omega(n/k)$ total communication bound for promise set disjointness, which was the result of a long sequence of work. Our proof of Theorem 1 gives a much simpler proof of an $\Omega(n/k)$ total communication lower bound, avoiding Hellinger distance and Poincare inequalities altogether, which were the main ingredients in obtaining the optimal $\Omega(n/k)$ lower bound for promise set disjointness in previous work. Moreover, as far as we are aware, an $\Omega(n/k)$ lower bound for the MostlyDISJ problem suffices to recover all of the lower bounds in applications that promise set disjointness has been applied to. Unlike our work, however, existing lower bounds for promise set disjointness do not give improved bounds for small error probability δ. Indeed, it is impossible for them to do so because of the deterministic protocol described above. We next use this bound in terms of δ to obtain the first lower bounds for deterministic streaming algorithms and randomized δ-error algorithms for a large number of problems.

We note that other work on deterministic communication lower bounds for streaming, e.g., the work of Chakrabarti and Kale [17], does not apply here. They study multi-party equality problems and it is not clear how to use their fooling set arguments to prove a lower bound for MostlyDISJ. One of the challenges in designing a fooling set is the promise, namely, that a single item occurs on a constant fraction of the players *and* all remaining items occur on at most one player. This promise is crucial for the applications of MostlyDISJ.

We now formally introduce notation for the data stream model. In the streaming model, an integer vector x is initialized to 0^n and undergoes a sequence of $L = \mathrm{poly}(n)$ updates. The streaming algorithm is typically allowed one (or a few) passes over the stream, and the goal is to use a small amount of memory. We cannot afford to store the entire stream since n and L are typically very large. In this paper, we mostly restrict our focus to the *insertion-only* *model* where the updates to the vector are of the form $x \leftarrow x + \delta$ where $\delta \in \{e_1, \ldots, e_n\}$ is a standard basis vector. There are also the turnstile data stream models in which $x \leftarrow x + \delta$ where $\delta \in \{e_1, \ldots, e_n, -e_1, \ldots, -e_n\}$. In the *strict turnstile model* it is promised that $x \geq 0^n$ at all times in the stream, whereas in the *general turnstile model* there are no restrictions on x. Therefore, an algorithm in the general turnstile model works also in the strict turnstile model and insertion-only models.

Finding Heavy Hitters

Finding the heavy hitters, or frequent items, is one of the most fundamental problems in data streams. These are useful in IP routers [29], in association rules and frequent itemsets [1, 68, 73, 44, 42] and databases [30, 9, 41]. Finding the heavy hitters is also frequently used as a subroutine in data stream algorithms for other problems, such as moment estimation [46], entropy estimation [16, 43], ℓ_p-sampling [65], finding duplicates [37], and so on. For surveys on algorithms for heavy hitters, see, e.g., [25, 76].

In the *ϵ-ℓ_p-heavy hitters problem*, for $p \geq 1$, the goal is to find a set S which contains all indices $i \in [n]$ for which $|x_i|^p \geq \epsilon^p \|x\|_p^p$, and contains no indices $i \in [n]$ for which $|x_i|^p \leq \frac{\epsilon^p}{2} \|x\|_p^p$.

The first heavy hitters algorithms were for $p = 1$, given by Misra and Gries [64], who achieved $O(\epsilon^{-1})$ words of memory, where a word consists of $O(\log n)$ bits of space. Interestingly, their algorithm is *deterministic*, i.e., the failure probability $\delta = 0$. This algorithm was rediscovered by Demaine, López-Ortiz, and Munro [27], and again by Karp, Shenker, and Papadimitriou [53]. Other than these algorithms, which are deterministic, there are a number of randomized algorithms, such as the Count-Min sketch [26], sticky sampling [62], lossy counting [62], space-saving [63], sample and hold [29], multi-stage bloom filters [15], and sketch-guided sampling [54]. One can also achieve stronger residual error guarantees [8].

An often much stronger notion than an ℓ_1-heavy hitter is an ℓ_2-heavy hitter. Consider an n-dimensional vector $x = (\sqrt{n}, 1, 1, \ldots, 1)$. The first coordinate is an ℓ_2-heavy hitter with parameter $\epsilon = 1/\sqrt{2}$, but it is only an ℓ_1 heavy hitter with parameter $\epsilon = 1/\sqrt{n}$. Thus, the algorithms above would require at least $\sqrt{n}$ words of memory to find this heavy hitter. In [20] this problem was solved by the CountSketch algorithm, which provides a solution to the ϵ-ℓ_2-heavy hitters problem, and more generally to the ℓ_p-heavy hitters problem for any $p \in (0, 2]$[1], in 1-pass and in the general turnstile model using $O\left(\frac{1}{\epsilon^p} \log(n/\delta)\right)$ words of memory. For insertion-only streams, this was recently improved [13, 12] to $O\left(\frac{1}{\epsilon^p}\right)$ words of memory for constant δ, and $O\left(\frac{1}{\epsilon^p} \log(1/\delta)\right)$ in general. See also work [55] on reducing the decoding time for finding the heavy hitters from the algorithm's memory contents, without sacrificing additional memory.

There is also work establishing lower bounds for heavy hitters. The works of [28, 50] establish an $\Omega\left(\frac{1}{\epsilon^p} \log n\right)$ word lower bound for any value of $p > 0$ and constant δ, for any algorithm in the strict turnstile model. This shows that the above algorithms are optimal for constant δ. Also for $p > 2$, it is known that solving the ϵ-ℓ_p-heavy hitters problem even with constant ϵ and δ requires $\Omega(n^{1-2/p})$ words of memory [5, 39, 49], and thus $p = 2$ is often considered the gold standard for space-efficient streaming algorithms since it is the largest value of p for which there is a poly$(\log n)$ space algorithm. For deterministic algorithms computing linear sketches, the work of [31] shows the sketch requires $\Omega(n^{2-2/p}/\epsilon^2)$ dimensions for $p \geq 1$ (also shown for $p = 2$ by [23]). This also implies a lower bound for general turnstile algorithms for streams with several important restrictions; see also [57, 51]. There is also work on the related compressed sensing problem which studies small δ [36].

[1] For $p < 1$ the quantity $\|x\|_p$ is not a norm, but it is still a well-defined quantity.

Despite the work above, for all we knew it could be entirely possible that, in the insertion-only model, an ϵ-ℓ_2-heavy hitters algorithm could achieve $O\left(\frac{1}{\epsilon^2}\right)$ words of memory and solve the problem *deterministically*, i.e., with $\delta = 0$. In fact, it is well-known that the above $\Omega(n)$ lower bound for ϵ-ℓ_2-heavy hitters for linear sketches does not hold in the insertion-only model. Indeed, by running a deterministic algorithm for ϵ-ℓ_1-heavy hitters, we have that if $x_i^2 \geq \epsilon^2 \|x\|_2^2$, then $x_i \geq \epsilon \|x\|_2 \geq \frac{\epsilon}{\sqrt{n}} \|x\|_1$, and consequently one can find all ℓ_2-heavy hitters using $O\left(\frac{\sqrt{n}}{\epsilon}\right)$ words of memory. Thus, for constant ϵ, there is a deterministic $O\left(\sqrt{n}\right)$ words of memory upper bound, but only a trivial $\Omega\left(1\right)$ word lower bound. Surprisingly, this factor $\sqrt{n}$ gap was left wide open, and the main question we ask about heavy hitters is:

Can one deterministically solve ϵ-ℓ_2-heavy hitters in insertion-only streams in constant memory?

One approach to solve MostlyDISJ would be for each player to insert their elements into a stream and apply a heavy hitters algorithm. For example, if $k = \sqrt{n}$, there will be a $\Theta(1)$-ℓ_2-heavy hitter if and only if the MostlyDISJ instance is a YES instance. For a space-S streaming algorithm, this uses Sk communication to pass the structure from player to player. Hence $S \gtrsim n/k = \sqrt{n}$. In general:

▶ **Theorem 21.** *Given $\varepsilon \in (\frac{1}{n^{1/p}}, \frac{1}{2})$ and $p \geq 1$, any δ-error r-pass insertion-only streaming algorithm for ε-ℓ_p-heavy hitters requires $\Omega(\min(\frac{n^{1-1/p}}{r\varepsilon}, n^{1-2/p}\frac{\log(1/\delta)}{r\varepsilon^2}))$ bits of space.*

Most notably, setting $\delta = 0$ and $p = 2$ and $r = O(1)$, this gives an $\Omega(\sqrt{n}/\varepsilon)$ bound for deterministic ℓ_2 heavy hitters. The FrequentElements algorithm [64] matches this up to a factor of $\log n$ (i.e., it uses this many words, not bits). For $n^{-.1} > \delta > 0$, the other term $(\frac{\log(1/\delta)}{r\varepsilon^2})$ is also achievable up to the bit/word distinction, this time by CountSketch. For larger δ, we note that it takes $\Omega(\frac{1}{\varepsilon^2}\log \varepsilon^2 n)$ bits already to encode the output size. As a result, we show that the existing algorithms are within a $\log n$ factor of optimal.

One common motivation for heavy hitters is that many distributions are power-law or Zipfian distributions. For such distributions, the i-th most frequent element has frequency approximately proportional to $i^{-\zeta}$ for some constant ζ, typically $\zeta \in (0.5, 1)$ [22]. Such distributions have significant $\ell_{1/\zeta}$-heavy hitters. Despite our lower bound for general heavy hitters, one might hope for more efficient deterministic/very high probability insertion-only algorithms in this special case. We rule this out as well, getting an $\Omega(\min(n^{1-\zeta}, n^{1-2\zeta}\log(1/\delta)))$ lower bound for finding the heavy hitters of these distributions (see Theorem 24). This again matches the upper bounds from FrequentElements or CountSketch up to a logarithmic factor.

To extend our lower bound to power-law distributions, we embed our hard instance as the single largest and $n/2$ smallest entries of a power-law distribution; we then insert the rest of the power-law distribution deterministically, so the overall distribution is power-law distributed. Solving heavy hitters will identify whether this single largest element exists or not, solving the communication problem.

Frequency Moments

We next turn to the problem of estimating the frequency moments F_p, which in our reduction from the MostlyDISJ problem, just corresponds to estimating $\|x\|_p^p = \sum_{i=1}^n |x_i|^p$. Our hard instance for MostlyDISJ immediately gives us the following theorem:

▶ **Theorem 2.** *For any constant $\epsilon \in (0,1)$ and $p \geq 2$, any δ-error r-pass insertion-only streaming algorithm for ε-F_p-estimation must have space complexity of $\Omega(\min(\frac{n^{1-1/p}}{r}, n^{1-2/p}\frac{\log(1/\delta)}{r}))$ bits.*

The proof of Theorem 2 follows immediately by setting the number of players in MostlyDISJ to be $\Theta((\epsilon n)^{1/p})$, and performing the reduction to F_p-estimation described before Section 1.1. This improves the previous $\Omega((n^{1-2/p} + \log(1/\delta))/r)$ lower bound, which follows from [5, 49], as well as a simple reduction from the Equality function [3], see also [17]. It matches an upper bound of [14] for constant ϵ, by repeating their algorithm independently $O(\log(1/\delta))$ times. Our lower bound instance shows that to approximate $\|x\|_\infty = \max_i |x_i|$ of an integer vector, with $O(\log n)$-bit coordinates in n dimensions, up to an additive $\Theta(\sqrt{\|x\|_2})$ deterministically, one needs $\Omega(\sqrt{n})$ memory. This follows from our hard instance. Approximating the ℓ_∞ norm is an important problem in streaming, and its complexity was asked about in Question 3 of [24].

Low Rank Approximation

Our ℓ_2-heavy hitters lower bound also has applications to *deterministic* low rank approximation in a stream, a topic of recent interest [59, 35, 75, 34, 33, 45]. Here we see rows $A_1, A_2, \ldots, A_n$ of an $n \times d$ matrix A one at a time. At the end of the stream we should output a projection P onto a rank-k space for which $\|A - AP\|_F^2 \leq (1+\epsilon)\|A - A_k\|_F^2$, where A_k is the best rank-k approximation to A. A natural question is if the deterministic FrequentDirections algorithm of [34] using $O(dk/\epsilon)$ words of memory can be improved when the rows of A are $O(1)$-sparse. The sparse setting was shown to have faster running times in [33, 45], and more efficient randomized communication protocols in [10]. Via a reduction from our MostlyDISJ problem, we show a polynomial dependence on d is necessary.

▶ **Theorem 25.** *Any 1-pass deterministic streaming algorithm outputting a rank-k projection matrix P providing a $(1+\epsilon)$-approximate rank-k low rank approximation requires $\Omega(\sqrt{d})$ bits of memory, even for $k = 1$, $\epsilon = \Theta(1)$, and when each row of A has only a single non-zero entry.*

Algorithms and Lower Bounds in Other Streaming Models

We saw above that deterministic insertion-only ℓ_2 heavy hitters requires $\widetilde{\Theta}(\sqrt{n})$ space for constant ε. We now consider turnstile streaming and linear sketching.

The work of [31, 23] shows that $\Omega(n)$ space is needed for general deterministic linear sketching, but the corresponding hard instances have negative entries. We extend this in two ways: when negative entries are allowed, an $\Omega(n)$ lower bound is easy even in turnstile streaming (for heavy hitters, but not the closely related ℓ_∞/ℓ_2 sparse recovery guarantee; see Remark 27). If negative entries are not allowed, we still get an $\Omega(n)$ bound on the number of linear measurements for deterministic linear sketching (see Theorem 20).

A question is if we can solve ℓ_2 heavy hitters deterministically in the strict turnstile model in $o(n)$ space. In some sense the answer is no, due to the near equivalence between turnstile streaming and linear sketching [31, 58, 2], but this equivalence has significant limitations. Recent work has shown that with relatively mild restrictions on the stream, such as a bound on the length L, significant improvements over linear sketching are possible [47, 51]. Can we get that here? We show that this is indeed possible: streams with $O(n)$ updates can be solved in $O(n^{2/3})$ space. While this does not reach the $\sqrt{n}$ lower bound from insertion-only streams (Theorem 22), it is significantly better than the $\Omega(n)$ for linear sketches. In general, we show:

▶ **Theorem 26.** *There is a deterministic ℓ_2 heavy hitters algorithm for length-L strict turnstile streams with ± 1 updates using $O((L/\varepsilon)^{2/3})$ words of space.*

Our algorithm for short strict turnstile streams is a combination of FrequentElements and exact sparse recovery. With space S, FrequentElements (modified to handle negative updates) gives estimation error L/S, which is good unless $\|x\|_2 \ll L/S$. But if it is not good, then $\|x\|_0 \leq \|x\|_2^2 \ll (L/S)^2$. Hence in that case $(L/S)^2$-sparse recovery will recover the vector (and hence the heavy hitters). Running both algorithms and combining the results takes $S + (L/S)^2$ space, which is optimized at $L^{2/3}$.

1.2 Our Techniques

Our key lemma is that solving MostlyDISJ on n elements, k items, and $l = ck$ with probability $1 - e^{-k}$ has $\Omega(n)$ conditional information complexity for any constant $c \in (0, 1)$. It is well-known that the conditional information complexity of a problem lower bounds its communication complexity (see, e.g., [5]).

This can then be extended to $\delta \gg e^{-\Theta(k)}$ using repetition, namely, we can amplify the success probability of the protocol to $1 - e^{-\Theta(k)}$ by independent repetition, apply our $\Omega(n)$ lower bound on the new protocol with $\delta = e^{-\Theta(k)}$, and then conclude a lower bound on the original protocol. Indeed, this is how we obtain our total communication lower bound of $\Omega(n/k)$ for constant δ, providing a much simpler proof than that of the $\Omega(n/k)$ total communication lower bound for promise set disjointness in prior work.

Our bound is tight up to a $\log k$ factor. It can be solved deterministically with $O(n \log k)$ communication (for each bit, the first player with that bit publishes it), and with probability $1 - (1 - \varepsilon)^{l-1}$ using $O(\varepsilon n \log k)$ communication (only publish the bit with probability ε). Setting $\varepsilon = o(1)$, any $e^{-o(k)}$ failure probability is possible with $o(n \log k)$ communication.

We lower bound MostlyDISJ using conditional information complexity. Using the direct sum property of conditional information cost, analogous to previous work (see, e.g., [5]), it suffices to get an $\Omega(1)$ conditional information cost bound for the $n = 1$ problem F_k: we have k players, each of whom receives one bit, and the players must distinguish (with probability $1 - e^{-k}$) between at most one player having a 1, and at least $\Omega(k)$ players having 1s. In particular, it suffices to show for correct protocols π that

$$\mathop{\mathbb{E}}_{i \in [t]} d_{\mathrm{TV}}(\pi_0, \pi_{e_i}) = \Omega(1) \tag{1}$$

where π_0 is the distribution of protocol transcripts if the players all receive 0, and π_{e_i} is the distribution if player i receives a 1. The main challenge is therefore in bounding this expression.

Consider any protocol that does not satisfy (1). We show that, when $d_{\mathrm{TV}}(\pi_0, \pi_{e_i}) \ll 1$, player i can be implemented with an equivalent protocol for which the player usually does not even observe its input bit. That is, if every other player receives a 0, player i will only observe its bit with probability $d_{\mathrm{TV}}(\pi_0, \pi_{e_i})$. This means that most players only have a small probability of observing their bit. The probability that any two players i, i' observe their bits may be correlated; still, we show that this implies the existence of a large set S of ck players such that the probability – if every player receives a zero – that *no* player $i \in S$ observes their bit throughout the protocol is above e^{-k}. But then $d_{\mathrm{TV}}(\pi_0, \pi_{e_S}) < 1 - e^{-k}$, so the protocol cannot distinguish these cases with the desired probability. We now give the full proof.

2 Preliminaries

We use the following measures of distance between distributions in our proofs.

▶ **Definition 3.** *Let P and Q be probability distributions over the same countable universe $\mathcal{U}$. The total variation distance between P and Q is defined as: $d_{TV}(P,Q) = \frac{1}{2}\|P-Q\|_1$.*

In our proof we also use the Jensen-Shannon divergence and Kullback-Liebler divergence. We define these notions of divergence here:

▶ **Definition 4.** *Let P and Q be probability distributions over the same discrete universe $\mathcal{U}$. The Kullback-Liebler divergence or KL-divergence from Q to P is defined as: $D_{KL}(P,Q) = \sum_{x\in\mathcal{U}} P(x)\log(\frac{P(x)}{Q(x)})$. This is an asymmetric notion of divergence. The Jensen-Shannon divergence between two distributions P and Q is the symmetrized version of the KL divergence, defined as: $D_{JS}(P,Q) = \frac{1}{2}(D_{KL}(P,Q) + D_{KL}(Q,P))$.*

From Pinsker's inequality, for any two distributions P and Q, $D_{KL}(P,Q) \geq \frac{1}{2}d^2_{\mathrm{TV}}(P,Q)$.

In the multiparty communication model we consider k-ary functions $F : \mathcal{L} \to \mathcal{Z}$ where $\mathcal{L} \subseteq \mathcal{X}_1 \times \mathcal{X}_2 \times \cdots \times \mathcal{X}_k$. There are k parties(or players) who receive inputs $X_1, \ldots, X_k$ which are jointly distributed according to some distribution μ. We consider protocols in the blackboard model where in any protocol π players speak in any order and each player broadcasts their message to all other players. So, the message of player i is a function of the messages they receive, their input and randomness i.e., $m_i = M_i(X_i, m_{i-1}, R_i)$. The final player's message is the output of the protocol.

The communication cost of a multiparty protocol π is the sum of the lengths of the individual messages $\|\pi\| = \sum |M_j|$. A protocol π is a δ-error protocol for the function f if for every input $x \in \mathcal{L}$, the output of the protocol equals $f(x)$ with probability $1-\delta$. The randomized communication complexity of f, denoted $R_\delta(f)$, is the cost of the cheapest randomized protocol that computes f correctly on every input with error at most δ over the randomness of the protocol.

The distributional communication complexity of the function f for error parameter δ is denoted as $D^\delta_\mu(f)$. This is the communication cost of the cheapest deterministic protocol which computes the function f with error at most δ under the input distribution μ. By Yao's minimax theorem, $R_\delta(f) = \max_\mu D^\delta_\mu(f)$ and hence it suffices to prove a lower bound for a hard distribution μ. In our proofs, we bound the conditional information complexity of a function in order to prove lower bounds on $R_\delta(f)$. We define this notion below.

▶ **Definition 5.** *Let π be a randomized protocol whose inputs belong to $\mathcal{K} \subseteq \mathcal{X}_1 \times \mathcal{X}_2 \ldots \times \mathcal{X}_k$. Suppose $((X_1, X_2, \ldots, X_k), D) \sim \eta$ where η is a distribution over $\mathcal{K} \times \mathcal{D}$ for some set $\mathcal{D}$. The **conditional information cost** of π with respect to η is defined as: $cCost_\eta(\pi) = I(X_1, \ldots, X_k; \pi(X_1, \ldots, X_k) \mid D)$.*

▶ **Definition 6.** *The δ-error **conditional information complexity** of f with respect to η, denoted $CIC_{\eta,\delta}(f)$ is defined as the minimum conditional information cost of a δ-error protocol for f with respect to η.*

In [5] it was shown that the randomized communication complexity of a function is at least the conditional information complexity of the function f with respect to any input distribution η.

▶ **Proposition 7** (Corollary 4.7 of [5])**.** *Let $f : \mathcal{K} \to \{0,1\}$, and let η be a distribution over $\mathcal{K} \times \mathcal{D}$ for some set $\mathcal{D}$. Then, $R_\delta(f) \geq CIC_{\eta,\delta}(f)$.*

Direct Sum

Per [5], conditional information complexity obeys a Direct Sum Theorem condition under various conditions. The Direct Sum Theorem of [5] allows us to reduce a t-player conditional information complexity problem with an n-dimensional input to each player to a t-player conditional information complexity with a 1-dimensional input to each player. This theorem applies when the function is "decomposable" and the input distribution is "collapsing". We define both these notions here.

▶ **Definition 8.** *Suppose $\mathcal{L} \subseteq \mathcal{X}_1 \times \mathcal{X}_2 \times \ldots \times \mathcal{X}_t$ and $\mathcal{L}_n \subseteq \mathcal{L}^n$. A function $f : \mathcal{L}_n \to \{0,1\}$ is g-**decomposable** with primitive $h : \mathcal{L} \to \{0,1\}$ if it can be written as:*

$$f(X_1, \ldots, X_t) = g(h(X_{1,1}, \ldots, X_{1,t}), \ldots, h(X_{n,1}, \ldots, X_{n,t}))$$

for $g : \{0,1\}^n \to \{0,1\}$.

▶ **Definition 9.** *Suppose $\mathcal{L} \subseteq \mathcal{X}_1 \times \mathcal{X}_2 \times \ldots \times \mathcal{X}_t$ and $\mathcal{L}_n \subseteq \mathcal{L}^n$. A distribution η over $\mathcal{L}_n$ is a **collapsing distribution** for $f : \mathcal{L}_n \to \{0,1\}$ with respect to $h : \mathcal{L} \to \{0,1\}$ if for all $Y_1, \ldots, Y_n$ in the support of η, for all $y \in \mathcal{L}$ and for all $i \in [n]$, $f(Y_1, \ldots, Y_{i-1}, y, Y_{i+1}, \ldots, Y_n) = h(y)$.*

We state the Direct Sum Theorem for conditional information complexity below. The proof of this theorem in [5] applies to the blackboard model of multiparty communication. We state this in the most general form here and then show that it may be applied to the hard distribution η_0 which we choose in Section 3.

▶ **Theorem 10** (Multiparty version of Theorem 5.6 of [5]). *Let $\mathcal{L} \subseteq \mathcal{X}_1 \times \mathcal{X}_2 \times \ldots \mathcal{X}_t$ and let $\mathcal{L}_n \subseteq \mathcal{L}^n$. Suppose that the following conditions hold:*

(i) *$f : \mathcal{L}_n \to \{0,1\}$ is a decomposable function with primitive $h : \mathcal{L} \to \{0,1\}$,*

(ii) *ζ is a distribution over $\mathcal{L} \times \mathcal{D}$, such that for any $d \in \mathcal{D}$ the distribution $(\zeta \mid D = d)$ is a product distribution,*

(iii) *$\eta = \zeta^n$ is supported on $\mathcal{L}_n \times \mathcal{D}^n$, and*

(iv) *the marginal probability distribution of η over $\mathcal{L}_n$ is a collapsing distribution for f with respect to h.*

Then $CIC_{\eta,\delta}(f) \geq n \cdot CIC_{\zeta,\delta}(h)$.

3 Communication Lower Bound for Mostly Set Disjointness

Let $[n] = \{1, 2, \ldots, n\}$. We let $H(X)$ denote the entropy of a random variable X, and $I(X;Y) = H(X) - H(X|Y)$ be the mutual information.

3.1 The Hard Distribution

▶ **Definition 11.** *Denote by* $\mathsf{MostlyDISJ}_{n,l,t}$*, the multiparty Mostly Set-Disjointness problem in which each player $j \in [t]$ receives an n-dimensional input vector $X_j = (X_{j,1}, \ldots, X_{j,n})$ where $X_{j,i} \in \{0,1\}$ and the input to the protocol falls into either of the following cases:*

- ***NO:** For all $i \in [n]$, $\sum_{j\in[t]} X_{j,i} \leq 1$*
- ***YES:** There exists a unique $i \in [n]$ such that $\sum_{j\in[t]} X_{j,i} = l$ and for all other $i' \neq i, \sum_{j\in[t]} X_{j,i'} \leq 1$.*

The final player must output 1 if the input is in the YES case and 0 in the NO case.

Let $\mathcal{L} \subset \{0,1\}^t$ be the set of valid inputs along one index in $[n]$ for $\mathsf{MostlyDISJ}_{n,l,t}$, i.e., the set of elements in $x \in \{0,1\}^t$ with $\sum_{j\in[t]} x_j \leq 1$ or $\sum_{j\in[t]} x_j = l$. Let $\mathcal{L}_n \subset \mathcal{L}^n$ denote the set of valid inputs to the $\mathsf{MostlyDISJ}_{n,l,t}$ function.

Then $\mathsf{MostlyDISJ}_{n,l,t} : \mathcal{L}_n \to \{0,1\}$ is defined as: $\mathsf{MostlyDISJ}_{n,l,t}(X_1,\ldots,X_t) = \bigvee_{i\in[n]} F_{l,t}(X_{1,i},\ldots,X_{t,i})$ for the function $F_{l,t} : \mathcal{L} \to \{0,1\}$ defined as: $F_{l,t}(x_1,\ldots,x_t) = \bigvee_{\substack{S\subseteq[t]\\|S|=l}} \bigwedge_{j\in S} x_j$. This means that $\mathsf{MostlyDISJ}_{n,l,t}$ is OR-decomposable into n copies of $F_{l,t}$ and we may hope to apply a direct sum theorem with an appropriate distribution over the inputs.

In order to prove a lower bound on the conditional information complexity, we need to define a "hard" distribution over the inputs to $\mathsf{MostlyDISJ}_{n,l,t}$. We define the distribution η over $\mathcal{L}_n \times \mathcal{D}^n$ where $\mathcal{D} = [t]$ as follows:

- For each $i \in [n]$ pick $D_i \in [t]$ uniformly at random and sample $X_{D_i,i}$ uniformly from $\{0,1\}$ and for all $j' \neq D_i$ set $X_{j',i} = 0$.
- Pick $I \in [n]$ uniformly at random and $Z \in \{0,1\}$
- if $Z = 1$, pick a set $S \subseteq [t]$ such that $|S| = l$ uniformly at random and for all $j \in S$ set $X_{j,I} = 1$ and for all $j \notin S$, set $X_{j,I} = 0$

Let μ_0 denote the distribution for each $i \in [n]$ conditioned on $Z = 0$. For any $d \in [t]$, when $D = d$, the conditional distribution over $\mathcal{L}$ is the uniform distribution over $\{0, e_d\}$ and hence a product distribution. Let η_0 be the distribution η conditioned on $Z = 0$. Clearly, $\eta_0 = \mu_0^n$.

This definition of $\mathsf{MostlyDISJ}_{n,l,t}$ and the hard distribution η_0 allows us to apply the Direct Sum theorem (Theorem 10) of [5]. Note that: (i) $\mathsf{MostlyDISJ}_{n,l,t}$ is OR-decomposable by $F_{l,t}$, (ii) μ_0 is a distribution over $\mathcal{L} \times [t]$ such that the marginal distribution $(\mu_0 \mid D = d)$ over $\mathcal{L}$ is uniform over $\{0, e_d\}$(and hence a product distribution), (iii) $\eta_0 = \mu_0^n$, and (iv) since $\mathsf{MostlyDISJ}_{n,l,t}$ is OR-decomposable and η_0 has support only on inputs in the NO case, η_0 is a collapsing distribution for $\mathsf{MostlyDISJ}_{n,l,t}$ with respect to $F_{l,t}$. Hence:

$$CIC_{\eta_0,\delta}(\mathsf{MostlyDISJ}_{n,l,t}) \geq n \cdot CIC_{\mu_0,\delta}(F_{l,t}) \tag{2}$$

3.2 Information Cost for a Single Bit

A key lemma for our argument is that the players can be implemented so that they only "observe" their input bits with small probability. The model here is that each player's input starts out hidden, but they can at any time choose to observe their input. Before they observe their input, however, all their decisions (including messages sent and choice of whether to observe) depend on the transcript and randomness, but not the player's input.

In this section we use π to denote the protocol in consideration and abuse notation slightly by using π_x to denote the distribution of the transcript of the protocol π on input x.

▶ **Definition 12.** *Any (possibly multi-round) communication protocol involving n players, where each player receives one input bit, is defined to be a "clean" protocol with respect to player i if, in each round,*

1. *if player i has previously not "observed" his input bit, he "observes" his input bit with some probability that is a function only of the previous messages in the protocol,*
2. *if player i has not observed his input bit in this round or any previous round, then his message distribution depends only on the previous messages in the protocol but not his input bit, and*

3. *if player i* has *observed his input bit in this round or any previous round, then – for a fixed value of the previous messages in the protocol – his distribution of messages on input* 0 *and on input* 1 *are* disjoint.

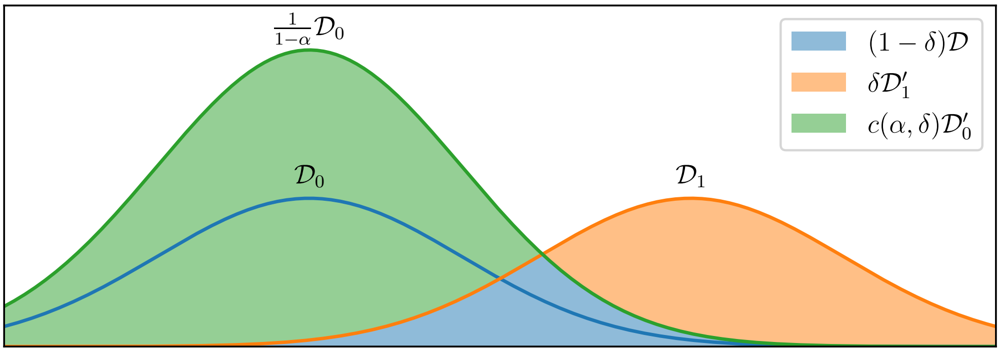

Figure 1 An illustration of Lemma 13, given a parameter α and pair of distributions $(\mathcal{D}_0, \mathcal{D}_1)$. We set $(1-\delta)\mathcal{D}$ to be the overlap between $\mathcal{D}_1$ and $\frac{1}{1-\alpha}\mathcal{D}_0$, then $\mathcal{D}'_0$ and $\mathcal{D}'_1$ to be proportional to the remainder of $\frac{1}{1-\alpha}\mathcal{D}_0$ and $\mathcal{D}_1$, respectively. These $\mathcal{D}'_0$ and $\mathcal{D}'_1$ are disjoint.

We start off by proving a lemma about decomposing any two arbitrary distributions into one "common" distribution and two disjoint different distributions. This lemma will enable us to show that any communication protocol can be simulated in a clean manner.

▶ **Lemma 13.** *Let $\mathcal{D}_0, \mathcal{D}_1$ be two distributions, and $\alpha \in [0,1]$. There exist three distributions $\mathcal{D}, \mathcal{D}'_0, \mathcal{D}'_1$ and a parameter $\delta \in (0,1)$ such that: $\mathcal{D}_0 = (1-\alpha)(1-\delta)\mathcal{D} + (1-(1-\alpha)(1-\delta))\mathcal{D}'_0$, $\mathcal{D}_1 = (1-\delta)\mathcal{D} + \delta\mathcal{D}'_1$, and $\mathcal{D}'_0$ has a disjoint support from $\mathcal{D}'_1$.*

We refer the reader to Figure 1 for an illustration corresponding to Lemma 13.

Proof. We begin with two special cases. If $\alpha = 1$, then setting $\delta = 0$ allows us to set $\mathcal{D}'_0 = \mathcal{D}_0$, $\mathcal{D} = \mathcal{D}_1$. $\mathcal{D}'_1$ may be any arbitrary distribution that has disjoint support from $\mathcal{D}'_0$. If $\mathrm{supp}(\mathcal{D}_0) \cap \mathrm{supp}(\mathcal{D}_1) = \varnothing$, we may set $\delta = 1$, $\mathcal{D}'_0 = \mathcal{D}_0$ and $\mathcal{D}'_1 = \mathcal{D}_1$.

So it suffices to consider the case where $\alpha < 1$ and $\mathrm{supp}(\mathcal{D}_0) \cap \mathrm{supp}(\mathcal{D}_1) \neq \varnothing$. Let $\mathcal{D}$ and δ be such that $\mathcal{D}(x) = \frac{1}{1-\delta}\min(\frac{1}{1-\alpha}\mathcal{D}_0(x), \mathcal{D}_1(x))$ is a distribution over the support of $\mathcal{D}_0$. Then, it suffices to define:

$$\mathcal{D}'_0(x) = \begin{cases} 0 & \text{if } \frac{1}{1-\alpha}\mathcal{D}_0(x) \le \mathcal{D}_1(x) \\ \frac{1}{1-(1-\alpha)(1-\delta)}(\mathcal{D}_0(x) - (1-\alpha)\mathcal{D}_1(x)) & \text{otherwise} \end{cases}$$

and we define:

$$\mathcal{D}'_1(x) = \begin{cases} 0 & \text{if } \frac{1}{1-\alpha}\mathcal{D}_0(x) \ge \mathcal{D}_1(x) \\ \frac{1}{\delta}(\mathcal{D}_1(x) - \frac{\mathcal{D}_0(x)}{1-\alpha}) & \text{otherwise} \end{cases}$$

If $\alpha = 1$, we set $\mathcal{D}'_0 = \mathcal{D}_0$, $\mathcal{D}(x) = \frac{1}{1-\delta}\min(\mathcal{D}_1(x), \mathcal{D}_0(x))$ where δ is a scaling term which ensures that $\mathcal{D}(x)$ is a valid distribution. ◀

▶ **Lemma 14.** *Consider any (possibly multi-round) communication protocol π where each player receives one input bit. Then for any player i, the protocol can simulated in a manner that is "clean" with respect to that player.*

Proof. Let b denote player i's bit. We use "round r" to refer to the rth time that player i is asked to speak. Let m_r be the transcript of the protocol just before player i speaks in round r, and let m_r^+ denote the transcript immediately after player i speaks in round r. Let $\mathcal{D}^b_{m_r}$ be the distribution of player i's message the rth time he is asked to speak, conditioned on the transcript so far being m_r and on player i having the bit b. We will describe an implementation of player i that produces outputs with the correct distribution $\mathcal{D}^b_{m_r}$ such that the implementation only looks at b with relatively small probability.

In the first round, given m_1, player i looks at b with probability $d_{\mathrm{TV}}(\mathcal{D}^0_{m_1}, \mathcal{D}^1_{m_1})$. If he does not look at the bit, he outputs each message m with probability proportional to $\min(\mathcal{D}^0_{m_1}(m), \mathcal{D}^1_{m_1}(m))$; if he sees the bit b, he outputs each message m with probability proportional to $\max(0, \mathcal{D}^b_{m_1}(m) - \mathcal{D}^{1-b}_{m_1}(m))$. His output is then distributed according to $\mathcal{D}^b_{m_r}$. Note also that, for any message m, it is not possible that the player can send m both after reading a 0 and after reading a 1.

In subsequent rounds r, given m_r, player i needs to output a message with distribution $\mathcal{D}^b_{m_r}$. Let p_0 denote the probability that the player has already observed his bit in a previous round, conditioned on m_r and $b = 0$; let p_1 be analogous for $b = 1$. We will show by induction that $\min(p_0, p_1) = 0$ for all m_r. That is, any given transcript may be compatible with having already observed a 0 or a 1 but not both. As noted above, this is true for $r = 2$.

Without loss of generality, suppose $p_1 = 0$. We apply Lemma 13 to $\mathcal{D}^0_{m_r}$ and $\mathcal{D}^1_{m_r}$ with $\alpha = p_0$, obtaining three distributions $(\mathcal{D}, \mathcal{D}_0, \mathcal{D}_1)$ such that $\mathcal{D}^0_{m_r} = (1 - p_0)(1 - \delta)\mathcal{D} + (1 - (1 - p_0)(1 - \delta))\mathcal{D}_0$ and $\mathcal{D}^1_{m_r} = (1 - \delta)\mathcal{D} + \delta\mathcal{D}_1$, and $\mathcal{D}_0$ is disjoint from $\mathcal{D}_1$.

Player i behaves as follows: if he has not observed his bit already, he does so with probability δ. After this, if he still has not observed his bit, he outputs a message according to $\mathcal{D}$; if he has observed his bit b, he outputs according to $\mathcal{D}_b$.

The resulting distribution is $\mathcal{D}^b_{m_r}$ regardless of b, and the set of possible transcripts where a 1 has been observed is disjoint from those possible where a 0 has been observed. By induction, this holds for all rounds r. Thus, this is a simulation of the original protocol that is "clean" with respect to player i. ◀

▶ **Lemma 15.** *Consider any (possibly multiround) communication protocol π where each player receives one bit. Each player i can be implemented such that, if every other player receives a 0 input, player i only observes his input with probability $d_{TV}(\pi_{e_i}, \pi_0)$.*

Proof. Using Lemma 14, we know that player i can be implemented such that the protocol is clean with respect to that player.

We may now analyze the probability p^* that player i ever observes his bit, assuming that all other players receive the input zero. For every possible transcript m let $p_0(m)$ denote the probability, conditioned on the transcript being m and player i's bit being 0, that player i observes his bit at any point during the protocol; let $p_1(m)$ be analogous for the bit being 1. Because the choice of player i to observe his input bit in a clean protocol is independent of the bit, we have that $p^* = \sum_m \Pr_{\pi_0}[m]p_0(m) = \sum_m \Pr_{\pi_{e_i}}[m]p_1(m)$. Moreover, because the protocol is independent of the bit if it is not observed,

$$(1 - p_0(m)) \Pr_{\pi_0}[m] = (1 - p_1(m)) \Pr_{\pi_{e_i}}[m]$$

for all m. By the definition of a clean protocol, the last message player i sends can be consistent with him observing a 0 or a 1 but not both; therefore $p_0(m) = 0$ or $p_1(m) = 0$ for all m. Now, define $S := \{m \mid p_0(m) > 0\} = \{m \mid \Pr_{\pi_0}(m) > \Pr_{\pi_{e_i}}(m)\}$. Therefore

$$d_{\mathrm{TV}}(\pi_0, \pi_{e_i}) = \sum_{m \in S} \Pr_{\pi_0}[m] - \Pr_{\pi_{e_i}}[m] = \sum_{m \in S} p_0(m) \Pr_{\pi_0}[m] = p^*$$

as desired. ◀

Lemma 15 will be used to show that each player has a decent chance of not reading their input. But to get a lower bound for MostlyDISJ, we need a large *set* of players that have a nontrivial chance of all ignoring their input at the same time. We show the existence of such a set, despite the players not being independent. For any $c \in (0, 1)$, define

$$\gamma_c := \frac{1}{c \log(e/c)} \tag{3}$$

We have

▶ **Lemma 16.** *Let $c \in (0, 1)$, $p \in (0, \frac{1-c}{2})$, and γ_c as in (3). For a set of 0-1 random variables $Y_1, \dots, Y_k$ such that $\mathbb{E}[\sum_i Y_i] = pk$, there exists $S \subset \{1, 2, \dots, n\}$ of size ck such that $\Pr[\forall j \in S, Y_j = 0] > e^{-k/\gamma_c - 1}$.*

Proof. We wish to show that there exists a set S such that $Y_i = 0$ for all $i \in S$ with nontrivial probability. Observe that if S were chosen uniformly at random,

$$\mathop{\mathbb{E}}_{S:|S|=ck} \Pr[\forall j \in S, Y_j = 0] \geq \frac{1}{\binom{k}{ck}} \Pr[wt(Y) \leq k - ck] \geq \left(\frac{c}{e}\right)^{ck} \cdot (1 - \frac{p}{1-c}) \geq e^{-1-kc\log(e/c)}.$$

where the first inequality considers the existence of such a set, the second inequality uses $\binom{a}{b} \leq (\frac{e \cdot a}{b})^b$ and Markov's inequality, and $wt(Y)$ denotes the Hamming weight of Y, i.e., number of non-zero entries of the vector Y. Therefore there exists a set S of size $\Omega(ck)$ such that $\Pr[Y_S = 0] \geq e^{-1-kc\log(e/c)}$. ◀

We can now bound the 1-bit communication cost of our problem.

▶ **Lemma 17.** *Given $0 < \delta, c < 1$, γ_c as in (3), and $k \leq \gamma_c \log(\frac{1}{2e\delta})$, for any δ-error protocol for $F_{ck,k}$ we have that $cCost_{\mu_0,\delta}(\pi) = \Omega((1-c)^2)$.*

Proof. Let π be a protocol for $F_{ck,k}$. Let π_x is the distribution of the transcript of the protocol on input x. We start by establishing a connection between conditional information cost and total variation distances. First observe that due to the choice of distribution μ_0, we may write the conditional mutual information as:

$$cCost_{\mu_0,\delta}(\pi) = I(\pi(X_1, \dots, X_k); X_1, \dots, X_k \mid D) = \mathop{\mathbb{E}}_{i \in [k]}[I(X_i; \pi_{0,0,0,\dots X_i,\dots 0,0,0})].$$

Since X_i is uniformly picked from $\{0, 1\}$, this mutual information is a Jensen-Shannon divergence (see, for example, Wikipedia [74] or Proposition A.6 of [5]):

$$I(X_i; \pi_{0,0,0,\dots X_i,\dots 0,0,0}) = D_{JS}(\pi_0, \pi_{e_i}) = \frac{1}{2}\left(D_{KL}(\pi_0, \frac{1}{2}(\pi_0 + \pi_{e_i})) + D_{KL}(\pi_{e_i}, \frac{1}{2}(\pi_0 + \pi_{e_i}))\right)$$

From Pinsker's inequality, $D_{KL}(P, Q) \geq \frac{1}{2} d_{\mathrm{TV}}^2(P, Q)$, so:

$$cCost_{\mu_0,\delta}(\pi) \geq \frac{1}{4} \mathop{\mathbb{E}}_{i \in [k]}[d_{\mathrm{TV}}^2(\pi_0, \frac{1}{2}(\pi_0 + \pi_{e_i})) + d_{\mathrm{TV}}^2(\frac{1}{2}(\pi_0 + \pi_{e_i}), \pi_{e_i})] = \frac{1}{8} \mathop{\mathbb{E}}_{i \in [k]}[d_{\mathrm{TV}}^2(\pi_0, \pi_{e_i})]. \tag{4}$$

This is similar to the connection established in Lemma 6.2 of [5] between conditional information cost and squared Hellinger distance (it is weaker but simpler to show).

Suppose, for the sake of contradiction, that $\sum_i d_{\mathrm{TV}}(\pi_{e_i}, \pi_0) = kp$ where $p < \frac{1-c}{2}$. Suppose for each player $i \in [k]$, that $d_{\mathrm{TV}}(\pi_{e_i}, \pi_0) = p_i$. By Lemma 15, this implies that each player in the protocol can be equivalently implemented in a manner such that – if everyone else receives a 0 – player i only looks at their input with probability p_i. If a player does not look at his bit, it means the player's messages are independent of his input. Let Y_i denote the indicator random variable for the event that player i looks at his input in this equivalent protocol.

For the input $X = 0$, we have $\mathbb{E}[\sum_i Y_i] = \sum p_i = kp$. Observe, that for any set S, if $Y_i = 0$ for all $i \in S$, the players do not see their input. So if E_S denotes the event that $\forall i \in S, Y_i = 0$, then

$$d_{\mathrm{TV}}(\pi_{e_S}, \pi_0) = \Pr[E_S] \cdot d_{\mathrm{TV}}(\pi_{e_S} \mid E_S, \pi_0 \mid E_S) + \Pr[\overline{E_S}] d_{\mathrm{TV}}(\pi_{e_S} \mid \overline{E_S}, \pi_0 \mid \overline{E_S}) \leq \Pr[\overline{E_S}]$$

Since $\mathbb{E}[\sum_i Y_i] = kp$ for $p < \frac{1-c}{2}$, this means by Lemma 16 that there exists a set S with $|S| = ck$ such that $\Pr[E_S] \geq e^{-k/\gamma_c - 1}$. Since $k \leq \gamma_c \log(\frac{1}{2e\delta})$, we have $\Pr[E_S] > 2\delta$. For this S, we have that $d_{\mathrm{TV}}(\pi_{e_S}, \pi_0) < 1 - 2\delta$ and this means that the protocol errs with probability $> \delta$. This is a contradiction. So, we must have $\sum_i d_{\mathrm{TV}}(\pi_{e_i}, \pi_0) > \frac{1-c}{2} k$. By (4) and Jensen's inequality, this gives

$$cCost_{\mu_0,\delta}(\pi) \geq \frac{1}{8} \mathop{\mathbb{E}}_{i \in [k]} [d^2_{\mathrm{TV}}(\pi_{e_i}, \pi_0)] \geq \frac{1}{8} \mathop{\mathbb{E}}_{i \in [k]} [d_{\mathrm{TV}}(\pi_{e_i}, \pi_0)]^2 \geq \frac{(1-c)^2}{32}. \qquad ◀$$

3.3 Finishing it Off

We prove a lower bound on the randomized communication complexity of MostlyDISJ.

▶ **Theorem 18.** *Given $0 < \delta, c < 1$ and $k \leq \gamma_c \log(\frac{1}{2e\delta})$ for γ_c as in (3),*

$$R_\delta(\mathsf{MostlyDISJ}_{n,ck,k}) = \Omega((1-c)^2 n).$$

To prove this, it suffices to prove Lemma 17 where we show a lower bound on the conditional information cost of δ-error protocols for $F_{ck,k}$. This implies a lower bound on the conditional information complexity of $F_{ck,k}$ which together with (2) implies the desired result.

Proof. Combining Proposition 7, Equation (2), and Lemma 17 gives:

$$R_\delta(\mathsf{MostlyDISJ}_{n,ck,k}) \geq CIC_{\eta_0,\delta}(\mathsf{MostlyDISJ}_{n,ck,k}) \geq n \cdot CIC_{\mu_0,\delta}(F_{ck,k}) \gtrsim n(1-c)^2$$

as desired. ◀

In the Lemma 17 we showed that for any protocol for $F_{ck,k}$ with input drawn from μ_0, if the conditional information cost is $o(1)$, there exists an input on which it errs with probability $> \delta$. This implies a lower bound on the conditional information complexity of $F_{ck,k}$.

For algorithms that have large error probability, the success probability can be amplified by using independent copies of the algorithm and taking the majority vote. We use this observation to obtain a lower bound for algorithms with error probability larger than e^{-k}.

▶ **Theorem 1.** MostlyDISJ *with n elements, k players, and $l = ck$ for an absolute constant $c \in (0,1)$ requires $\Omega(\min(n, n\frac{\log(1/\delta)}{k}))$ bits of communication for failure probability δ.*

Proof. For the absolute constant γ_c, when $k < \gamma_c \log(1/\delta)$ (or $\delta < e^{-k/\gamma_c}$), Theorem 18 gives us a lower bound of $\Omega(n)$. Now, consider the case where $\delta > e^{-k/\gamma_c}$. Suppose π is a protocol whose communication cost is C. Then, we may amplify the success probability of this protocol. We create a new protocol π' which runs r independent copies of π in parallel and outputs the majority vote across these copies. The probability of failure for this new protocol is: $\Pr[\geq r/2$ copies of π fail$] \leq \binom{r}{r/2}\delta^{r/2} \leq (4\delta)^{r/2}$. This achieves failure probability e^{-k/γ_c} for $r = O_c(\frac{k}{\log(1/\delta)})$. The lower bound of $\Omega_c(n)$ on the communication complexity of e^{-k/γ_c}-error protocols implies that the communication cost of π is lower bounded by $\Omega(n\frac{\log(1/\delta)}{k})$ in this case. ◀

4 Lower Bounds for ℓ_2-Heavy Hitters

In this section, we will prove lower bounds for certain variants of the ℓ_2 heavy hitters problem in the insertion-only model. Our first lower bound follows from some simple observations and the lower bounds that follow use reductions from the Mostly Set Disjointness problem and the lower bound proved in the previous section.

▶ **Definition 19.** *Given $p > 1$, in the ε-ℓ_p-heavy hitters problem, we are given $\varepsilon \in (0, 1)$ and a stream of items $a_1, \ldots, a_m$ where $a_i \in [n]$. If f_i denotes the frequency of item i in the stream, the algorithm should output all the elements $j \in [n]$ such that:*

$$|f_j| \geq \varepsilon \|f\|_p$$

▶ **Theorem 20.** *Given $\varepsilon \in (0, \frac{1}{4}]$, any deterministic linear sketching algorithm for the ε-ℓ_2-heavy hitters problem must use at least $\Omega(n)$ bits of space even for nonnegative vectors.*

Proof. Assume for the sake of contradiction that $r = o(n)$ and $M \in \mathbb{R}^{r\times n}$ is the sketching matrix which is associated with a deterministic algorithm for $1/4$-ℓ_2 heavy hitters. We may assume that M has orthonormal rows (else there is an orthonormal $r \times n$ matrix whose sketch is linearly related to the sketch in the algorithm and we consider that matrix).

Since M is orthonormal we have $\sum_{i\in[n]} \left\|M^T Me_i\right\|_2^2 \leq r$. So, there must exists an $i^* \in [n]$ such that $\left\|M^T Me_{i^*}\right\|_2^2 \leq r/n$. Consider the vector $v = e_{i^*} - M^T Me_{i^*}$ which lies in the kernel of M. Observe that $v_{i^*}^2 \geq (1 - r/n)^2 \geq 1/2$ and $\|v\|_2^2 \leq 1$ since $I - M^T M$ is a projection.

Now, let us define $w \in \mathbb{R}^n$ such that for all $j \neq i^*$, $w_j = |v_j|$ and $w_{i^*} = 0$. Observe that $w+v$ is a non-negative vector and that i^* is a heavy hitter in $(w+v)$ because $(w+v)_{i^*}^2 \geq 1/2$ and $\|w + v\|_2^2 \leq (2 \|v\|_2)^2 \leq 4$. Since v is in the kernel of M , $M(w + v) = Mw$ and the algorithm must give the same output for both $(w + v)$ and w. However, i^* is a heavy hitter in $(w + v)$ and is not a heavy hitter in w. Hence, by contradiction, $r = \Omega(n)$. ◀

In Theorem 21, we prove a lower bound on the space complexity of δ-error r-pass streaming algorithm for ε-ℓ_p-heavy hitters through a reduction from Mostly Set Disjointness.

▶ **Theorem 21.** *Given $\varepsilon \in (\frac{1}{n^{1/p}}, \frac{1}{2})$ and $p \geq 1$, any δ-error r-pass insertion-only streaming algorithm for ε-ℓ_p-heavy hitters requires $\Omega(\min(\frac{n^{1-1/p}}{r\varepsilon}, n^{1-2/p}\frac{\log(1/\delta)}{r\varepsilon^2}))$ bits of space.*

Proof. Let $\mathcal{A}$ be a δ-error r-pass streaming algorithm for ε-ℓ_p-heavy hitters in the insertion-only model. We describe a multiparty protocol to deterministically solve the Mostly Set Disjointness problem i.e., $\mathsf{MostlyDISJ}_{n,\varepsilon(4n)^{\frac{1}{p}},2\varepsilon(4n)^{\frac{1}{p}}}$ that uses the $\mathcal{A}$. The players simulate a stream which updates a vector $x \in \mathbb{R}^{2n}$. Instead of starting with 0^{2n} (as is the case with most streaming algorithms), the protocol starts off with a vector

$$f_0 = \begin{pmatrix} 0 \\ \vdots \\ 0 \\ 1 \\ \vdots \\ 1 \end{pmatrix} \begin{matrix} \left.\vphantom{\begin{matrix}0\\\vdots\\0\end{matrix}}\right\} n \\ \left.\vphantom{\begin{matrix}1\\\vdots\\1\end{matrix}}\right\} n \end{matrix}$$

Each player performs an update $f \leftarrow f + \delta_i$ to the vector and passes the state of $\mathcal{A}$ to the next player. The update vector δ_i that is processed by player i is just their input x_i padded to length $2n$.

$$\delta = \begin{pmatrix} \\ x_i \\ \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

Observe that if the input to the players is a NO-instance of $\mathsf{MostlyDISJ}_{n,\varepsilon(4n)^{1/p},2\varepsilon(4n)^{1/p}}$, then the final vector f' in the turnstile stream consists of 0-1 entries with at least n 1-s. So, $\|f'\|_p^p \geq n$ and since $\varepsilon \geq n^{1/p}$, no element is a ε-ℓ_p heavy hitter.

If the input is a YES-instance, then the final vector f' consists of $\leq 2n-1$ entries that are 1 and one entry at which is $\varepsilon(4n)^{\frac{1}{p}}$. Since $4\varepsilon^p n \geq \varepsilon^p(2n + 4\varepsilon^p n)$, that entry is a ε-heavy hitter. Using the lower bound of Theorem 1, we know that the total communication in the protocol is $\Omega(\min(n, n\frac{\log(1/\delta)}{\varepsilon n^{1/p}}))$. Since the number of messages sent over r rounds in the protocol is $r \cdot 2\varepsilon(4n)^{1/p}$, there exists at least one player whose communication is:

$$\Omega\Big(\min\big(\frac{n^{1-\frac{1}{p}}}{r\varepsilon}, \frac{n^{1-\frac{2}{p}}\log(1/\delta)}{r\varepsilon^2}\big)\Big)$$

bits and this is a lower bound on the space complexity of $\mathcal{A}$. ◀

A deterministic lower bound follows as a consequence of this lower bound.

▶ **Theorem 22.** *For any $\varepsilon \in (\frac{1}{n^{1/p}}, \frac{1}{2})$ and $p \geq 1$, any r-pass deterministic insertion-only streaming algorithm for ε-ℓ_p-heavy hitters must have a space complexity of $\Omega(\frac{n^{1-1/p}}{r\varepsilon})$ bits.*

In real world applications, one is concerned with lower bounds for naturally occurring frequency vectors. One such naturally occurring frequency distribution is a power law frequency distribution where the i^{th} most frequent element has frequency $\propto \frac{1}{i^\zeta}$ where ζ typically lies in $(0.5, 1]$. Formally:

▶ **Definition 23.** *Let $f \in \mathbb{R}^n$ be a vector such that $|f_{(1)}| \geq |f_{(2)}| \geq \dots |f_{(n)}|$. We say that this vector is power law distributed with parameter ζ if for all $i \in [n]$,*

$$|f_{(i)}| = \Theta(f_{(1)} \cdot i^{-\zeta}) + O(1)$$

In the next theorem, we prove a lower bound on the space complexity of streaming algorithms for ℓ_p-heavy hitters when the frequency vector is power law distributed. We denote $H_m = \sum_{i=1}^{\infty} i^{-m}$ which is finite when $m > 1$.

▶ **Theorem 24.** *Given $p \geq 1$, $\zeta \in (\frac{1}{p}, 1]$ and $\varepsilon \in (\frac{1}{n^\zeta}, \frac{1}{(2+2\cdot H_{p\zeta})^{1/p}})$, any δ-error r-pass streaming algorithm for the ε-ℓ_p-heavy hitters problem where the frequency vector is power law distributed with parameter ζ must have space complexity of $\Omega(\min(n^{1-\zeta}, n^{1-2\zeta}\log(1/\delta)))$.*

Proof. Let $\mathcal{A}$ be a one-pass deterministic streaming algorithm for ℓ_2 heavy hitters when the frequency vector is power law distributed with parameter ζ. We will use a reduction similar to the Theorem 21 to deterministically solve $\mathsf{MostlyDISJ}_{n,n^\zeta,2n^\zeta}$ using $\mathcal{A}$.

Instead of padding the initial vector f_0 with 1's as in Theorem 22, we pad with $\frac{2n^\zeta}{i^\zeta}$ for $i \in [2, n]$.

$$f_0 = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 2n^{\zeta} \cdot 2^{-\zeta} \\ 2n^{\zeta} \cdot 3^{-\zeta} \\ \vdots \\ 2 \end{pmatrix} \begin{matrix} \left.\vphantom{\begin{matrix} 0 \\ 0 \\ \vdots \\ 0 \end{matrix}}\right\} n \\ \left.\vphantom{\begin{matrix} 0 \\ 0 \\ \vdots \\ 0 \end{matrix}}\right\} n \end{matrix}$$

Now, suppose the players pass this frequency vector and successively perform updates to obtain the final frequency vector f'. In the YES instance, there exists one index $i \in [n]$ such that $|f_i'|^p = n^{p\zeta}$ and in the NO instance for all $i \in [n]$, we have $|f_i'| \leq 1$. In the NO case, we have $\|f'\|_p^p \geq \sum_{i \in [2,n+1]} 2n^{p\zeta} \cdot i^{-p\zeta} \geq n^{p\zeta}$ and in the YES case

$$\begin{aligned} \|f'\|_p^p &= \sum_{i \in [2n]} (f_i')^p \\ &\leq n^{p\zeta} + n + \sum_{i \in [2,n+1]} 2n^{p\zeta} \cdot i^{-p\zeta} \\ &\leq n^{p\zeta} + n + H_{p\zeta} 2n^{p\zeta} \\ &< (2 + 2H_{p\zeta}) n^{p\zeta}. \end{aligned}$$

So, in the YES instance, the heavy element is a ε-ℓ_p-heavy hitter since $\varepsilon^p < \frac{1}{2(1+H_{p\zeta})}$ and in the NO instance all the ℓ_p-heavy hitters are indices in $[n+1, 2n]$. Now, the final player runs the ℓ_p-heavy hitter algorithm and if any element from $[1, n]$ is a heavy hitter they output YES and they output NO otherwise.

So, we have described a reduction from ℓ_p heavy hitters for power law distributed vectors to Mostly Set Disjointness. Using Theorem 1, the total communication here is lower bounded by $\Omega(n, n^{1-\zeta} \log(1/\delta))$. Since there are n^{ζ} players, the space complexity lower bound for the streaming algorithm is $\Omega(n^{1-\zeta}, n^{1-2\zeta} \log(1/\delta))$. ◀

5 Application to Low Rank Approximation

As an application of our deterministic ℓ_2-heavy hitters lower bound in insertion streams, we prove a lower bound for the low rank approximation problem in the standard row-arrival model in insertion streams: we see rows $A_1, A_2, \ldots, A_n$ each in $\mathbb{R}^d$, one at a time. At the end of the stream we should output a projection P onto a rank-k space for which $\|A - AP\|_F^2 \leq (1+\epsilon)\|A - A_k\|_F^2$, where A_k is the best rank-k approximation to A. The FrequentDirections algorithm provides a deterministic upper bound of $O(dk/\epsilon)$ words of memory (assuming entries of A are $O(\log(nd))$ bits and a word is $O(\log(nd))$ bits) was shown in [59, 35], and a matching lower bound of $\Omega(dk/\epsilon)$ words of memory was shown in [75]. See also [34] where the upper and lower bounds were combined and additional results for deterministic algorithms were shown.

A natural question is if FrequentDirections can be improved when the rows of your matrix are sparse. Indeed, the sparse setting was shown to have faster running times in [33, 45]. Assuming there are n rows and each row has s non-zero entries, the running time was shown to be $O(sn(k + \log n) + nk^3 + d(k/\epsilon)^3)$, significantly improving the nd time required for dense matrices. Another question is if one can improve the memory required in the sparse setting. The above lower bound has an $\Omega(d)$ term in its complexity because of the need to store directions in $\mathbb{R}^d$. However, it is well-known [40] that any matrix A contains $O(k/\epsilon)$ rows whose row-span contains a rank-k projection P for which $\|A - AP\|_F^2 \leq (1+\epsilon)\|A - A_k\|_F^2$. Consequently, it is conceivable in the stream one could use $O(sk/\epsilon)$ words of memory in the sparse setting, which would be a significant improvement if $s \ll d$. Indeed, in the related communication setting, this was shown to be possible in [10], whereby assuming the rows have at most s non-zero entries it is possible to find such a P with communication only $O(sk/\epsilon)$ words per server, improving upon the $O(dk/\epsilon)$ words per server bound for general protocols, at least in the randomized case. It was left open if the analogous improvement was possible in the streaming setting, even for deterministic algorithms such as FrequentDirections.

Here we use our deterministic lower bound to show it is not possible to remove a polynomial dependence on d in the memory required in streaming setting for deterministic algorithms.

► **Theorem 25.** *Any 1-pass deterministic streaming algorithm outputting a rank-k projection matrix P providing a $(1+\epsilon)$-approximate rank-k low rank approximation requires $\Omega(\sqrt{d})$ bits of memory, even for $k = 1$, $\epsilon = \Theta(1)$, and when each row of A has only a single non-zero entry.*

Proof. Recall in one instantiation of our hard communication problem, the players have sets $S_1, \ldots, S_{\sqrt{d}} \subseteq \{1, 2, \ldots, d\}$ each of size $\sqrt{d}/2$ and either the sets are pairwise disjoint or there exists a unique element i^* occurring in at least $2/3$ fraction of the sets. We associate each element i in each set S_ℓ with a row of A which the standard unit vector e_i which is 1 in position i and 0 in all remaining positions. The stream is defined by seeing all the rows corresponding to elements in S_1, then in S_2, and so on.

Suppose we have seen the first $1/2$ fraction of sets in the stream. In this case, the row i^* must have occurred in at least $1/2 - 1/3 = 1/6$ fraction of sets. Thus, at this point in the stream, the top singular value of A is $\sqrt{d}/6$ and all remaining singular values of A equal 1. Now, the algorithm outputs a rank-1 projection P from its internal memory state. Suppose $P = vv^T$ for a unit vector v. Then

$$\|A - Avv^T\|_F^2 = \|A\|_F^2 - \|Av\|_2^2 \geq \|A\|_F^2 - 1 + (d/36)v_{i^*}^2.$$

Consequently, to obtain a C-approximation for a sufficiently small constant $C > 1$, we must have $v_{i^*}^2 = \Omega(1)$. Since $\|v\|_2^2 = 1$, there is a set T of size $O(1)$ which contains all indices j for which $v_j^2 = \Omega(1)$.

Now, since we have only observed a $1/2$ fraction of sets in the stream, the element i^* must occur in at least $2/3 - 1/2 = 1/6$ fraction of sets in the remaining half of the stream. Thus, for each element in the set T, we can check if it occurs at all in the second half of the stream. However, if there is such an element i^*, it must be the only element in T occurring in the second half of the stream. In case the sets in our hard instance are pairwise disjoint, no element in T will occur in the second half of the stream. Thus, we can deterministically distinguish which of the two cases we are in.

Note that the maximum communication of this reduction is the memory size of the streaming algorithm, together with an additional additive $O(\log d)$ bits of memory to store T. Thus, we get that the memory required of our streaming algorithm is at least $\Omega(\sqrt{d}) - O(\log d) = \Omega(\sqrt{d})$ bits. ◄

6 Algorithm for bounded-length turnstile streams

In this section we show that ℓ_2 heavy hitters on turnstile streams of length $O(n)$ can be solved in $O(n^{2/3})$ space. This is intermediate between the $O(\sqrt{n})$ possible in the insertion-only model and the $\Omega(n)$ necessary in linear sketching.

▶ **Theorem 26.** *There is a deterministic ℓ_2 heavy hitters algorithm for length-L strict turnstile streams with ± 1 updates using $O((L/\varepsilon)^{2/3})$ words of space.*

Proof. Let S be a parameter to be determined later. We run three algorithms in parallel: space-$O(S)$ FrequentElements on the positive updates to x; space-$O(S)$ FrequentElements on the negative updates to x (with sign flipped to be positive); and a linear sketching algorithm for exact S-sparse recovery (e.g., Reed-Solomon syndrome decoding).

Let P, N be the number of positive/negative updates, respectively, so $L = P + N$. Let x^+ and x^- be the sum of positive/negative updates, so $x = x^+ - x^-$. The two FrequentElements sketches give us estimates $\widehat{x}^+$ and $\widehat{x}^-$, respectively, such that for each i:

$$x_i^+ - P/S \le \widehat{x}_i^+ \le x_i^+ \qquad\qquad x_i^- - N/S \le \widehat{x}_i^- \le x_i^-$$

Therefore $\widehat{x} := \widehat{x}^+ - \widehat{x}^-$ satisfies

$$\|\widehat{x} - x\|_\infty \le \max(P/S, N/S) \le L/S.$$

Second, the S-sparse recovery algorithm gives us a $\widehat{y}$ such that, if $\|x\|_0 \le S$, $\widehat{y}_i = x_i$ for all i.

For a strict turnstile stream, we can compute $\|x\|_1 = P - N$. Our algorithm outputs the ε-heavy hitters of $\widehat{y}$ if $\|x\|_1 \le S$, and otherwise outputs the entries of $\widehat{x}$ larger than $3L/S$.

Since $\|x\|_0 \le \|x\|_1$, the output is exactly correct when $\|x\|_1 \le S$. Otherwise, $\|x\|_2 \ge \sqrt{\|x\|_1} \ge \sqrt{S}$, so for $S \ge (L/\varepsilon)^{2/3}$,

$$\|\widehat{x} - x\|_\infty \le L/S \le \varepsilon\sqrt{S} \le \varepsilon\|x\|_2.$$

Therefore the algorithm will output all 4ε-heavy hitters and only 2ε-heavy hitters. Rescaling ε by 4 gives the standard ℓ_2 heavy hitters guarantee. ◀

▶ Remark 27. For non-strict turnstile streams, one can still achieve the ℓ_∞/ℓ_2 guarantee

$$\|\widehat{z} - x\|_\infty \le \varepsilon\|x\|_2$$

with the same space, but the ℓ_2 heavy hitters guarantee (of outputting all ε-heavy hitters and only $\varepsilon/2$-heavy hitters) requires $\Omega(\min(n, L))$ space.

Proof. To achieve the ℓ_∞/ℓ_2 guarantee, we combine $\widehat{x}$ and $\widehat{y}$ in the above algorithm slightly differently: if $\|\widehat{y} - \widehat{x}\|_\infty \le L/S$, output $\widehat{y}$; else, output $\widehat{x}$. Call this output $\widehat{z}$. We have that $\|\widehat{z} - x\|_\infty \le \|\widehat{z} - \widehat{x}\|_\infty + \|\widehat{x} - x\|_\infty \le 2L/S$ unconditionally, and $\widehat{z} = x$ if $\|x\|_0 \le S$. The algorithm outputs $\widehat{z}$.

So when $\|x\|_0 \le S$, this algorithm recovers x exactly and certainly finds the heavy hitters. On the other hand, when $\|x\|_0 \ge S$, we have $\|x\|_2 \ge \sqrt{S}$. Therefore for $S \ge 2(L/\varepsilon)^{2/3}$,

$$\|\widehat{z} - x\|_\infty \le 2L/S \le \varepsilon\sqrt{S} \le \varepsilon\|x\|_2$$

as desired.

For the lower bound, it suffices to consider $L = \Theta(n)$ [otherwise, restrict to the first $\Theta(L)$ coordinates/do nothing interesting after the first $O(n)$ updates]. We can solve EQUALITY on $b = n/10$ bits as follows: using a constant-distance, constant-rate code, associate each input $y \in \{0,1\}^b$ with a codeword $C_y \in \{0,1\}^{n-1}$, such that $\|C_y - C_{y'}\|_1 > n/10$ for all $y \neq y'$. Alice, given the input y, inserts $x_1 = 1$, then inserts C_y on the remaining coordinates. She sends the sketch of the result to Bob, who subtracts his $C_{y'}$ from coordinates $2, \ldots, n$ and asks for the ε-heavy hitters of the result. For any $1 > \varepsilon > 10/\sqrt{n}$, this list will contain coordinate 1 if and only if $y = y'$, solving equality, giving the desired $\Omega(n)$ bound. [And since ε-heavy hitters exactly reconstructs binary vectors on $1/\varepsilon^2$ coordinates, an $\Omega(n)$ bound for $\varepsilon \leq O(1/\sqrt{n})$ is trivial.] ◀

References

1 Rakesh Agrawal and Ramakrishnan Srikant. Fast algorithms for mining association rules in large databases. In *VLDB'94, Proceedings of 20th International Conference on Very Large Data Bases, September 12-15, 1994, Santiago de Chile, Chile*, pages 487–499, 1994.

2 Yuqing Ai, Wei Hu, Yi Li, and David P Woodruff. New characterizations in turnstile streams with applications. In *LIPIcs-Leibniz International Proceedings in Informatics*, volume 50. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2016.

3 Noga Alon, Yossi Matias, and Mario Szegedy. The space complexity of approximating the frequency moments. *J. Comput. Syst. Sci.*, 58(1):137–147, 1999.

4 Brian Babcock, Shivnath Babu, Mayur Datar, Rajeev Motwani, and Jennifer Widom. Models and issues in data stream systems. In *Proceedings of the Twenty-first ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, June 3-5, Madison, Wisconsin, USA*, pages 1–16, 2002.

5 Ziv Bar-Yossef, T. S. Jayram, Ravi Kumar, and D. Sivakumar. An information statistics approach to data stream and communication complexity. *J. Comput. Syst. Sci.*, 68(4):702–732, 2004. `doi:10.1016/j.jcss.2003.11.006`.

6 Paul Beame and Trinh Huynh. Multiparty communication complexity and threshold circuit size of \sfac^0. *SIAM Journal on Computing*, 41(3):484–518, 2012.

7 Paul Beame, Toniann Pitassi, Nathan Segerlind, and Avi Wigderson. A strong direct product theorem for corruption and the multiparty communication complexity of disjointness. *Computational Complexity*, 15(4):391–432, 2006.

8 Radu Berinde, Piotr Indyk, Graham Cormode, and Martin J. Strauss. Space-optimal heavy hitters with strong error bounds. *ACM Trans. Database Syst.*, 35(4):26, 2010. `doi:10.1145/1862919.1862923`.

9 Kevin S. Beyer and Raghu Ramakrishnan. Bottom-up computation of sparse and iceberg cubes. In *SIGMOD 1999, Proceedings ACM SIGMOD International Conference on Management of Data, June 1-3, 1999, Philadelphia, Pennsylvania, USA.*, pages 359–370, 1999.

10 Christos Boutsidis, David P Woodruff, and Peilin Zhong. Optimal principal component analysis in distributed and streaming models. In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pages 236–249, 2016.

11 Mark Braverman, Ankit Garg, Tengyu Ma, Huy L Nguyen, and David P Woodruff. Communication lower bounds for statistical estimation problems via a distributed data processing inequality. In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pages 1011–1020, 2016.

12 Vladimir Braverman, Stephen R. Chestnut, Nikita Ivkin, Jelani Nelson, Zhengyu Wang, and David P. Woodruff. Bptree: an ℓ_2 heavy hitters algorithm using constant memory. *CoRR*, abs/1603.00759, 2016.

13 Vladimir Braverman, Stephen R. Chestnut, Nikita Ivkin, and David P. Woodruff. Beating countsketch for heavy hitters in insertion streams. *STOC*, 2016.

14 Vladimir Braverman, Jonathan Katzman, Charles Seidell, and Gregory Vorsanger. An optimal algorithm for large frequency moments using o(nˆ(1-2/k)) bits. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM 2014, September 4-6, 2014, Barcelona, Spain*, pages 531–544, 2014.

15 Yousra Chabchoub, Christine Fricker, and Hanene Mohamed. Analysis of a bloom filter algorithm via the supermarket model. In *21st International Teletraffic Congress, ITC 2009, Paris, France, September 15-17, 2009*, pages 1–8, 2009.

16 Amit Chakrabarti, Graham Cormode, and Andrew McGregor. A near-optimal algorithm for estimating the entropy of a stream. *ACM Transactions on Algorithms*, 6(3), 2010.

17 Amit Chakrabarti and Sagar Kale. Strong fooling sets for multi-player communication with applications to deterministic estimation of stream statistics. In *IEEE 57th Annual Symposium on Foundations of Computer Science, FOCS 2016, 9-11 October 2016, Hyatt Regency, New Brunswick, New Jersey, USA*, pages 41–50, 2016.

18 Amit Chakrabarti, Subhash Khot, and Xiaodong Sun. Near-optimal lower bounds on the multi-party communication complexity of set disjointness. In *18th IEEE Annual Conference on Computational Complexity, 2003. Proceedings.*, pages 107–117. IEEE, 2003.

19 Ho-Leung Chan, Tak-Wah Lam, Lap-Kei Lee, Jiangwei Pan, Hing-Fung Ting, and Qin Zhang. Edit distance to monotonicity in sliding windows. In *International Symposium on Algorithms and Computation*, pages 564–573. Springer, 2011.

20 Moses Charikar, Kevin Chen, and Martin Farach-Colton. Finding frequent items in data streams. *Theor. Comput. Sci.*, 312(1):3–15, 2004.

21 Arkadev Chattopadhyay and Anil Ada. Multiparty communication complexity of disjointness. *arXiv preprint*, 2008. `arXiv:0801.3624`.

22 Aaron Clauset, Cosma Rohilla Shalizi, and Mark EJ Newman. Power-law distributions in empirical data. *SIAM review*, 51(4):661–703, 2009.

23 A. Cohen, W. Dahmen, and R. DeVore. Compressed sensing and best k-term approximation. *J. Amer. Math. Soc*, 22(1):211–231, 2009.

24 Graham Cormode. Open problem in data streams and related topics. *IITK Workshop on Algorithms for Data Streams*, 2006.

25 Graham Cormode and Marios Hadjieleftheriou. Finding frequent items in data streams. *PVLDB*, 1(2):1530–1541, 2008.

26 Graham Cormode and S Muthukrishnan. An improved data stream summary: the count-min sketch and its applications. *Journal of Algorithms*, 55(1):58–75, 2005.

27 Erik D Demaine, Alejandro López-Ortiz, and J Ian Munro. Frequency estimation of internet packet streams with limited space. In *Algorithms—ESA 2002*, pages 348–360. Springer, 2002.

28 Khanh Do Ba, Piotr Indyk, Eric Price, and David P. Woodruff. Lower bounds for sparse recovery. *CoRR*, abs/1106.0365, 2011.

29 Cristian Estan and George Varghese. New directions in traffic measurement and accounting: Focusing on the elephants, ignoring the mice. *ACM Trans. Comput. Syst.*, 21(3):270–313, 2003.

30 Min Fang, Narayanan Shivakumar, Hector Garcia-Molina, Rajeev Motwani, and Jeffrey D. Ullman. Computing iceberg queries efficiently. In *VLDB'98, Proceedings of 24rd International Conference on Very Large Data Bases, August 24-27, 1998, New York City, New York, USA*, pages 299–310, 1998.

31 Sumit Ganguly. Deterministically estimating data stream frequencies. In Ding-Zhu Du, Xiaodong Hu, and Panos M. Pardalos, editors, *Combinatorial Optimization and Applications, Third International Conference, COCOA 2009, Huangshan, China, June 10-12, 2009. Proceedings*, volume 5573 of *Lecture Notes in Computer Science*, pages 301–312. Springer, 2009.

32 Ankit Garg, Tengyu Ma, and Huy Nguyen. On communication cost of distributed statistical estimation and dimensionality. In *Advances in Neural Information Processing Systems*, pages 2726–2734, 2014.

33 Mina Ghashami, Edo Liberty, and Jeff M Phillips. Efficient frequent directions algorithm for sparse matrices. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 845–854, 2016.

34 Mina Ghashami, Edo Liberty, Jeff M Phillips, and David P Woodruff. Frequent directions: Simple and deterministic matrix sketching. *SIAM Journal on Computing*, 45(5):1762–1792, 2016.

35 Mina Ghashami and Jeff M Phillips. Relative errors for deterministic low-rank matrix approximations. In *Proceedings of the twenty-fifth annual ACM-SIAM symposium on Discrete algorithms*, pages 707–717. SIAM, 2014.

36 Anna C Gilbert, Hung Q Ngo, Ely Porat, Atri Rudra, and Martin J Strauss. L2/l2-foreach sparse recovery with low risk. *arXiv preprint*, 2013. `arXiv:1304.6232`.

37 Parikshit Gopalan and Jaikumar Radhakrishnan. Finding duplicates in a data stream. In *Proceedings of the Twentieth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 402–411, 2009.

38 Vince Grolmusz. The bns lower-bound for multiparty protocols is nearly optimal. *Information and computation*, 112(1):51–54, 1994.

39 André Gronemeier. Asymptotically optimal lower bounds on the nih-multi-party information complexity of the and-function and disjointness. In Susanne Albers and Jean-Yves Marion, editors, *26th International Symposium on Theoretical Aspects of Computer Science, STACS 2009, February 26-28, 2009, Freiburg, Germany, Proceedings*, volume 3 of *LIPIcs*, pages 505–516. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, Germany, 2009.

40 Venkatesan Guruswami and Ali Kemal Sinop. Optimal column-based low-rank matrix reconstruction. In *Proceedings of the twenty-third annual ACM-SIAM symposium on Discrete Algorithms*, pages 1207–1214. SIAM, 2012.

41 Jiawei Han, Jian Pei, Guozhu Dong, and Ke Wang. Efficient computation of iceberg cubes with complex measures. In *Proceedings of the 2001 ACM SIGMOD international conference on Management of data, Santa Barbara, CA, USA, May 21-24, 2001*, pages 1–12, 2001.

42 Jiawei Han, Jian Pei, and Yiwen Yin. Mining frequent patterns without candidate generation. In *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data, May 16-18, 2000, Dallas, Texas, USA.*, pages 1–12, 2000.

43 Nicholas J. A. Harvey, Jelani Nelson, and Krzysztof Onak. Sketching and streaming entropy via approximation theory. In *49th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 489–498, 2008.

44 Christian Hidber. Online association rule mining. In *SIGMOD 1999, Proceedings ACM SIGMOD International Conference on Management of Data, June 1-3, 1999, Philadelphia, Pennsylvania, USA.*, pages 145–156, 1999.

45 Zengfeng Huang. Near optimal frequent directions for sketching dense and sparse matrices. In *International Conference on Machine Learning*, pages 2048–2057, 2018.

46 Piotr Indyk and David P. Woodruff. Optimal approximations of the frequency moments of data streams. In *Proceedings of the 37th Annual ACM Symposium on Theory of Computing (STOC)*, pages 202–208, 2005.

47 Rajesh Jayaram and David P Woodruff. Data streams with bounded deletions. In *Proceedings of the 35th ACM SIGMOD-SIGACT-SIGAI Symposium on Prin ciples of Database Systems*, pages 341–354. ACM, 2018.

48 Thathachar S Jayram and David P Woodruff. The data stream space complexity of cascaded norms. In *2009 50th Annual IEEE Symposium on Foundations of Computer Science*, pages 765–774. IEEE, 2009.

49 TS Jayram. Hellinger strikes back: A note on the multi-party information complexity of and. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, pages 562–573. Springer, 2009.

50 Hossein Jowhari, Mert Saglam, and Gábor Tardos. Tight bounds for lp samplers, finding duplicates in streams, and related problems. In Maurizio Lenzerini and Thomas Schwentick, editors, *Proceedings of the 30th ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, PODS 2011, June 12-16, 2011, Athens, Greece*, pages 49–58. ACM, 2011.

51 John Kallaugher and Eric Price. Separations and equivalences between turnstile streaming and linear sketching. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, pages 1223–1236, 2020.

52 Ravi Kannan, Santosh Vempala, and David Woodruff. Principal component analysis and higher correlations for distributed data. In *Conference on Learning Theory*, pages 1040–1057, 2014.

53 Richard M Karp, Scott Shenker, and Christos H Papadimitriou. A simple algorithm for finding frequent elements in streams and bags. *ACM Transactions on Database Systems (TODS)*, 28(1):51–55, 2003.

54 Abhishek Kumar and Jun (Jim) Xu. Sketch guided sampling - using on-line estimates of flow size for adaptive data collection. In *INFOCOM 2006. 25th IEEE International Conference on Computer Communications, Joint Conference of the IEEE Computer and Communications Societies, 23-29 April 2006, Barcelona, Catalunya, Spain*, 2006.

55 Kasper Green Larsen, Jelani Nelson, Huy L. Nguyen, and Mikkel Thorup. Heavy hitters via cluster-preserving clustering. *Commun. ACM*, 62(8):95–100, 2019.

56 Troy Lee and Adi Shraibman. Disjointness is hard in the multiparty number-on-the-forehead model. *Computational Complexity*, 18(2):309–336, 2009.

57 Yi Li, Huy L Nguyen, and David P Woodruff. Turnstile streaming algorithms might as well be linear sketches. In *Proceedings of the forty-sixth annual ACM symposium on Theory of computing*, pages 174–183, 2014.

58 Yi Li, Huy L. Nguyen, and David P. Woodruff. Turnstile streaming algorithms might as well be linear sketches. In *Symposium on Theory of Computing, STOC 2014, New York, NY, USA, May 31 - June 03, 2014*, pages 174–183, 2014. doi:10.1145/2591796.2591812.

59 Edo Liberty. Simple and deterministic matrix sketching. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 581–588, 2013.

60 Debmalya Mandal, Ariel D. Procaccia, Nisarg Shah, and David P. Woodruff. Efficient and thrifty voting by any means necessary. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, 8-14 December 2019, Vancouver, BC, Canada*, pages 7178–7189, 2019.

61 Debmalya Mandal, Nisarg Shah, and David P. Woodruff. Optimal communication-distortion tradeoff in voting. In *EC '20: The 21st ACM Conference on Economics and Computation, Virtual Event, Hungary, July 13-17, 2020*, pages 795–813, 2020.

62 Gurmeet Singh Manku and Rajeev Motwani. Approximate frequency counts over data streams. In *Proceedings of the 28th international conference on Very Large Data Bases*, pages 346–357. VLDB Endowment, 2002.

63 Ahmed Metwally, Divyakant Agrawal, and Amr El Abbadi. Efficient computation of frequent and top-k elements in data streams. In *Proceedings of the 10th International Conference on Database Theory*, ICDT'05, pages 398–412, Berlin, Heidelberg, 2005. Springer-Verlag. doi:10.1007/978-3-540-30570-5_27.

64 Jayadev Misra and David Gries. Finding repeated elements. *Sci. Comput. Program.*, 2(2):143–152, 1982.

65 Morteza Monemizadeh and David P. Woodruff. 1-pass relative-error L_p-sampling with applications. In *Proceedings of the Twenty-First Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1143–1160, 2010.

66 Shanmugavelayutham Muthukrishnan. *Data streams: Algorithms and applications*. Now Publishers Inc, 2005.

67 Eric Price and David P Woodruff. Lower bounds for adaptive sparse recovery. In *Proceedings of the twenty-fourth annual ACM-SIAM symposium on Discrete algorithms*, pages 652–663. SIAM, 2013.

68 Ashok Savasere, Edward Omiecinski, and Shamkant B. Navathe. An efficient algorithm for mining association rules in large databases. In *VLDB'95, Proceedings of 21th International Conference on Very Large Data Bases, September 11-15, 1995, Zurich, Switzerland.*, pages 432–444, 1995.

69 Alexander A Sherstov. The multiparty communication complexity of set disjointness. In *Proceedings of the forty-fourth annual ACM symposium on Theory of computing*, pages 525–548, 2012.

70 Alexander A Sherstov. Communication lower bounds using directional derivatives. *Journal of the ACM (JACM)*, 61(6):1–71, 2014.

71 Xiaoming Sun, David P. Woodruff, Guang Yang, and Jialin Zhang. Querying a matrix through matrix-vector products. In *46th International Colloquium on Automata, Languages, and Programming, ICALP 2019, July 9-12, 2019, Patras, Greece*, pages 94:1–94:16, 2019.

72 Pascal Tesson. Computational complexity questions related to finite monoids and semigroups, 2003.

73 Hannu Toivonen. Sampling large databases for association rules. In *VLDB'96, Proceedings of 22th International Conference on Very Large Data Bases, September 3-6, 1996, Mumbai (Bombay), India*, pages 134–145, 1996.

74 Wikipedia contributors. Jensen–Shannon divergence – Wikipedia, the free encyclopedia, 2020. [Online; accessed 06-November-2020]. URL: `https://en.wikipedia.org/w/index.php?title=Jensen%E2%80%93Shannon_divergence&oldid=980081721`.

75 David Woodruff. Low rank approximation lower bounds in row-update streams. In *Advances in Neural Information Processing Systems*, pages 1781–1789, 2014.

76 David P Woodruff. New algorithms for heavy hitters in data streams. *arXiv preprint*, 2016. `arXiv:1603.01733`.

77 David P. Woodruff and Qin Zhang. Tight bounds for distributed functional monitoring. In *Proceedings of the 44th Symposium on Theory of Computing Conference, STOC 2012, New York, NY, USA, May 19 - 22, 2012*, pages 941–960, 2012.

78 David P Woodruff and Qin Zhang. An optimal lower bound for distinct elements in the message passing model. In *Proceedings of the twenty-fifth annual ACM-SIAM symposium on Discrete algorithms*, pages 718–733. SIAM, 2014.