

Space-Efficient Estimation of Statistics over Sub-Sampled Streams

Andrew McGregor^{*}
University of Massachusetts
mcgregor@cs.umass.edu

Srikanta Tirthapura[†]
Iowa State University
snt@iastate.edu

A. Pavan[†]
Iowa State University
pavan@cs.iastate.edu

David P. Woodruff
IBM Almaden
dpwoodru@us.ibm.com

ABSTRACT

In many stream monitoring situations, the data arrival rate is so high that it is not even possible to observe each element of the stream. The most common solution is to sample a small fraction of the data stream and use the sample to infer properties and estimate aggregates of the original stream. However, the quantities that need to be computed on the sampled stream are often different from the original quantities of interest and their estimation requires new algorithms. We present upper and lower bounds (often matching) for estimating frequency moments, support size, entropy, and heavy hitters of the original stream from the data observed in the sampled stream.

Categories and Subject Descriptors

F.2 [Analysis of Algorithms & Problem Complexity]

General Terms

Algorithms, Theory

Keywords

data streams, frequency moments, sub-sampling

1. INTRODUCTION

In many stream monitoring situations, the data arrival rate is so high that it is not possible to observe each element of the stream. The most common solution is to sample a small fraction of the data stream and use the sample to infer properties of the original stream. For example, in an IP router, aggregated statistics of the packet stream are maintained through a protocol such as Netflow [6]. In

^{*}Supported by NSF CAREER Award CCF-0953754.

[†]Supported in part by NSF CCF-0916797.

[‡]Supported in part by NSF CNS-0834743, CNS-0831903.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

PODS'12, May 21–23, 2012, Scottsdale, Arizona, USA.

Copyright 2012 ACM 978-1-4503-1248-6/12/05 ...\$10.00.

high-end routers, the load due to statistics maintenance can be so high that a variant of Netflow called *sampled Netflow* has been developed. In randomly sampled netflow, the monitor gets to view only a random sample of the packet stream, and must maintain statistics on the original stream, using this view.

In such scenarios of extreme data deluge, we are faced with two constraints on data processing. First, the entire data set is not seen by the monitor; only a random sample is seen. Second, even the random sample of the input is too large to be stored in main memory (or in secondary memory), and must be processed in a single pass through the data, as in the usual data stream model.

While there has been a large body of work that has dealt with data processing using a random sample (see for example, [2, 3]), and extensive work on the one-pass data stream model (see for example, [1, 15, 17]), there has been little work so far on data processing in the presence of both constraints, where only a random sample of the data set must be processed in a streaming fashion. We note that the estimation of frequency moments over a sampled stream is one of the open problems from [16], posed as Question 13, “Effects of Subsampling”.

1.1 Problem Setting

We assume the setting of *Bernoulli sampling*, described as follows. Consider an input stream $P = \langle a_1, a_2, \dots, a_n \rangle$ where $a_i \in \{1, 2, \dots, m\}$. For a parameter p , $0 < p \leq 1$, a sub-stream of P , denoted L is constructed as follows. For $1 \leq i \leq n$, a_i is included in L with probability p . The stream processor is only allowed to see L , and cannot see P . The goal is to estimate properties of P through processing stream L . In the following discussion, L is called the *sampled stream*, and P is called the *original stream*.

1.2 Our Results

We present algorithms and lower bounds for estimating key aggregates of a data stream by processing a randomly sampled sub-stream. We consider the basic frequency related aggregates, including the number of distinct elements, the k th frequency moments (F_k for $k \geq 1$), the empirical entropy of the frequency distribution, and the heavy hitters.

1. **Frequency Moments:** For the frequency moments F_k , $k \geq 2$, we present $(1 + \epsilon, \delta)$ -approximation algorithms with space complexity¹ $\tilde{O}(p^{-1}m^{1-2/k})$. We show a matching lower

¹Where \tilde{O} notation suppresses factors polynomial in $1/\epsilon$ and $1/\delta$ and factors logarithmic in m and n .

bound of $\Omega(p^{-1}m^{1-2/k})$, showing that the above algorithm is space optimal. This result yields an interesting tradeoff between the sampling probability and the space used by the algorithm. The smaller the sampling probability (up to a certain minimum probability), the greater is the streaming space complexity of F_k . The algorithms and lower bounds for F_k are presented in Section 3.

2. **Distinct Elements:** For the number of distinct elements, F_0 , we show that the current best offline methods for estimating F_0 from a random sample can be implemented in a streaming fashion using very small space. While it is known that random sampling can significantly reduce the accuracy of an estimate for F_0 [5], we show that the need to process this stream using small space does not. The upper and lower bounds are presented in Section 4.
3. **Entropy:** For estimating entropy we first show that no multiplicative approximation is possible in general even when p is constant. However, we show that estimating the empirical entropy on the sampled stream yields a constant factor approximation to the entropy of the original stream given that the entropy is larger than some vanishingly small function of p and n . These results are presented in Section 5.
4. **Heavy Hitters:** We show tight bounds for identifying a set of $O(1/\alpha)$ elements whose frequency exceeds $\alpha F_k^{1/k}$ for $k \in \{1, 2\}$. In the case of $k = 1$, we show that existing heavy hitter algorithms can be used if the stream is sufficiently long compared with p . In the case of $k = 2$, we show how to adapt ideas used in Section 3 to prove matching $\tilde{O}(1/p)$ upper and lower bounds. These results are presented in Section 6.

Another way of interpreting our results is in terms of time-space tradeoffs for data stream problems. Almost every streaming algorithm has a time complexity of at least n , since the algorithm reads and processes each stream update. We show that for estimating F_k (and other problems) it is unnecessary to process each update; instead, it suffices for the algorithm to read each item independently with probability p , and maintain a data structure of size $\tilde{O}(p^{-1} \cdot m^{1-2/k})$. Interestingly, the time to update the data structure per sampled stream item is still only $\tilde{O}(1)$. The time to output an estimate at the end of observation is $\tilde{O}(p^{-1} \cdot m^{1-2/k})$, i.e., roughly linear in the size of the data structure. As an example of the type of tradeoffs that are achievable, for estimating F_2 if $n = \Theta(m)$ we can set $p = \tilde{O}(1/\sqrt{n})$ and obtain an algorithm using $\tilde{O}(\sqrt{n})$ time and $\tilde{O}(\sqrt{n})$ space.

1.3 Related Work

Duffield et al. [9] study issues in Internet traffic measurement using a sampled packet stream, and consider the estimation of the sizes of IP flows and the number of IP flows in a packet stream through observing the sampled stream. In a follow up work [10], they provide methods for estimating the distribution of the sizes of the input flows by observing samples of the original stream; this can be viewed as constructing an approximate histogram. This work is focused on Internet traffic estimation. The techniques used here are maximum likelihood estimation, as well as protocol level detail at the IP and TCP level. While this work deals with inference from a random sample in detail, it does not consider the streaming aspect of the computation, as we do here. Further, the aggregates that we consider (frequency moments, entropy) are not considered there.

Rusu and Dobra [18] consider the estimation of the second frequency moment of a stream, equivalently, the size of the self-join,

through processing the sampled stream. Our work differs from theirs in the following ways. First, their algorithm for the second frequency moment does not yield an $(1 + \epsilon, \delta)$ approximation of the second frequency moment, while ours does. Next, we consider higher frequency moments F_k , for $k > 2$, as well as the entropy, while they do not. Finally, our technique for estimating F_2 is different from theirs; ours relies on counting the number of collisions in the sampled stream, while theirs relies on estimating the second frequency moment of the sampled stream; this way we are able to get the theoretically optimal dependence of space on the sampling probability p . Rusu and Dobra have not explicitly mentioned the space bound of their algorithm; when we derived an $(1 + \epsilon, \delta)$ estimator for F_2 based on their algorithm, we found that the estimator took $\tilde{O}(1/p^2)$ space. We improve the dependency on the sampling probability and obtain an algorithm that only requires $\tilde{O}(1/p)$ space.

Bhattacharya et al. [4] consider stream processing in the model where the stream processor can adaptively “skip” past stream elements, and look at only a fraction of the input stream, thus speeding up stream computation. In their model, the stream processor has the power to decide which elements to see and which to skip past, hence it is “adaptive”; in our model, the stream processor does not have such power, and must deal with the randomly sampled stream that is presented to it. Our model reflects the setup in current network monitoring equipment, such as Randomly Sampled Netflow [6]. They present a constant factor approximation for F_2 , while we present $(1 + \epsilon, \delta)$ approximations for all frequency moments $F_k, k \geq 2$.

There is work on *probabilistic data streams* [7, 14], where the data stream itself consists of “probabilistic” data, and each element of the stream is a probability distribution over a set of possible events. In contrast with our model, in this model the stream processor gets to see the entire input.

2. NOTATION AND PRELIMINARIES

Throughout this paper, we will denote the original length- n stream by $P = \langle a_1, a_2, \dots, a_n \rangle$ and will assume that each element $a_i \in \{1, 2, \dots, m\}$. We denote the sampling probability with p . The sampled stream L is constructed by including each $a_i, 1 \leq i \leq n$, with probability p . It is assumed that the sampling probability p is fixed in advance and is known to the algorithm.

Throughout let f_i be the frequency of item i in the original stream P . Let g_i be the frequency in the sub-sampled stream and note that $g_i \sim \text{Bin}(f_i, p)$. Thus the stream P can be thought of as a frequency vector $\mathbf{f} = (f_1, f_2, \dots, f_m)$. Similarly L can be represented by frequency vector $\mathbf{g} = (g_1, g_2, \dots, g_m)$.

When considering a function F on a stream (e.g., a frequency moment or the entropy) we will denote $F(P)$ and $F(L)$ to indicate that value of the function on the original and sampled stream respectively. When the context is clear, we will also abuse notation and use F to indicate $F(P)$.

We will primarily be interested in (randomized) multiplicative approximations.

DEFINITION 2.1. For $\alpha > 1$ and $\delta \in [0, 1]$, we say \tilde{X} is an (α, δ) -estimator for X if $\Pr \left[\alpha^{-1} \leq X/\tilde{X} \leq \alpha \right] \geq 1 - \delta$.

3. FREQUENCY MOMENTS

We first present our algorithm for estimating the k th frequency moment F_k , and then in Section 3.3, we present lower bounds on space for estimating F_k . We assume k is constant throughout. The guarantee provided by our algorithm is as follows.

THEOREM 3.1. *There is a one pass streaming algorithm which observes L and outputs a $(1 + \epsilon, \delta)$ -estimator to $F_k(P)$ using $\tilde{O}(p^{-1}m^{1-2/k})$ space, assuming $p = \tilde{\Omega}(\min(m, n)^{-1/k})$.*

For $p = \tilde{O}(\min(m, n)^{-1/k})$ there is not enough information to obtain an $(1 + \epsilon, \delta)$ approximation to $F_k(P)$ with any amount of space, see Theorem 4.33 of [2].

DEFINITION 3.1. *The number of ℓ -wise collisions in P is $C_\ell(P) = \sum_{i=1}^m \binom{f_i}{\ell}$. Similarly $C_\ell(L) = \sum_{i=1}^m \binom{g_i}{\ell}$.*

Our algorithm is based on the following connection between the ℓ th frequency moment of a stream and the ℓ -wise collisions in the stream.

$$F_\ell(P) = \ell! \cdot C_\ell(P) + \sum_{i=1}^{\ell-1} \beta_i^\ell F_i(P) \quad (1)$$

where

$$\beta_i^\ell = (-1)^{\ell-i+1} \sum_{1 \leq j_1 \leq \dots \leq j_{\ell-i} \leq \ell-1} j_1 \cdot j_2 \cdots j_{\ell-i}$$

The following lemma relates the expectation of $C_\ell(L)$ to $C_\ell(P)$ and bounds the variance.

LEMMA 3.1.

$$\begin{aligned} \mathbb{E}[C_\ell(L)] &= p^\ell C_\ell(P) \\ \mathbb{V}[C_\ell(L)] &= O(p^{2\ell-1} F_\ell^{2-1/\ell}). \end{aligned}$$

PROOF. Let C denote $C_\ell(L)$. Since each ℓ -way collision in P appears in L with probability p^ℓ , we have $\mathbb{E}[C] = p^\ell C_\ell(P)$. For each $i \in [m]$, let C_i be the number of ℓ -wise collisions in L among items that equal i . Then $C = \sum_{i \in [m]} C_i$. By independence of the C_i ,

$$\mathbb{V}[C] = \sum_{i \in [m]} \mathbb{V}[C_i].$$

Fix an $i \in [m]$. Let S_i be the set of indices in the original stream equal to i . For each $J \subseteq S_i$ with $|J| = k$, let X_J be an indicator random variable if each of the stream elements in J appears in the sampled stream. Then $C_i = \sum_J X_J$. Hence,

$$\begin{aligned} \mathbb{V}[C_i] &= \sum_{J, J'} \mathbb{E}[X_J X_{J'}] - \mathbb{E}[X_J] \mathbb{E}[X_{J'}] \\ &= \sum_{J, J'} p^{|J \cup J'|} - p^{2\ell} \\ &= \sum_{j=1}^{\ell} \binom{f_i}{j} \cdot \binom{f_i - j}{2\ell - 2j} \cdot \binom{2\ell - 2j}{\ell - j} \cdot (p^{2\ell-j} - p^{2\ell}) \\ &= \sum_{j=1}^{\ell} O(f_i^{2\ell-j} p^{2\ell-j}). \end{aligned}$$

Since $F_{2\ell-j}^{1/(2\ell-j)} \leq F_\ell^{1/\ell}$ for all $j = 1, \dots, \ell$, we have

$$\mathbb{V}[C] = O(1) \cdot \sum_{j=1}^{\ell} F_{2\ell-j} \cdot p^{2\ell-j} = O(1) \cdot \sum_{j=1}^{\ell} F_\ell^{2-j/\ell} \cdot p^{2\ell-j}.$$

If we can show that the first term of this sum dominates, the desired variance bound follows. This is the case if $p \cdot F_\ell^{1/\ell} \geq 1$,

Algorithm 1: $F_k(P)$

- 1 Compute $F_1(L)$ exactly and set $\tilde{\phi}_1 = F_1(L)/p$.
- 2 **for** $\ell = 2$ to k **do**
- 3 Let $\tilde{C}_\ell(L)$ be an estimate for $C_\ell(L)$, computed as described in the text.
- 4 Compute

$$\tilde{\phi}_\ell = \frac{\tilde{C}_\ell(L)\ell!}{p^\ell} + \sum_{i=1}^{\ell-1} \beta_i^\ell \tilde{\phi}_i$$
- 5 **end**
- 6 Return $\tilde{\phi}_k$.

since this is the ratio of two consecutive summands. Note that F_ℓ is minimized when there are F_0 frequencies each of value F_1/F_0 . In this case,

$$F_\ell^{1/\ell} = (F_0 \cdot (F_1/F_0)^\ell)^{1/\ell} = F_1/F_0^{1-1/\ell}.$$

Hence, $p \geq 1/F_\ell^{1/\ell}$ if $p \geq F_0^{1-1/\ell}/F_1$, which holds by assumption. \square

We will first describe the intuition behind our algorithm. To estimate $F_k(P)$, by Eq. 1, it suffices to obtain estimates for $F_1(P)$, $F_2(P), \dots, F_{k-1}(P)$ and $C_k(P)$. Our algorithm attempts to estimate $F_\ell(P)$ for $\ell = 1, 2, \dots$ inductively. Since, by Chernoff bounds, $F_1(P)$ is very close to $F_1(L)/p$, $F_1(P)$ can be estimated easily. Thus our problem reduces to estimating $C_k(P)$ by observing the sub-sampled stream L . Since the expected number of collisions in L equals $p^k C_k(P)$, our algorithm will attempt to estimate $C_k(L)$, the number of k -wise collisions in the sub-sampled stream. However, it is not possible to find a good relative approximation of $C_k(L)$ in small space if $C_k(L)$ is small. However, when $C_k(L)$ is small, it does not contribute significantly to the final answer and we do not need a good relative error approximation! We only need that our estimator does not grossly over estimate $C_k(L)$. Our algorithm to estimate $C_k(L)$ will have the following property: If $C_k(L)$ is large, then it outputs a good relative error approximation, and if $C_k(L)$ is small then it outputs a value that is at most $3C_k(L)$.

3.1 The Algorithm

Define a sequence of random variables ϕ_ℓ :

$$\phi_1 = \frac{F_1(L)}{p}, \quad \text{and} \quad \phi_\ell = \frac{C_\ell(L)\ell!}{p^\ell} + \sum_{i=1}^{\ell-1} \beta_i^\ell \phi_i \quad \text{for } \ell > 1.$$

Algorithm 1 inductively computes an estimate $\tilde{\phi}_i$ for each ϕ_i . Note that if $C_\ell(L)/p^\ell$ takes its expected value of $C_\ell(P)$ and we could compute $C_\ell(L)$ exactly, then Eq. 1 implies that the algorithm would return $F_k(P)$ exactly. While this is excessively optimistic we will show that $C_\ell(L)/p^\ell$ is sufficiently close to $C_\ell(P)$ with high probability and that we can construct an estimate for $\tilde{C}_\ell(L)$ for $C_\ell(L)$ such that the final result returned is still a $(1 + \epsilon)$ approximation for $F_k(P)$ with probability at least $1 - \delta$.

We compute our estimate of $\tilde{C}_\ell(L)$ via an algorithm by Indyk and Woodruff [13]. This algorithm attempts to obtain a $1 + \epsilon_{\ell-1}$ approximation of $C_\ell(L)$ for some value of $\epsilon_{\ell-1}$ to be determined. The estimator is as follows. For $i = 0, 1, 2, \dots$ define

$$S_i = \{j \in [m] : \eta(1 + \epsilon')^i \leq g_j \leq \eta(1 + \epsilon')^{i+1}\}$$

where η is randomly chosen between 0 and 1 and $\epsilon' = \epsilon_{\ell-1}/4$. The algorithm of Indyk and Woodruff [13] returns an estimate \tilde{s}_i

for $|S_i|$ and our estimate for $C_\ell(L)$ is defined as

$$\tilde{C}_\ell(L) := \sum_i \tilde{s}_i \binom{\eta(1+\epsilon')^i}{k}$$

The space used by the algorithm is $\tilde{O}(p^{-1}m^{1-2/\ell})$. We defer the details to Section 3.2.

We next define an event \mathcal{E} that corresponds to our collision estimates being sufficiently accurate and the sampled stream being ‘‘well-behaved.’’ The next lemma establishes that $\Pr[\mathcal{E}] \geq 1 - \delta$. We will defer the proof until Section 3.2.

LEMMA 3.2. *Define the event $\mathcal{E} = \mathcal{E}_1 \cap \mathcal{E}_2 \cap \dots \cap \mathcal{E}_k$ where*

$$\mathcal{E}_1 : \tilde{\phi}_1 \in (1 \pm \epsilon_1)F_1(P)$$

$$\mathcal{E}_\ell : |\tilde{C}_\ell(L)/p^\ell - C_\ell(P)| \leq \epsilon_{\ell-1}F_\ell(P)/\ell! \text{ for } \ell \geq 2$$

where $\epsilon_k = \epsilon$, $\epsilon_\ell = (A_\ell + 1)\epsilon_{\ell-1}$, and $A_\ell = \sum_{i=1}^{\ell} |\beta_i^\ell|$. Then

$$\Pr[\mathcal{E}] \geq 1 - \delta.$$

The next theorem establishes that, conditioned on the event \mathcal{E} , the algorithm returns a $(1 \pm \epsilon)$ approximation of $F_k(P)$ as required.

LEMMA 3.3. *Conditioned on \mathcal{E} , we have $\tilde{\phi}_\ell \in (1 \pm \epsilon_\ell)F_\ell(P)$ for all $\ell \in [k]$ and specifically $\tilde{\phi}_k \in (1 \pm \epsilon)F_k(P)$.*

PROOF. The proof is by induction on ℓ . By our assumption $\tilde{\phi}_1$ is an $(1 \pm \epsilon_1)$ approximation of $F_1(P)$. If $i \leq j$, $\epsilon_i \leq \epsilon_j$. Thus the induction hypothesis ensures that $\tilde{\phi}_1, \tilde{\phi}_2, \dots, \tilde{\phi}_{\ell-1}$ are each $(1 \pm \epsilon_{\ell-1})$ approximations for $F_1(P), \dots, F_{\ell-1}(P)$ respectively. Now consider $\tilde{\phi}_\ell$:

$$\begin{aligned} \tilde{\phi}_\ell &= \frac{\tilde{C}_\ell(L)\ell!}{p^\ell} + \sum_{i=1}^{\ell-1} \beta_i^\ell \tilde{\phi}_i \\ &\in \frac{\tilde{C}_\ell(L)\ell!}{p^\ell} + \sum_{i=1}^{\ell-1} \beta_i^\ell (1 \pm \epsilon_{\ell-1})F_i(P) \\ &\subset \ell!C_\ell(P) \pm \epsilon_{\ell-1}F_\ell(P) + \sum_{i=1}^{\ell-1} \beta_i^\ell (1 \pm \epsilon_{\ell-1})F_i(P) \\ &\subset [\ell!C_\ell(P) + \sum_{i=1}^{\ell-1} \beta_i^\ell F_i(P)] \pm [\epsilon_{\ell-1}F_\ell(P) \\ &\quad + \sum_{i=1}^{\ell-1} \beta_i^\ell \epsilon_{\ell-1}F_i(P)] \\ &\subset F_\ell(P) \pm (A_\ell + 1)\epsilon_{\ell-1}F_\ell(P) \\ &\subset (1 \pm \epsilon_\ell)F_\ell(P). \end{aligned}$$

□

3.2 Proof of Lemma 3.2.

Our goal is to show that $\Pr[\mathcal{E}_1 \cap \mathcal{E}_2 \cap \dots \cap \mathcal{E}_k] \geq 1 - \delta$. To do this it will suffice to show that for each $\ell \in [k]$, $\Pr[\mathcal{E}_\ell] \geq 1 - \delta/k$ and appeal to the union bound. It is easy to see that, by Chernoff bounds, the event \mathcal{E}_1 happens with probability at least $1 - \delta/k$. To analyze $\Pr[\mathcal{E}_\ell]$ for $2 \leq \ell \leq k$ we consider the events:

$$\mathcal{E}_\ell^1 : \left| C_\ell(L)/p^\ell - C_\ell(P) \right| \leq \frac{\epsilon_{\ell-1}F_\ell(P)}{2\ell!}$$

$$\mathcal{E}_\ell^2 : \left| \tilde{C}_\ell(L) - C_\ell(L) \right| \leq \frac{\epsilon_{\ell-1}F_\ell(P)}{2\ell!}.$$

By the triangle inequality it is easy to see that $\Pr[\mathcal{E}_\ell^1 \cap \mathcal{E}_\ell^2] \leq \Pr[\mathcal{E}_\ell]$ and hence it suffices to show that $\Pr[\mathcal{E}_\ell^1] \geq 1 - \delta/(2k)$ and $\Pr[\mathcal{E}_\ell^2] \geq 1 - \delta/(2k)$. The first part follows easily from the variance bound in Lemma 3.1.

LEMMA 3.4. $\Pr[\mathcal{E}_\ell^1] \geq 1 - \frac{\delta}{4k}$.

PROOF. There are two cases depending on the size of $\mathbb{E}[C_\ell(L)]$.

Case I: First assume $\mathbb{E}[C_\ell(L)] \leq \frac{\delta\epsilon_{\ell-1}p^\ell F_\ell}{8k\ell!}$. Therefore, by Lemma 3.1, we also know that

$$C_\ell(P) \leq \frac{\delta\epsilon_{\ell-1}F_\ell}{8k\ell!}. \quad (2)$$

By Markov’s bound

$$\Pr\left[C_\ell(L) \leq \frac{\epsilon_{\ell-1}p^\ell F_\ell}{2\ell!}\right] \geq 1 - \frac{\delta}{4k}. \quad (3)$$

Eq. 2 and Eq. 3 together imply that with probability at least $1 - \frac{\delta}{4k}$

$$\left| C_\ell(L)/p^\ell - C_\ell(P) \right| \leq \max\left(C_\ell(L)/p^\ell, C_\ell(P)\right) \leq \frac{\epsilon_{\ell-1}F_\ell}{2\ell!}$$

Case II: Next assume $\mathbb{E}[C_\ell(L)] > \frac{\delta\epsilon_{\ell-1}p^\ell F_\ell}{8k\ell!}$. By Chebyshev’s bound, and using Lemma 3.1, we get:

$$\begin{aligned} \Pr\left[|C_\ell(L) - \mathbb{E}[C_\ell(L)]| \geq \frac{\epsilon_{\ell-1}\mathbb{E}[C_\ell(L)]}{2}\right] \\ \leq \frac{4\mathbb{V}[C_\ell(L)]}{\epsilon_{\ell-1}^2(\mathbb{E}[C_\ell(L)])^2} \leq \frac{Dk^2(\ell!)^2}{\delta^2\epsilon_{\ell-1}^4 p F_\ell^{1/\ell}} \leq \frac{\delta}{4k} \end{aligned}$$

where D is a sufficiently large constant. The last inequality follows because $F_\ell^{1/\ell} \geq F_1/F_0^{1-1/\ell}$ and our assumption on p .

Since $\mathbb{E}[C_\ell(L)] = p^\ell C_\ell(P)$ and $C_\ell(P) \leq F_\ell(P)/\ell!$, we have that

$$\Pr\left[\left|C_\ell(L)/p^\ell - C_\ell(P)\right| \leq \frac{\epsilon_{\ell-1}F_\ell(P)}{2\ell!}\right] \geq 1 - \frac{\delta}{4k}$$

as required. □

We will now show that \mathcal{E}_ℓ^2 happens with high probability by analyzing the algorithm that computes $\tilde{C}_\ell(L)$. We need the following result due to Indyk and Woodruff [13]. Recall that $\epsilon' = \epsilon_{\ell-1}/4$.

THEOREM 3.2 (INDYK AND WOODRUFF [13]). *Let G be the set of indices i for which*

$$|S_i|(1 + \epsilon')^{2i} \geq \frac{\gamma F_2(L)}{\text{poly}(\epsilon'^{-1} \log n)}, \quad (4)$$

then

$$\Pr[\forall i \in G, \tilde{s}_i \in (1 \pm \epsilon')|S_i|] \geq 1 - \frac{\delta}{8k}.$$

For every i (whether it is in G or not) $\tilde{s}_i \leq 3|S_i|$. Moreover, the algorithm runs in space $\tilde{O}(1/\gamma)$.

We say that a set S_i contributes if

$$|S_i| \cdot \binom{(1 + \epsilon')^i}{k} > \frac{C_\ell(L)}{B}.$$

where $B = \text{poly}(\epsilon'^{-1} \log n)$. We will first show that for every S_i that contributes, Eq. (4) holds with high probability with $\gamma = pm^{-1+2/\ell}$.

LEMMA 3.5. Suppose that $C_\ell(L) > \frac{\epsilon_{\ell-1} p^\ell F_\ell(P)}{4\ell!}$. If S_i contributes then

$$\Pr \left[|S_i|(1 + \epsilon')^{2i} \geq \frac{\delta p F_2(L)}{m^{1-2/\ell} \text{poly}(\epsilon'^{-1} \log n)} \right] \geq 1 - \frac{\delta}{8k}.$$

PROOF. Consider a set S_i that contributes. Note that the probability that number $\eta < 1/\text{poly}(\delta^{-1}\epsilon'^{-1} \log n)$ with is at most $1/\text{poly}(\delta^{-1}\epsilon'^{-1} \log n)$. Without loss of generality we can take this probability to be less than $\delta/16k$. By our assumption on $C_\ell(L)$,

$$|S_i|(1 + \epsilon')^{2i} \geq \frac{\epsilon' p^\ell F_\ell(P)}{B\ell!}$$

holds with probability at least $1 - \delta/8k$. Thus

$$|S_i|(1 + \epsilon')^{2i} \geq \frac{\epsilon'^{2/\ell} p^2 F_\ell^{2/\ell}(P)}{(B\ell!)^{2/\ell}} \geq \frac{p^2 F_2(P)}{m^{1-2/\ell} \text{poly}(\epsilon'^{-1} \log n)}$$

where the second inequality is an application of Hölder's inequality.

Note that

$$\mathbb{E}[F_2(L)] = p^2 F_2(P) + p(1-p)F_1(P) \leq 2pF_2(P).$$

Thus, an application of the Markov bound,

$$\Pr \left[F_2(L) \leq \frac{32kpF_2(P)}{\delta} \right] \geq 1 - \frac{\delta}{16k}.$$

Thus

$$|S_i|(1 + \epsilon')^{2i} \geq \frac{\delta p F_2(L)}{m^{1-2/\ell} \text{poly}(\epsilon'^{-1} \log n)}$$

with probability at least $1 - \delta/8k$. \square

Now we are ready to prove that the event \mathcal{E}_ℓ^2 holds with high probability.

$$\text{LEMMA 3.6. } \Pr[\mathcal{E}_\ell^2] \geq 1 - \frac{\delta}{2k}$$

PROOF. There are two cases depending on the size of $C_\ell(L)$.

Case I: Assume $C_\ell(L) \leq \frac{\epsilon_{\ell-1} p^\ell F_\ell(P)}{4\ell!}$. By Theorem 3.2, it follows that $\tilde{C}_\ell(L) \leq 3C_\ell(L)$. Thus

$$\left| \tilde{C}_\ell(L) - C_\ell(L) \right| \leq 2C_\ell(L) \leq \frac{\epsilon_{\ell-1} p^\ell F_\ell(P)}{2\ell!}$$

Case 2: Assume $C_\ell(L) > \frac{\epsilon_{\ell-1} p^\ell F_\ell(P)}{4\ell!}$. By Lemma 3.5, for every S_i that contributes,

$$\Pr \left[|S_i|(1 + \epsilon')^{2i} \geq \frac{\delta p F_2(L)}{m^{1-2/\ell} \text{poly}(\epsilon'^{-1} \log n)} \right] \geq 1 - \frac{\delta}{8k}.$$

Now by Theorem 3.2 for each S_i that contributes $\tilde{s}_i \in (1 \pm \epsilon')|S_i|$, with probability at least $1 - \frac{\delta}{8k}$. Therefore,

$$\Pr \left[\left| \tilde{C}_\ell(L) - C_\ell(L) \right| \leq \epsilon' C_\ell(L) \right] \geq 1 - \frac{\delta}{4k}.$$

If \mathcal{E}_ℓ^1 is true, then:

$$C_\ell(L) \in C_\ell(P)p^\ell \pm \frac{\epsilon_{\ell-1} F_\ell(P)p^\ell}{2\ell!}.$$

Since \mathcal{E}_ℓ^1 holds with probability at least $1 - \frac{\delta}{4k}$, the following inequalities hold with probability at least $1 - \frac{\delta}{2k}$.

$$\begin{aligned} \left| \tilde{C}_\ell(L) - C_\ell(L) \right| &\leq \epsilon' C_\ell(L) \leq \epsilon' C_\ell(P)p^\ell + \frac{\epsilon_{\ell-1} \epsilon' F_\ell(P)p^\ell}{2\ell!} \\ &\leq \frac{\epsilon' F_\ell(P)p^\ell}{\ell!} + \frac{\epsilon_{\ell-1} \epsilon' F_\ell(P)p^\ell}{2\ell!} \\ &\leq \frac{F_\ell(P)p^\ell}{4\ell!} (\epsilon_{\ell-1} + \epsilon_{\ell-1} \epsilon_{\ell-1}) \\ &\leq \frac{F_\ell(P)p^\ell \epsilon_{\ell-1}}{2\ell!} \end{aligned}$$

\square

3.3 Lower Bounds

In this section we prove that $\Omega(n^{1-2/k}/p)$ bits of space are necessary for estimating F_k in the Bernoulli sampling model for $n = \Theta(m)$. Henceforth we assume $p < 1/2$ since for any constant $p > 0$, an $\Omega(n^{1-2/k})$ bound follows immediately from existing bounds when there is no subsampling [11]. The bound will also apply if the original stream P is not ordered adversarially but has been permuted at random.

3.3.1 Intuition

The intuition behind the result is as follows. Existing data stream research establishes that $\Omega(nt^{-2})$ bits of space are required to distinguish between the cases a) all n elements in a stream are unique and b) there exists a high frequency element with multiplicity t . If $t \geq n^{1/k}$, then a good constant approximation of F_k distinguishes these cases. However, if every element of the stream is only observed with probability p , then the length of the new stream is roughly $n' = np$ and any high frequency element now only has frequency roughly $t' = pt$. Hence, distinguishing the two cases now requires $\Omega(n't'^{-2}) = \Omega(nt^{-2}/p) = \Omega(n^{1-2/k}/p)$ bits of space.

3.3.2 Details

Consider the following distribution $\mu(n, p, t)$ over data streams:

- With probability $1/2$: for each of n items, include it in the stream once with probability p , otherwise do not include it. Output a random ordering.
- With probability $1/2$: choose a random special item i , include i in the stream t times, and for each $j \in [n] \setminus \{i\}$, include j in the stream once with probability p , otherwise do not include it. Output a random ordering.

Guha and Huang [11] show that any $1/3$ -error, 1-pass streaming algorithm that determines whether there is a special item in a stream stream distributed according to $\mu(n, 1/2, t)$, requires $\Omega(n/t^2)$ bits of space. Furthermore, and importantly for our application, they show this even if the streaming algorithm is given an arbitrarily large read-only random tape. This follows from the fact that the lower bound in Theorem 2 of their paper is for a multi-party communication game with public coins. In the reduction from the streaming algorithm to a communication game, the first player runs the streaming algorithm on his input, who passes the state of the algorithm to the second player, etc. Therefore, we can assume the public-coin encodes an arbitrarily large read-only random tape that the streaming algorithm can access.

We need to generalize the lower bound of [11] to hold for streams distributed according to $\mu(n, p, t)$. We assume, w.l.o.g., that $n \cdot (2p)$ and n are powers of 2.

LEMMA 3.7. *There is a constant $\delta_0 > 0$ for which any constant-pass streaming algorithm which with probability at least $1 - \delta_0$ decides whether there is a special item in the stream, when the stream is distributed according to $\mu(n, p, t)$, requires $\Omega(np/t^2)$ bits of space.*

PROOF. Suppose we had a constant-pass streaming algorithm A for $\mu(n, p, t)$ which succeeds with sufficiently large constant probability $1 - \delta_0$. We claim that A also succeeds with probability at least $2/3$ on $\mu(n \cdot (2p), 1/2, t)$. The reduction is as follows.

Given a stream S from $\mu(n \cdot (2p), 1/2, t)$, we use the algorithm's random tape to specify a uniformly random injection $h : [n \cdot (2p)] \rightarrow [n]$. This can be done using $O(\log n)$ bits of space (not counting that for the random tape), since given an $i \in [n \cdot (2p)]$, specified with $O(\log n)$ bits, the streaming algorithm sets $h(i)$ to equal the i -th chunk of $\log n$ bits in the random tape. The algorithm replaces each item $i \in [n \cdot (2p)]$ in S with the item $h(i)$, obtaining a stream S' . Observe that there is a special item in S' if and only if there is a special item in S .

It is clear that S' is randomly ordered since S is randomly ordered, so we just need to compare the induced distribution on frequency vectors of S' and of those from a stream drawn from $\mu(n, p, t)$. The latter corresponds to n i.i.d. Bernoulli(p) variables. For the former, we choose a random subset of $[n]$ of size $n \cdot (2p)$, then include each element in our subset in the stream independently with probability $1/2$.

We argue that the variation distance of these two distributions on frequency vectors is sufficiently small. To make this argument, it suffices to consider the number of non-zero frequencies in both distributions, since conditioned on this number any set is equally likely in both distributions.

In one case the number of non-zero frequencies is distributed as $\text{Bin}(n, p)$, and in the other case it is distributed as $\text{Bin}(2pn, 1/2)$. We use the following well-known fact about binomial distributions.

FACT 3.1 (FOLKLORE). *Consider a $\text{Bin}(m, q)$ distribution μ , where $q \geq (\log m)/m$. There are absolute constants $C_{m,q}^U \geq C_{m,q}^L > 0$ so that for any $i \in [qm - \sqrt{qm}, qm + \sqrt{qm}]$,*

$$\frac{C_{m,q}^L}{\sqrt{qm}} \leq \mu(i) \leq \frac{C_{m,q}^U}{\sqrt{qm}}.$$

Let μ_1 be the $\text{Bin}(n, p)$ distribution, and μ_2 the $\text{Bin}(2pn, 1/2)$ distribution. Applying Fact 3.1, there are positive constants $D^L \leq D^U$ so that for any $i \in [pn - \sqrt{pn}, pn + \sqrt{pn}]$ and $\mu \in \{\mu_1, \mu_2\}$,

$$\frac{D^L}{\sqrt{pn}} \leq \mu(i) \leq \frac{D^U}{\sqrt{pn}}.$$

If $\Delta(\mu_1, \mu_2)$ denotes the variation distance, it follows that

$$\Delta(\mu_1, \mu_2) = \frac{1}{2} \|\mu_1 - \mu_2\|_1 \leq \frac{1}{2} \cdot (2 - 2D^L) = 1 - D^L < 1.$$

Hence, if A succeeds with sufficiently high success probability on $\mu(n, p, t)$, then it succeeds with probability at least $2/3$ on $\mu(n \cdot (2p), 1/2, t)$. By the lower bound of [11], A requires $\Omega(np/t^2)$ bits of space. \square

In our Bernoulli sampling with probability p model, parameterized by s , we have the following distribution:

- With probability $1/2$, the frequency vector is $(1, 1, \dots, 1)$.
- With probability $1/2$, the frequency vector is $(s, 1, \dots, 1)$.

In each case we randomly permute the multiset of items (in the first case there are n , while in the second case there are $n + s$), then the stream is formed by walking through the permutation and including each item independently with probability p .

The claim is that if we had a streaming algorithm A for distinguishing these two cases with sufficiently large probability $1 - \delta_1$, then we could design a streaming algorithm for deciding whether there is a special item in the stream when the stream is distributed according to $\mu(n, p, t)$ for a value $t \in [ps - \sqrt{ps}, ps + \sqrt{ps}]$. Indeed, by Fact 3.1 and averaging, there must be a value t in this range so that A succeeds with probability at least $2/3$ conditioned on the number of samples of the special item equaling t . The resulting distribution is equal to $\mu(n, p, t)$, and so by Lemma 3.7, A must use $\Omega(np/(p^2 s^2)) = \Omega(n/(ps^2))$ bits of space.

To get a lower bound for F_k , we set $s = n^{1/k}$ to obtain a constant factor gap in the F_k -value of the streams. Hence, we have the following theorem.

THEOREM 3.3. *Any constant-pass streaming algorithm which $(1+\epsilon, \delta_1)$ -approximates F_k , for sufficiently small constants $\epsilon, \delta_1 > 0$, in the Bernoulli sampling model, requires $\Omega(m^{1-2/k}/p)$ bits of space.*

4. DISTINCT ELEMENTS

There are strong lower bounds for the accuracy of estimates for the number of distinct values through random sampling. The following theorem is from Charikar et al. [5], which we have restated slightly to fit our notation (the original theorem is about database tables). Let F_0 be the number of elements in a data set T of total size n . Note that T maybe a stored data set, and need not be processed in a one-pass manner.

THEOREM 4.1 (CHARIKAR ET AL. [5]). *Consider any (randomized) estimator \hat{F}_0 for the number of distinct values F_0 of T , that examines at most r out of the n elements in T . For any $\gamma > e^{-r}$, there exists a choice of the input T such that with probability at least γ , the multiplicative error is at least $\sqrt{(n-r)/(2r)} \ln \gamma^{-1}$.*

The above theorem implies that if we observe $o(n)$ elements of P , then it is not possible to get even an estimate with a constant multiplicative error. This lower bound for the non-streaming model leads to the following lower bound for sampled streams.

THEOREM 4.2 (F_0 LOWER BOUND). *For sampling probability $p \in (0, 1/12]$, any algorithm that estimates F_0 by observing L , there is an input stream such that the algorithm will have a multiplicative error of $\Omega(1/\sqrt{p})$ with probability at least $(1 - e^{-np})/2$.*

PROOF. Let \mathcal{E}_1 denote the event $|L| \leq 6np$. Let β denote the multiplicative error of any algorithm (perhaps non-streaming) that estimates $F_0(P)$ by observing L . Let $\alpha = \sqrt{\frac{\ln 2}{12p}}$. Let \mathcal{E}_2 denote the event $\beta \geq \alpha$.

Note that $|L|$ is a binomial random variable. The expected size of the sampled stream is $\mathbb{E}[|L|] = np$. By using a Chernoff bound:

$$\Pr[\mathcal{E}_1] = 1 - \Pr[|L| > 6\mathbb{E}[|L|]] \geq 1 - 2^{-6\mathbb{E}[|L|]} > 1 - e^{-np}$$

If \mathcal{E}_1 is true, then the number of elements in the sampled stream is no more than $6np$. Substituting $r = 6np$ and $\gamma = 1/2$ in Theorem 4.1, we get:

$$\Pr[\mathcal{E}_2 | \mathcal{E}_1] \geq \Pr\left[\beta > \sqrt{\left(\frac{n-6np}{12np}\right) \ln 2} \mid \mathcal{E}_1\right] \geq \frac{1}{2}$$

Simplifying, and using $p \leq 1/12$, we get:

$$\Pr[\mathcal{E}_2] \geq \Pr[\mathcal{E}_1 \wedge \mathcal{E}_2] = \Pr[\mathcal{E}_1] \cdot \Pr[\mathcal{E}_2 | \mathcal{E}_1] \geq \frac{1}{2}(1 - e^{-np})$$

□

We now describe a simple streaming algorithm for estimating $F_0(P)$ by observing $L(P, p)$, which has an error of $O(1/\sqrt{p})$ with high probability.

Algorithm 2: $F_0(P)$

- 1 Let X denote a $(1/2, \delta)$ -estimate of $F_0(L)$, derived using any streaming algorithm for F_0 (such as [15]).
 - 2 Return X/\sqrt{p}
-

LEMMA 4.1 (F_0 UPPER BOUND). *Algorithm 2 returns an estimate Y for $F_0(P)$ such that the multiplicative error of Y is no more than $4/\sqrt{p}$ with probability at least $1 - (\delta + e^{-pF_0(P)/8})$.*

PROOF. Let $D = F_0(P)$, and $D_L = F_0(L)$. Let \mathcal{E}_1 denote the event $(D_L \geq pD/2)$, \mathcal{E}_2 denote $(X \geq D_L/2)$, and \mathcal{E}_3 denote the event $(X \leq 3D_L/2)$. Let $\mathcal{E} = \cap_{i=1}^3 \mathcal{E}_i$.

Without loss of generality, let $1, 2, \dots, D$ denote the items that occurred in stream P . Define $X_i = 1$ if at least one copy of item i appeared in L , and 0 otherwise. The different X_i s are all independent. Thus $D_L = \sum_{i=1}^D X_i$ is the sum of independent Bernoulli random variables and

$$\mathbb{E}[D_L] = \sum_{i=1}^D \Pr[X_i = 1].$$

Since each copy of item i is included in D_L with probability p , we have $\Pr[X_i = 1] \geq p$. Thus, $\mathbb{E}[D_L] \geq pD$. Applying a Chernoff bound,

$$\begin{aligned} \Pr[\overline{\mathcal{E}_1}] &= \Pr\left[D_L < \frac{pD}{2}\right] \leq \Pr\left[D_L < \frac{\mathbb{E}[D_L]}{2}\right] \\ &\leq e^{-\mathbb{E}[D_L]/8} \leq e^{-pD/8} \end{aligned}$$

Suppose \mathcal{E} is true. Then we have the following:

$$\frac{pD}{4} \leq \frac{D_L}{2} \leq X \leq \frac{3D_L}{2} \leq \frac{3D}{2}$$

and therefore X has a multiplicative error of no more than $4/\sqrt{p}$.

We now bound the probability that \mathcal{E} is false.

$$\Pr[\overline{\mathcal{E}}] \leq \sum_{i=1}^4 \Pr[\overline{\mathcal{E}_i}] \leq \delta + e^{-pD/8}$$

where we have used the union bound, Eq. (5), and the fact that X is a $(1/2, \delta)$ -estimator of D_L . □

5. ENTROPY

In this section we consider approximating the entropy of a stream.

DEFINITION 5.1. *The entropy of a frequency vector*

$$\mathbf{f} = (f_1, f_2, \dots, f_m)$$

is defined as $H(\mathbf{f}) = \sum_{i=1}^m \frac{f_i}{n} \lg \frac{n}{f_i}$ where $n = \sum_{i=1}^m f_i$.

Unfortunately, in contrast to F_0 and F_k , it is not possible to multiplicatively approximate $H(\mathbf{f})$ even if p is constant.

LEMMA 5.1. *No multiplicative error approximation is possible with probability 9/10 even with $p > 1/2$. Furthermore,*

1. *There exists \mathbf{f} such that $H(\mathbf{f}) = \Theta(\log n/pn)$ but $H(\mathbf{g}) = 0$ with probability at least 9/10.*
2. *There exists \mathbf{f} such that $|H(\mathbf{f}) - H(\mathbf{g})| \geq |\lg(2p)|$ with probability at least 9/10.*

PROOF. First consider the following two scenarios for the contents of the stream. In Scenario 1, $f_1 = n$ and in Scenario 2, $f_1 = n - k$ and $f_2 = f_3 = \dots = f_{k+1} = 1$. In the first case the entropy $H(\mathbf{f}) = 0$ whereas in the second,

$$\begin{aligned} H(\mathbf{f}) &= \frac{n-k}{n} (\lg e) \ln \frac{n}{n-k} + \frac{k}{n} \lg n \\ &= \frac{n-k}{n} \Theta(k/(n-k)) + \frac{k}{n} \lg n \\ &= (\Theta(1) + \lg n) \frac{k}{n}. \end{aligned}$$

Distinguishing these streams requires that at least one value other than 1 is present in the subsampled stream. This happens with probability $(1-p)^k > 1-pk$ and hence with $k = p^{-1}/10$ this probability is less than 9/10.

For the second part of the lemma consider the stream with $f_1 = f_2 = \dots = f_m = 1$ and hence $H(\mathbf{f}) = \lg m$. But $H(\mathbf{g}) = \lg |L|$ where $|L|$ is the number of elements in the sampled stream. By an application of the Chernoff bound $|L|$ is at most $2pm$ with probability at least 9/10 and the result follows. □

Instead we will show that it is possible to approximate $H(\mathbf{f})$ up to a constant factor with an additional additive error term that tends to zero if $p = \omega(n^{-1/3})$. It will also be convenient to consider the following quantity:

$$H_{pn}(\mathbf{g}) = \sum_{i=1}^m \frac{g_i}{pn} \lg \frac{pn}{g_i}.$$

The following proposition establishes that $H_{pn}(\mathbf{g})$ is a very good approximation to $H(\mathbf{g})$.

PROPOSITION 5.1. *With probability 199/200,*

$$|H_{pn}(\mathbf{g}) - H(\mathbf{g})| = O(\log m/\sqrt{pn}).$$

PROOF. By an application of the Chernoff bound, with probability 199/200

$$|pn - \sum_{i=1}^m g_i| \leq c\sqrt{pn}$$

for some constant $c > 0$. Hence, if $n' = \sum_{i=1}^m g_i$ and $\gamma = n'/pn$ it follows that $\gamma = 1 \pm O(1/\sqrt{pn})$. Then

$$\begin{aligned} H_{pn}(\mathbf{g}) &= \sum_{i=1}^m \frac{g_i}{pn} \lg \frac{pn}{g_i} \\ &= \sum_{i=1}^m \frac{\gamma g_i}{n'} \lg \frac{n'}{\gamma g_i} \\ &= H(\mathbf{g}) + O(1/\sqrt{pn}) + O(H(\mathbf{g})/\sqrt{pn}). \end{aligned}$$

□

The next lemma establishes that the entropy of \mathbf{g} is within a constant factor of the entropy of \mathbf{f} plus a small additive term.

LEMMA 5.2. *With probability 99/100, if $p = \omega(n^{-1/3})$,*

1. $H_{pn}(\mathbf{g}) \leq O(H(\mathbf{f}))$.

2. $H_{pn}(\mathbf{g}) \geq H(\mathbf{f})/2 - O\left(\frac{1}{p^{1/2}n^{1/6}}\right)$

PROOF. For the first part of the lemma, first note that

$$\begin{aligned} \mathbb{E}[H_{pn}(\mathbf{g})] &= \sum_{i=1}^m \mathbb{E}\left[\frac{g_i}{pn} \lg \frac{pn}{g_i}\right] \\ &\leq \sum_{i=1}^m \frac{\mathbb{E}[g_i]}{pn} \lg \frac{pn}{\mathbb{E}[g_i]} \\ &= \sum_{i=1}^m \frac{pf_i}{pn} \lg \frac{pn}{pf_i} \\ &= H(\mathbf{f}) \end{aligned}$$

where the inequality follows from Jensen's inequality since the function $x \lg x^{-1}$ is concave. Hence, by Markov's inequality

$$\Pr[H_{pn}(\mathbf{g}) \leq 100H(\mathbf{f})] \geq 99/100.$$

To prove the second part of the lemma, define

$f^* = cp^{-1}\epsilon^{-2} \lg n$ for some sufficiently large constant c and $\epsilon \in (0, 1)$. We then partition $[m]$ into $A = \{i : f_i < f^*\}$ and $B = \{i : f_i \geq f^*\}$ and consider $H(\mathbf{f}) = H^A(\mathbf{f}) + H^B(\mathbf{f})$ where

$$H^A(\mathbf{f}) = \sum_{i \in A} \frac{f_i}{n} \lg \frac{n}{f_i} \quad \text{and} \quad H^B(\mathbf{f}) = \sum_{i \in B} \frac{f_i}{n} \lg \frac{n}{f_i}.$$

By applications of the Chernoff and union bounds, with probability at least 299/300,

$$|g_i - pf_i| \leq \begin{cases} \epsilon pf^* & \text{if } i \in A \\ \epsilon pf_i & \text{if } i \in B \end{cases}.$$

Hence,

$$\begin{aligned} H_{pn}^B(\mathbf{g}) &= \sum_{i \in B} \frac{g_i}{pn} \lg \frac{pn}{g_i} \\ &= \sum_{i \in B} \frac{f_i(1 \pm \epsilon)}{n} \lg \frac{n}{(1 \pm \epsilon)f_i} \\ &= (1 \pm \epsilon)H^B(\mathbf{f}) + O(\epsilon). \end{aligned}$$

For $H_{pn}^A(\mathbf{g})$ we have two cases depending on whether $\sum_{i \in A} f_i$ is smaller or larger than $\theta := cp^{-1}\epsilon^{-2}$. If $\sum_{i \in A} f_i \leq \theta$ then

$$H^A(\mathbf{f}) = \sum_{i \in A} \frac{f_i}{n} \lg \frac{n}{f_i} \leq \frac{\theta \lg n}{n}.$$

On the other hand if $\sum_{i \in A} f_i \geq \theta$ then by an application of the Chernoff bound,

$$\left| \sum_{i \in A} g_i - p \sum_{i \in A} f_i \right| \leq \epsilon p \sum_{i \in A} f_i$$

and hence

$$\begin{aligned} H_{pn}^A(\mathbf{g}) &= \sum_{i \in A} \frac{g_i}{pn} \lg \frac{pn}{g_i} \\ &\geq \lg \frac{n}{(1 + \epsilon)f^*} \sum_{i \in A} \frac{g_i}{pn} \\ &\geq (1 - \epsilon) \lg \frac{n}{(1 + \epsilon)f^*} \sum_{i \in A} \frac{f_i}{n} \\ &\geq \left(1 - \epsilon - \frac{\lg(1 + \epsilon)f^*}{\lg n}\right) H^A(\mathbf{f}). \end{aligned}$$

Combining the above cases we deduce that

$$H_{pn}(\mathbf{g}) \geq (1 - \epsilon - \frac{\lg(p^{-1}\epsilon^{-2} \lg n)}{\lg n})H(\mathbf{f}) - O(\epsilon) - \frac{\epsilon^{-2} \lg n}{pn}.$$

Setting $\epsilon = p^{-1/2}n^{-1/6}$ we get

$$\begin{aligned} H_{pn}(\mathbf{g}) &\geq (1 - p^{-1/2}n^{-1/6} - \frac{\lg(n^{1/3} \lg n)}{\lg n})H(\mathbf{f}) \\ &\quad - O(p^{-1/2}n^{-1/6}) - O\left(\frac{\lg n}{n^{2/3}}\right) \\ &\geq H(\mathbf{f})/2 - O(p^{-1/2}n^{-1/6}). \end{aligned}$$

□

Therefore, by using an existing entropy estimation algorithm (e.g., [12]) to multiplicatively estimate $H(\mathbf{g})$ we have a constant factor approximation to $H(\mathbf{f})$ if $H(\mathbf{f}) = \omega(p^{-1/2}n^{-1/6})$. The next theorem follows directly from Proposition 5.1 and Lemma 5.2.

THEOREM 5.1. *It is possible to approximate $H(\mathbf{f})$ up to a constant factor in $O(\text{polylog}(m, n))$ space if $H(\mathbf{f}) = \omega(p^{-1/2}n^{-1/6})$.*

6. HEAVY HITTERS

There are two common notions for finding heavy hitters in a stream: the F_1 -heavy hitters, and the F_2 -heavy hitters.

DEFINITION 6.1. *In the F_k -heavy hitters problem, $k \in \{1, 2\}$ we are given a stream of updates to an underlying frequency vector f and parameters $\alpha > \epsilon$, and δ . The algorithm is required to output a set S of $O(1/\alpha)$ items such that: (1) every item i for which $f_i \geq \alpha(F_k)^{1/k}$ is included in S , and (2) no item i for which $f_i < (1 - \epsilon)\alpha(F_k)^{1/k}$ is included in S . The algorithm is additionally required to output approximations f'_i with*

$$\forall i \in S, \quad f'_i \in [(1 - \epsilon)f_i, (1 + \epsilon)f_i].$$

The overall success probability should be at least $1 - \delta$.

The intuition behind the algorithm for heavy hitters is as follows. Suppose an item i was an F_k heavy hitter in the original stream P , i.e. $f_i \geq \alpha(F_k)^{1/k}$. Then, by a Chernoff bound, it can be argued that with high probability, g_i the frequency of i in the sampled stream is also close to pf_i . In such a case, it can be shown that i is also a heavy hitter in the sampled stream and will be detected by an algorithm that identifies heavy hitters on the sampled stream (with the right choice of parameters). Similarly, it can be argued that an item i such that $f_i < (1 - \epsilon)\alpha(F_k)^{1/k}$ cannot reach the required frequency threshold on the sampled stream, and will not be returned by the algorithm. We present the analysis below assuming that the heavy hitter algorithm on the sampled stream is the CountMin sketch. Other algorithms for heavy hitters can be used too, such as the Misra-Gries algorithm [17]; note that the Misra-Gries algorithm works on insert-only streams, while the CountMin sketch works on general update streams, with additions as well as deletions.

THEOREM 6.1. *Suppose that*

$$F_1(P) \geq Cp^{-1}\alpha^{-1}\epsilon^{-2} \lg(n/\delta)$$

for a sufficiently large constant $C > 0$. There is a one pass streaming algorithm which observes the sampled stream L and computes the F_1 heavy hitters of the original stream P with probability at least $1 - \delta$. This algorithm uses $O(\epsilon^{-1} \lg^2(n/(\alpha\delta)))$ bits of space.

PROOF. The algorithm is to run the $\text{CountMin}(\alpha', \varepsilon', \delta')$ algorithm of [8] for finding the F_1 -heavy hitters problem on the sampled stream, for $\alpha' = (1 - 2\varepsilon/5) \cdot \alpha$, $\varepsilon' = \varepsilon/10$, and $\delta' = \delta/4$. We return the set S of items i found by CountMin , and we scale each of the f'_i by $1/p$.

Recall that g_i the frequency of item i in the sampled stream L . Then for sufficiently large $C > 0$ given in the theorem statement, by a Chernoff bound,

$$\Pr \left[g_i > \max \left\{ p \left(1 + \frac{\varepsilon}{5} \right) f_i, \frac{C}{2\varepsilon^2} \log \left(\frac{n}{\delta} \right) \right\} \right] \leq \frac{\delta}{4n}.$$

By a union bound, with probability at least $1 - \delta/4$, for all $i \in [n]$,

$$g_i \leq \max \left\{ p \left(1 + \frac{\varepsilon}{5} \right) f_i, \frac{C}{2\varepsilon^2} \log \left(\frac{n}{\delta} \right) \right\}. \quad (5)$$

We also need the property that if $f_i \geq (1 - \varepsilon)\alpha F_1(P)$, then $g_i \geq p(1 - \varepsilon/5)f_i$. For such i , by the premise of the theorem we have

$$\mathbb{E}[g_i] \geq p(1 - \varepsilon)\alpha F_1(P) \geq C(1 - \varepsilon)\varepsilon^{-2} \log(n/\delta).$$

Hence, for sufficiently large C , applying a Chernoff and a union bound is enough to conclude that with probability at least $1 - \delta/4$, for all such i , $g_i \geq p(1 - \varepsilon/5)f_i$.

We set the parameter δ' of CountMin to equal $\delta/4$, and so CountMin succeeds with probability at least $1 - \delta/4$.

Also, $\mathbb{E}[[F_1(L)]] = pF_1(P) \geq C\alpha^{-1}\varepsilon^{-2}(\log n/\delta)$, the inequality following from the premise of the theorem. By a Chernoff bound,

$$\Pr \left[\left(1 - \frac{\varepsilon}{5} \right) pF_1(P) \leq F_1(L) \leq \left(1 + \frac{\varepsilon}{5} \right) pF_1(P) \right] \geq 1 - \frac{\delta}{4}.$$

By a union bound, all events discussed thus far jointly occur with probability at least $1 - \delta$, and we condition on their joint occurrence in the remainder of the proof.

LEMMA 6.1. *If $f_i \geq \alpha F_1(P)$, then*

$$g_i \geq (1 - 2\varepsilon/5) \cdot \alpha F_1(L).$$

If $f_i < (1 - \varepsilon)\alpha F_1(P)$, then

$$g_i \leq (1 - \varepsilon/2)\alpha F_1(L).$$

PROOF. Since $g_i \geq p(1 - \varepsilon/5)f_i$ and also $F_1(L) \leq p(1 + \varepsilon/5)F_1(P)$. Hence,

$$g_i \geq \frac{1 - \varepsilon/5}{1 + \varepsilon/5} \cdot \alpha F_1(L) \geq (1 - 2\varepsilon/5) \cdot \alpha F_1(L).$$

Next consider any i for which $f_i < (1 - \varepsilon)\alpha F_1(P)$. Then

$$\begin{aligned} g_i &\leq \max \left\{ p \left(1 + \frac{\varepsilon}{5} \right) (1 - \varepsilon)\alpha F_1(P), \frac{C}{2\varepsilon^2} \log \left(\frac{n}{\delta} \right) \right\} \\ &\leq \max \left\{ \left(1 - \frac{3\varepsilon}{5} \right) \alpha F_1(L), \frac{C}{2\varepsilon^2} \log \left(\frac{n}{\delta} \right) \right\} \\ &\leq \max \left\{ \left(1 - \frac{\varepsilon}{2} \right) \alpha F_1(L), \frac{\alpha}{2} \cdot \mathbb{E}[F_1(L)] \right\} \\ &\leq \max \left\{ \left(1 - \frac{\varepsilon}{2} \right) \alpha F_1(L), \left(1 + \frac{\varepsilon}{5} \right) \frac{\alpha}{2} F_1(L) \right\} \\ &\leq \left(1 - \frac{\varepsilon}{2} \right) \alpha F_1(L). \end{aligned}$$

□

It follows that by setting $\alpha' = (1 - 2\varepsilon/5) \cdot \alpha$ and $\varepsilon' = \varepsilon/10$, $\text{CountMin}(\alpha', \varepsilon', \delta')$ does not return any $i \in S$ for which $f_i < (1 - \varepsilon)\alpha F_1(P)$, since for such i we have $g_i \leq (1 - \varepsilon/2)\alpha F_1(L)$, and so $g_i < (1 - \varepsilon/10)\alpha' F_1(L)$. On the other hand, for every $i \in S$ for which $f_i \geq \alpha F_1(P)$, we have $i \in S$, since for such i we have $g_i \geq \alpha' F_1(L)$.

It remains to show that for every $i \in S$, we have $f'_i \in [(1 - \varepsilon)f_i, (1 + \varepsilon)f_i]$. By the previous paragraph, for such i we have $f_i \geq (1 - \varepsilon)\alpha F_1(P)$. By the above conditioning, this means that $g_i \geq p(1 - \varepsilon/5)f_i$. We will also have $g_i \leq p(1 + \varepsilon/5)f_i$ if $p(1 + \frac{\varepsilon}{5})f_i \geq \frac{C}{2\varepsilon^2} \log \left(\frac{n}{\delta} \right)$. Since $f_i \geq (1 - \varepsilon)\alpha F_1(P)$, this in turn holds if

$$F_1(P) \geq \frac{1}{2(1 - \varepsilon)(1 + \varepsilon/5)} \cdot Cp^{-1}\alpha^{-1}\varepsilon^{-2} \log \left(\frac{n}{\delta} \right),$$

which holds by the theorem premise provided ε is less than a sufficiently small constant. This completes the proof.

The proof of the next theorem follows from the proofs of Theorem 3.1 for $k = 2$ and Theorem 6.1. We omit the details.

THEOREM 6.2. *Suppose that $p = \tilde{\Omega}(m^{-1/2})$. There is a one pass streaming algorithm which observes the sampled stream L and computes the F_2 heavy hitters of the original stream P with probability at least $1 - \delta$. This algorithm uses $\tilde{O}(p^{-1})$ bits of space.*

THEOREM 6.3. *Any algorithm for solving the F_2 -heavy hitters problem with probability at least $2/3$ in the Bernoulli sampling with probability p model must use $\Omega(p^{-1})$ bits of space.*

PROOF. This follows from our lower bound in Section 3.3 for estimating F_2 in this model. Indeed, there we show that any algorithm which distinguishes between the case when the frequency vector is $(1, 1, \dots, 1)$ and the case when the frequency vector is $(s, 1, \dots, 1)$ requires $\Omega(m/(ps^2))$ bits of space. If we set $s = m^{1/2}$, then in the first case the heavy hitters algorithm is required to return an empty list, while in the second case the heavy hitters algorithm must return a list of size 1. Hence the algorithm can distinguish the two cases and requires $\Omega(1/p)$ bits of space. □

7. CONCLUSION

In this paper we presented small-space stream algorithms and space lower bounds for estimating functions of interest when observing a random sample of the original stream.

There are numerous directions for future work. As we have seen, our results imply time/space tradeoffs for several natural streaming problems. What other data stream problems have interesting time/space tradeoffs? Also, we have so far assumed that the sampling probability p is fixed, and that the algorithm has no control over it. Suppose this was not the case, and the algorithm can change the sampling probability in an adaptive manner, depending on the current state of the stream. Is it possible to get algorithms that can observe fewer elements overall and get the same accuracy as our algorithms? For which precise models and problems is adaptivity useful?

8. REFERENCES

- [1] N. Alon, Y. Matias, and M. Szegedy. The Space Complexity of Approximating the Frequency Moments. *J. Comput. Syst. Sci.*, 58(1):137–147, 1999.
- [2] Z. Bar-Yossef. *The complexity of massive dataset computations*. PhD thesis, UC Berkeley, 2002.
- [3] Z. Bar-Yossef. Sampling lower bounds via information theory. In *STOC*, pages 335–344, 2003.

- [4] S. Bhattacharyya, A. Madeira, S. Muthukrishnan, and T. Ye. How to scalably and accurately skip past streams. In *ICDE Workshops*, pages 654–663, 2007.
- [5] M. Charikar, S. Chaudhuri, R. Motwani, and V. R. Narasayya. Towards estimation error guarantees for distinct values. In *PODS*, 2000.
- [6] Cisco Systems. *Random Sampled NetFlow*. http://www.cisco.com/en/US/docs/ios/12_0s/feature/guide/nfstatsa.html.
- [7] G. Cormode and M. Garofalakis. Sketching probabilistic data streams. In *Proceedings of the 2007 ACM SIGMOD international conference on Management of data*, SIGMOD '07, pages 281–292, 2007.
- [8] G. Cormode and S. Muthukrishnan. An improved data stream summary: the count-min sketch and its applications. *J. Algorithms*, 55(1):58–75, 2005.
- [9] N. G. Duffield, C. Lund, and M. Thorup. Properties and prediction of flow statistics from sampled packet streams. In *Internet Measurement Workshop*, pages 159–171, 2002.
- [10] N. G. Duffield, C. Lund, and M. Thorup. Estimating flow distributions from sampled flow statistics. In *SIGCOMM*, pages 325–336, 2003.
- [11] S. Guha and Z. Huang. Revisiting the direct sum theorem and space lower bounds in random order streams. In *ICALP (1)*, pages 513–524, 2009.
- [12] N. J. A. Harvey, J. Nelson, and K. Onak. Sketching and streaming entropy via approximation theory. In *FOCS*, pages 489–498, 2008.
- [13] P. Indyk and D. 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.
- [14] T. S. Jayram, A. McGregor, S. Muthukrishnan, and E. Vee. Estimating statistical aggregates on probabilistic data streams. *ACM Trans. Database Syst.*, 33:26:1–26:30, December 2008.
- [15] D. M. Kane, J. Nelson, and D. P. Woodruff. On the exact space complexity of sketching and streaming small norms. In *SODA*, pages 1161–1178, 2010.
- [16] A. McGregor, editor. *Open Problems in Data Streams and Related Topics*, 2007. <http://www.cse.iitk.ac.in/users/sganguly/data-stream-probs.pdf>.
- [17] J. Misra and D. Gries. Finding repeated elements. *Science of Computer Programming*, 2(2):143–152, 1982.
- [18] F. Rusu and A. Dobra. Sketching sampled data streams. In *ICDE*, pages 381–392, 2009.