Memory Complexity of Entropy Estimation

Tomer Berg, Or Ordentlich and Ofer Shayevitz

Abstract

We observe an infinite sequence of independent identically distributed random variables $X_1, X_2, \ldots$ drawn from an unknown distribution p over $[n]$, and our goal is to estimate the entropy $H(p) = -\mathbb{E}[\log p(X)]$ within an ε-additive error. To that end, at each time point we are allowed to update a finite-state machine with S states, using a possibly randomized but time-invariant rule, where each state of the machine is assigned an entropy estimate. Our goal is to characterize the minimax memory complexity S^* of this problem, which is the minimal number of states for which the estimation task is feasible with probability at least $1-\delta$ asymptotically, uniformly in p. Specifically, we show that there exist universal constants C_1 and C_2 such that $S^* \leq C_1 \cdot \frac{n(\log n)^4}{\varepsilon^2 \delta}$ for ε not too small, and $S^* \geq C_2 \cdot \max\{n, \frac{\log n}{\varepsilon}\}$ for ε not too large. The upper bound is proved using approximate counting to estimate the logarithm of p, and a finite memory bias estimation machine to estimate the expectation operation. The lower bound is proved via a reduction of entropy estimation to uniformity testing. We also apply these results to derive bounds on the memory complexity of mutual information estimation.

I. Introduction

The problem of inferring properties of an underlying distribution given sample access is called *statistical property estimation*. A typical setup is as follows: given independent samples $X_1, \ldots, X_n$ from an unknown distribution p, the objective is to estimate a property $g(p)$ (e.g., entropy, support size, L_p norm, etc.) under some resource limitation. A prominent example of such a limitation is the amount of available samples, and this limitation gives rise to the notion of sample complexity, namely the minimal number of samples one needs to see in order to estimate $g(p)$ with some given accuracy. Many real–world machine learning and data analysis tasks are limited by insufficient samples, and the challenge of inferring properties of a distribution given a small sample size is encountered in a variety of settings, including text data, customer data, and the study of genetic mutations across a population. The sample complexity of property estimation and, specifically, of entropy estimation, have therefore received much attention in the literature (see Section II for details).

However, in many contemporary settings, collecting enough samples for accurate estimation is less of a problem, and the bottleneck shifts to the computational resources available for the task and, in particular, the available memory size. In this work, we therefore focus on the problem of estimation under memory constraints, and, in particular, entropy estimation. In order to isolate the effect that finite memory has on the fundamental limits of the problem, we let the number of samples we process be arbitrarily large.

Formally, the problem is defined as follows. Let Δ_n be the collection of all distributions over $[n]$. The Shannon entropy of $p \in \Delta_n$ is $H(p) = -\sum_{x\in[n]} p(x)\log p(x)$. Given independent samples $X_1, X_2, \ldots$ from an unknown $p \in \Delta_n$, we would like to accurately estimate $H(p)$ using limited memory. To that end, an *S-state entropy estimator* is a finite-state machine with S states, defined by two functions: The (possibly randomized) memory update function $f : [S] \times [n] \to [S]$, describing the transition between states as a function of an input sample, and the entropy estimate function $\hat{H} : [S] \to [0, \log n]$, assigning an entropy estimate to each state. Letting M_t denote the state of the memory at time t, this finite-state machine evolves according to the rule:

$$M_0 = s_{\text{init}}, \tag{1}$$

$$M_t = f(M_{t-1}, X_t) \in [S], \tag{2}$$

for some predetermined initial state $s_{\text{init}} \in [S]$. If the machine is stopped at time t, it outputs the estimation $\hat{H}(M_t)$. We restrict the discussion to time-invariant memory update function f, since storing the time index necessarily incurs a memory cost, and, furthermore, since the number of samples is unbounded, simply storing the code generating a time-varying algorithm may require unbounded memory. We say that an ϵ-error occurred at time t if our estimate $\hat{H}(M_t)$ is ε-far from the correct entropy. Our figure of merit for the estimator is taken to be its worst-case asymptotic ϵ-error probability:

$$\mathsf{P_e}(f, \hat{H}, \epsilon) = \sup_{p \in \Delta_n} \limsup_{t \to \infty} \Pr\left(|\hat{H}(M_t) - H(p)| > \varepsilon \right). \tag{3}$$

We are interested in the *minimax memory complexity* $S^*(n, \varepsilon, \delta)$, defined as the smallest integer s for which there exist $(f, \hat{H})$ such that $\mathsf{P_e}(f, \hat{H}, \epsilon) \leq \delta$.

Our main result is an upper bound on $S^*(n, \varepsilon, \delta)$, which shows that $\log \frac{n}{\varepsilon^2} + o(\log n)$ bits suffice for entropy estimation when $\varepsilon > 10^{-5}$, thus improving upon the best known upper bounds thus far ([1], [2]). While our focus here is on minimizing the memory complexity of the problem in the limit of infinite number of available samples, we further show that the estimation algorithm attaining this memory complexity upper bound only requires $\tilde{O}(n^c)$ samples, for any $c > 1$.[1] Thus, in entropy estimation one can achieve almost optimal sample complexity and memory complexity, simultaneously. Our proposed algorithm approximates the logarithm of $p(x)$, for a given $x \in [n]$, using a *Morris counter* [3]. The inherent structure of the Morris counter is particularly suited for constructing a nearly-unbiased estimator for $\log p(x)$, making it a natural choice for memory efficient entropy estimation. In order to compute the mean of these estimators, $\mathbb{E}[\widehat{\log p(X)}]$, in a memory efficient manner, a finite-memory bias estimation machine (e.g., [4], [5]) is leveraged for simulating the expectation operator. The performance of a scheme based on this high-level idea is analyzed, and yields the following upper bound on the memory complexity:

Theorem 1. *For any $c > 1$, $\beta > 0$, $0 < \delta < 1$ and $\varepsilon = 10^{-5} + \beta + \psi_c(n)$, we have*

$$S^*(n, \varepsilon, \delta) \leq \frac{(c+1)^4 n \cdot (\log n)^4}{\beta^2 \delta}, \tag{4}$$

where

$$\psi_c(n) = \min\left\{ 1 + n^{-(c-1)+\sqrt{\frac{c}{\log n}}}, 2 \cdot 10^8 \cdot n^{-\frac{1}{2}\cdot(c-1)+\sqrt{\frac{c}{8\log n}}} \right\} = O\left(n^{-\frac{1}{2}\cdot(c-1)+\sqrt{\frac{c}{8\log n}}} \right). \tag{5}$$

Moreover, there is an algorithm that attains (4) *whenever the number of samples is $\Omega\left(\frac{n^c \cdot \mathrm{poly}(\log n)}{\delta} \cdot \mathrm{poly}(\log(1/\delta)) \right)$, and returns an estimation of $H(p)$ within an ε-additive error with probability at least $1 - 3\delta$.*

Note that while $\psi_c(n)$ vanishes for large n, our bound is always limited to $\varepsilon > 10^{-5}$. This small bias is due to inherent properties of the Morris counter, on which we elaborate in Section III. As in this work we are more interested in the large entropy regime (in which the entropy grows with the alphabet size n), the limitation of the attainable additive error to values above 10^{-5} is typically a moderate one. We also note that if n is large and ε not too small, one can choose c arbitrarily close to 1, resulting in an algorithm whose sample complexity has similar dependence on n as those of the limited-memory entropy estimation algorithms proposed in [1] and [2], while requiring far less memory states. This result might be of practical interest for applications in which memory is a scarcer resource than samples, e.g., a limited memory high-speed router that leverages entropy estimation to monitor IP network traffic [6].

Furthermore, we derive two lower bounds on the memory complexity. The first lower bound shows that when $H(p)$ is close to $\log n$, the memory complexity cannot be too small. This bound is obtained via a reduction of

[1]The $\tilde{O}$ suppresses poly-logarithmic terms.

entropy estimation to uniformity testing, by noting that thresholding the output of a good entropy estimation machine around $\log n$ can be used to decide whether p is close to the uniform distribution or not. The bound then follows from the $\Omega(n)$ lower bound of [7] on uniformity testing. The second lower bound follows from the observation that, if the number of states is too small, there must be some value of the entropy at distance greater than ε from all estimate values, hence for this value of the entropy we err with probability 1. Combining these lower bounds yields the following.

Theorem 2. *For any $\varepsilon > 0$, we have*

$$S^*(n, \varepsilon, \delta) \geq \frac{\log n}{2\varepsilon}. \tag{6}$$

Furthermore, if $\varepsilon < \frac{1}{4 \ln 2}$, then

$$S^*(n, \varepsilon, \delta) \geq n(1 - 2\sqrt{\varepsilon \ln 2}). \tag{7}$$

One of several open problems posed by the authors of [1] is to prove a lower bound on the space requirement of a sample optimal algorithm for entropy estimation. Theorem 2 answers this question by giving a lower bound on the memory size needed when the number of samples is infinite, which clearly also holds for any finite number of samples. In the concluding section of the paper, we extend our results to the mutual information estimation problem. Let $(X, Y) \sim p_{XY}$, where p_{XY} is an unknown distribution over $[n] \times [m]$ such that the marginal distribution of X is p_X and the marginal distribution of Y is p_Y. The mutual information between X and Y is given as $I(X;Y) = H(X) + H(Y) - H(X,Y)$. We derive the following bounds on the memory complexity of mutual information estimation, namely the minimal number of states needed to estimate $I(X;Y)$ with additive error at most ε with probability of at least $1 - \delta$, which we denote as $S^*_{\text{MI}}(n, m, \varepsilon, \delta)$.

Theorem 3. *For any $c > 1$, $\beta > 0$ and $\varepsilon = 3 \cdot 10^{-5} + \beta + O\left(n^{-\frac{1}{2}\cdot(c-1)} \vee m^{-\frac{1}{2}\cdot(c-1)}\right)$,*

$$S^*_{MI}(n, m, \varepsilon, \delta) \leq \frac{(c+1)^6 nm \cdot (\log nm)^6}{\beta^2 \delta}. \tag{8}$$

For $\varepsilon < \frac{1}{12 \ln 2}$, and if $\frac{n}{\log^3 n} = \Omega(\log^7 m)$ and $\frac{m}{\log^3 m} = \Omega(\log^7 n)$ both hold, then

$$S^*_{MI}(n, m, \varepsilon, \delta) = \Omega\left(\frac{n \cdot m}{\log^3 n \cdot \log^3 m}\right). \tag{9}$$

II. Related work

The study of estimation under memory constraints has received far less attention then the sample complexity of statistical estimation. References [8], [9] studied this setting for hypothesis testing with finite memory, and [10], [4] have studied estimating the bias of a coin using a finite state machine. It has then been largely abandoned, but recently there has been a revived interest in space-sample trade-offs in statistical estimation, and many works have addressed different aspects of the learning under memory constraints problem over the last few years. See, e.g., [11], [12], [13], [14], [15], [16], [17], [18], [19] for a non exhaustive list of recent works.

The problem of estimating the entropy with limited independent samples from the distribution has a long history. It was originally addressed by [20], who suggested the simple and natural empirical plug-in estimator. This estimator outputs the entropy of the empirical distribution of the samples, and its sample complexity [21] is $\Theta\left(\frac{n}{\varepsilon} + \frac{\log^2 n}{\varepsilon^2}\right)$. [21] showed that the plug-in estimator is always consistent, and the resulting sample complexity was shown to be linear in n. In the last two decades, many efforts were made to improve the bounds on the sample complexity. Paninski [22], [23]] was the first to prove that it is possible to consistently estimate the entropy using

sublinear sample size. While the scaling of the minimal sample size of consistent estimation was shown to be $\frac{n}{\log n}$ in the seminal results of [24], [25], the optimal dependence of the sample size on both n and ε was not completely resolved until recently. In particular, $\Omega\left(\frac{n}{\varepsilon \log n}\right)$ samples were shown to be necessary, and the best upper bound on the sample complexity was relied on an estimator based on linear programming that can achieve an additive error ε using $O\left(\frac{n}{\varepsilon^2 \log n}\right)$ samples [26]. This gap was partially amended in [27] by a different estimator, which requires $O\left(\frac{n}{\varepsilon \log n}\right)$ samples but is only valid when ε is not too small. The sharp sample complexity was shown by [28], [29], [29] to indeed be

$$\Theta\left(\frac{n}{\varepsilon \log n} + \frac{\log^2 n}{\varepsilon^2}\right). \tag{10}$$

The space-complexity (which is the minimal memory in bits needed for the algorithm) of estimating the entropy of the empirical distribution of the data stream is well-studied for worst-case data streams of a given length, see [30], [6], [31]. Reference [32] addressed the problem of deciding if the entropy of a distribution is above or beyond than some predefined threshold, using algorithms with limited memory. The trade-off between sample complexity and space/communication complexity for the entropy estimation of a distribution is the subject of a more recent line of work. The earliest work on the subject is [1], where the authors constructed an algorithm which is guaranteed to work with $O(k/\varepsilon^3 \cdot \mathrm{polylog}(1/\varepsilon))$ samples and any memory size $b \geq 20 \log\left(\frac{k}{\varepsilon}\right)$ bits (which corresponds to $O(n^{20}/\varepsilon^{20})$ memory states in our setup). Their upper bound on the sample complexity was later improved by [2] to $O(k/\varepsilon^2 \cdot \mathrm{polylog}(1/\varepsilon))$ with space complexity of $O\left(\log\left(\frac{k}{\varepsilon}\right)\right)$ bits. In both the above works, the constant in the space complexity upper bound can be shown to actually be smaller than 20 by a careful analysis, but it cannot be made smaller than 2.

III. Preliminaries

In this section, we introduce mathematical notations and some relevant background for the paper.

A. Notations

We write $[n]$ to denote the set $\{1, \ldots, n\}$, and consider discrete distributions over $[n]$. We use the notation p_i to denote the probability of element i in distribution p. When X is a random variable on $[n]$, p_X denotes the random variable obtained by evaluating p in location X. The entropy of p is defined as $H(p) = -\sum_{x \in [n]} p_x \log p_x = \mathbb{E}_{X \sim p}(-\log p_X)$, where $H(p) = 0$ for a single mass distribution and $H(p) = \log n$ a uniform distribution over $[n]$. The total variation distance between distributions p and q is defined as half their ℓ^1 distance, i.e., $d_{\mathsf{TV}}(p, q) = \frac{1}{2}||p - q||_1 = \frac{1}{2}\sum_{i=1}^{n} |p_i - q_i|$, and their KL (Kullback–Leibler) divergence is defined as $D_{\mathrm{KL}}(p||q) = \sum_{i=1}^{n} p_i \log \frac{p_i}{q_i}$. Logarithms are taken to base 2.

B. Morris Counter

Suppose one wishes to implement a counter that counts up to m. Maintaining this counter exactly can be accomplished using $\log m$ bits. In the first example of a non-trivial streaming algorithm, Morris gave a randomized "approximate counter", which allows one to retrieve a constant multiplicative approximation to m with high probability using only $O(\log \log m)$ bits (see [3]). The Morris Counter was later analyzed in more detail by Flajolet [33], who showed that $O(\log \log m + \log(1/\varepsilon) + \log(1/\delta))$ bits of memory are sufficient to return a $(1 \pm \varepsilon)$ approximation with success probability $1 - \delta$. A recent result of [34] shows that $O(\log \log m + \log(1/\varepsilon) + \log \log(1/\delta))$ bits suffice for the same task.

The original Morris counter is a random state machine with the following simple structure: At each state $l = 1, 2, 3, \ldots,$ an increment causes the counter to transition to state $l+1$ with probability 2^{-l}, and to remain in state l with probability $1 - 2^{-l}$. This is formally a discrete time pure birth process, which can be seen in Figure 1.

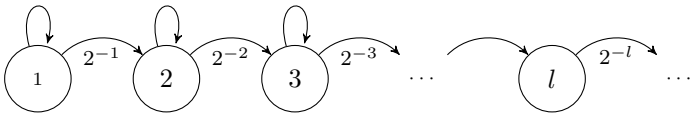

Figure 1: The original Morris counter

The performance of the above counter was characterized by Flajolet, who proved the following theorem.

Theorem 4 ([33])**.** *Let C_m be the value of the Morris counter after m increments. It holds that*

$$\mathbb{E}(C_m) = \log m + \mu + g(\log m) + \phi(m), \tag{11}$$

where $\mu \approx -0.273$ is a known constant, $g(\cdot)$ is a periodic function of amplitude less than 10^{-5}, $|\phi(m)| \leq \min\left\{1, \frac{2^{\sqrt{16\log m}\cdot(\log m)^{4.5}}}{2m}\right\}$ and the expectation is over the randomness of the counter.[2]

Thus, if we are interested in approximating $\log m$ using the counter, then taking our estimator to be $C_m - \mu$ guarantees that on average our additive error will not be more than $10^{-5} + \phi(m)$, a property that we leverage in our entropy estimation algorithm.

C. Finite-State Bias Estimation Machine

In the bias estimation problem, we are given access to i.i.d samples drawn from the $\mathsf{Bern}(p)$ distribution, and we wish to estimate the value of p under the expected quadratic loss (also known as mean squared error distortion measure). The S-state randomized machine with the state diagram depicted in Figure 2, was purposed by [10] and later carefully analyzed by [4], where it was shown to asymptotically induce a $\mathrm{Binomial}(S-1, \theta)$ stationary distribution on the memory state space. Thus, when this machine is initiated with a $\mathrm{Bern}(p)$ distribution and is run for a long enough time, it outputs an estimate $\hat{p}$ that has $\mathbb{E}(\hat{p} - p)^2 \leq \frac{1}{S-1}$. [4] further showed that the machine is order-optimal, by proving a lower bound of $\mathbb{E}(\hat{p} - p)^2 \geq \Omega(1/S)$ for any finite-state estimator.

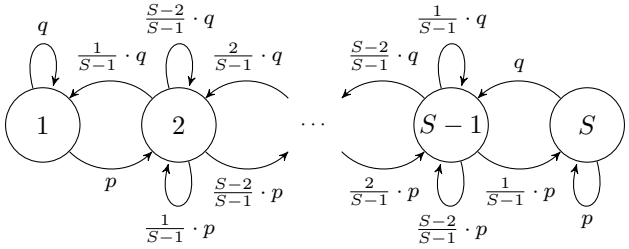

Figure 2: Randomized bias estimation machine with $q = 1 - p$

[2]In [33], Flajolet bounded $\phi(m)$ with $O(m^{-0.98})$. Here, we carefully follow the constants in his derivation and provide an explicit upper bound on the the error terms, since we are interested in bounds that can be applied for finite m.

IV. Upper Bound - Entropy Estimation Algorithm

In this section we prove Theorem 1, that is, we show the existence of an S-states randomized entropy estimation machine with $S \leq \frac{(c+1)^4 n \cdot (\log n)^4}{\beta^2 \delta}$ states, for $\varepsilon = 10^{-5} + \beta$ and n large enough. The basic idea is to let nature draw some X from p and use a Morris counter to approximate $-\log p_X$, then, since we are looking for $H(p) = \mathbb{E}(-\log p_X)$, use a bias estimation machine to simulate the averaging operation, by randomly generating coin tosses with bias that is proportionate to our estimate of $-\log p_X$. The bias estimation machine is incremented whenever a count is concluded in second Morris counter that simulates a clock, thus essentially averaging our estimates over x values. For a sufficiently large number of samples, this averaging converges (approximately) to the mean of $-\log p_X$, and thus outputs an approximation to the true underlying entropy. We divide our presentation to four parts: in the first part we describe the algorithm; in the second part we analyze the total number of states used by the algorithm; in the third part we assume the bias estimation machine is fed with an infinite number of i.i.d. samples and analyze the performance of the algorithm; and in the fourth part we relax this assumption by studying the mixing time of the Markovian process induced by our bias estimation machine. This allows us to prove an upper bound on the number of samples the developed algorithm requires.

A. Description of the algorithm

1) In each iteration of the algorithm we collect a fresh sample $X \sim p$, and store its value, which requires n states. Denote the realization of this random variable by x. We proceed to estimate $\log p_x$ based on more fresh samples, using Morris counters.
2) We use two Morris counters - one that approximates a clock, and one that approximates a count for x values:
   - The first counter has $M = \log T + 1$ states, where T is the smallest integral power of 2 larger than $\lceil n^c \rceil$ for some $c > 1$. The counter stops when we arrive at state M, which corresponds to some *random* time N. This counter essentially approximates a clock that counts until $\lceil n^c \rceil$ samples from p are observed and it uses at most $c \log n + 2$ states.
   - The second counter runs in parallel to the first one and approximates the logarithm of the number of observed x values. It stops when the first counter arrived at state M and output its value, which we denote as C_{N_x}. This counter also has M states. The randomness used for the two counters is statistically independent. In the event that the second counter reached state M before the first one, it also stops and outputs the value M.[3]
3) Denoting the number of observed x values in the previous stage as N_x, we define $C_{N_x}^{\text{centered}} \triangleq C_{N_x} - \mu - \mathbb{E}\log N$ to be the centralized output of the second counter. As we argue below, this is an almost unbiased estimator for $-\log p_x$.

We now use the bias estimation machine with $S_{\text{bias}} = \frac{M^2}{\beta^2 \delta} + 1$ states to simulate the expectation operation, where $\beta > 0$ is such that $\varepsilon = 10^{-5} + \beta$. Specifically, each time the first Morris counter concludes a count, we generate a $\text{Ber}(\theta_{N_x})$ random variable, with $\theta_{N_x} = -\frac{C_{N_x}^{\text{centered}}}{M} + a$, and use it as the input to our bias estimation machine. The offset $a \triangleq 1 - \frac{\mathbb{E}(\log N) + \mu}{M}$ guarantees that $\theta_{N_x} \in [0, 1)$ with probability 1, as $\theta_{N_x} = 1 - \frac{C_{N_x}}{M}$ and $1 \leq C_{N_x} \leq M$ as it is the output of a Morris counter with M states. Our estimator for the entropy $\hat{H}$ is the bias estimate of the machine, after subtraction of the known offset a and multiplication by M, that is, $\hat{H} = M(\hat{\theta} - a)$.

[3]We ignore this case in the analysis since its contribution to the estimation error can be shown to have a negligible effect. This follows since if $p_x \leq 1 - n^{-(c-1)}$ the probability of this event is extremely small, otherwise, the contribution of p_x to the entropy is negligible.

B. Number of states in our machine

In each time point, our algorithm keeps the value of x, the state of the Morris counter approximating the clock, the state of the Morris counter approximating the logarithm of the x counter, and the state of the bias estimation machine. Thus, calculating the product of the number of states needed at each step, and recalling that $M = \log T + 1 \leq (c+1)\log n$, the total number of states is

$$S = n \cdot M \cdot M \cdot S_{\text{bias}} = n \cdot M^2 \cdot \left(\frac{M^2}{\beta^2 \delta} + 1\right) \leq \frac{(c+1)^4 n \cdot (\log n)^4}{\beta^2 \delta}. \tag{12}$$

C. Analysis of the algorithm

We first show that the random run time of the first Morris counter N is concentrated around $\lceil n^c \rceil$, by upper bounding $\Pr(N < m)$ in Lemma 1, and then using it to upper bound $\mathbb{E}\left(N^{-\alpha}\right)$ in Lemma 2. We then show that $\mathbb{E}(C_{N_x}^{\text{centered}})$ is close to $H(p)$ in expectation in Lemma 3. We conclude by showing in Lemma 6 that the output of the bias estimation machine converges to the expectation of θ_{N_X}, which then implies that $\hat{H} = M(\hat{\theta} - a)$ is an ε-additive estimator of $H(p)$ with probability at least $1-\delta$. Expectations are taken with respect to the randomness at different stages of the algorithm, and we implicitly state the random variables with respect to which the expectations are taken, apart for the case where the expectation is taken over all the randomness of the algorithm.

Lemma 1. *For $m = 2^\ell, 1 \leq \ell \leq M-1$, it holds that*

$$\Pr(N < m) \leq e \cdot 2^{-\frac{1}{2}\cdot(M-\ell-1)^2} \tag{13}$$

Proof. let τ_k be the time it takes to move from state k to state $k+1$ in the first Morris counter and note that $\tau_k \sim \text{Geo}(2^{-k})$. Clearly, $N = \sum_{k=1}^{M-1} \tau_k$. The moment generating function of τ_k is

$$\mathbb{M}_{\tau_k}(s) \triangleq \mathbb{E}\left(e^{s\tau_k}\right) = \frac{1}{1 + 2^k(e^{-s} - 1)}, \tag{14}$$

and it is defined for all $s < -\ln(1 - 2^{-k})$. The moment generating function of N is therefore

$$\mathbb{M}_N(s) = \prod_{k=1}^{M-1} \mathbb{M}_{\tau_k}(s) = \prod_{k=1}^{M-1} \frac{1}{1 + 2^k(e^{-s} - 1)}. \tag{15}$$

From Chernoff bound, it holds that $\Pr(N < m) \leq e^{-sm} \cdot \mathbb{M}_N(s)$ for any $s < 0$. Setting $s = -\ln(1 + 1/m)$, we get

$$\Pr(N < m) \leq \left(1 + \frac{1}{m}\right)^m \cdot \prod_{k=1}^{M-1} \frac{1}{1 + \frac{2^k}{m}} \leq e \cdot 2^{-\sum_{k=1}^{M-1} \log\left(1 + \frac{2^k}{m}\right)}. \tag{16}$$

Now let $m = 2^\ell$ for $1 \leq \ell \leq M-1$. We conclude by lower bounding the exponent:

$$\sum_{k=1}^{M-1} \log(1 + 2^{k-\ell}) = \sum_{k=0}^{\ell-1} \log(1 + 2^{-k}) + \sum_{k=1}^{M-\ell-1} \log(1 + 2^k) \tag{17}$$

$$\geq \sum_{k=1}^{M-\ell-1} k \tag{18}$$

$$\geq \frac{1}{2} \cdot (M - \ell - 1)^2. \tag{19}$$

□

Lemma 2. *For any $0 < \alpha \leq 1$, we have that*

$$\mathbb{E}\left(N^{-\alpha}\right) \leq (e+1)n^{-c\cdot\alpha+v_n(\alpha)}, \tag{20}$$

where $v_n(\alpha) \triangleq \sqrt{\frac{\alpha^3 c}{\log n}}$.

Proof. Appealing to Lemma 1, for any $1 \leq \ell \leq M-1, m = 2^\ell$, we have

$$\mathbb{E}(N^{-\alpha}) \leq \Pr(N < m) + m^{-\alpha} \cdot \Pr(N \geq m) \tag{21}$$

$$\leq e \cdot 2^{-\frac{1}{2}\cdot(M-\ell-1)^2} + 2^{-\ell\cdot\alpha}. \tag{22}$$

Setting $\ell = M - 1 - \sqrt{2\alpha\cdot(M-1)}$ and recalling that $n^c \leq 2^{M-1}$, we get

$$\mathbb{E}(N^{-\alpha}) \leq e\cdot 2^{-\alpha(M-1)} + 2^{-\alpha(M-1)\cdot\left(1-\sqrt{\frac{2\alpha}{M-1}}\right)} \leq e\cdot n^{-c\cdot\alpha} + n^{-c\cdot\alpha\left(1-\sqrt{\frac{\alpha}{c\log n}}\right)}. \tag{23}$$

□

Lemma 3. *Let $\psi_c(n) = \min\left\{1 + (e+1)n^{-(c-1)+v_n(1)}, C\cdot n^{-\frac{1}{2}\cdot(c-1)+v_n(1/2)}\right\}$, where $C = 2(e+1)\cdot 10^8$. Then*

$$|\,\mathbb{E}(C_{N_X}^{\textit{centered}}) + H(p)| \leq 10^{-5} + \psi_c(n). \tag{24}$$

Proof. According to Theorem 4, the value of the Morris counter after m updates is close to $\log m$ in expectation, up to some small bias. Using this fact we show that, given N and $X = x$, $\mathbb{E}(C_{N_X}^{\text{centered}})$ is close to the expected logarithm of a normalized Binomial(N, p_x) random variable. Taking the expectation over X, this gives us $-H(p)$ plus some bias. Write

$$\mathbb{E}(C_{N_X}^{\text{centered}}) = \mathbb{E}(\mathbb{E}_{C_{N_X}|_{X=x,N}}(C_{N_x} - \mu - \mathbb{E}(\log N))) \tag{25}$$

$$= \mathbb{E}(\log N_X - \mathbb{E}(\log N) + g(\log N_X) + \phi(N_X)) \tag{26}$$

$$= \mathbb{E}\left(\log\frac{N_X}{N}\right) + \mathbb{E}(\gamma_{N_X}) \tag{27}$$

$$= \mathbb{E}(\log p_X) + \mathbb{E}\left(\log\frac{N_X}{N\cdot p_X}\right) + \mathbb{E}(\gamma_{N_X}) \tag{28}$$

$$= -H(p) + \mathbb{E}\left(\log\frac{N_X}{N\cdot p_X}\right) + \mathbb{E}(\gamma_{N_X}), \tag{29}$$

where $\gamma_{N_x} = g(\log N_x) + \phi(N_x)$. We conclude the proof by showing that $\mathbb{E}\left(\log\frac{N_X}{N\cdot p_X}\right)$ is small in Lemma 4, and showing that $\mathbb{E}(\gamma_{N_X})$ is small in Lemma 5. □

Lemma 4. *It holds that*

$$0 \leq \mathbb{E}\left(\log\frac{N_X}{N\cdot p_X}\right) \leq (e+1)n^{-(c-1)+v_n(1)}. \tag{30}$$

Proof. We first show that $\mathbb{E}\left(\log\frac{N_X}{N\cdot p_X}\right) \geq 0$. By Jensen's inequality and convexity of $t \mapsto -\log(t)$

$$\mathbb{E}\left(\log\frac{N_X}{N\cdot p_X}\right) = \mathbb{E}_{X,N}\left[\mathbb{E}_{N_X|N,X}\left(-\log\frac{N\cdot p_X}{N_X}\right)\right] \tag{31}$$

$$\geq -\,\mathbb{E}_{X,N}\left[\log\left(N\cdot p_X\cdot\mathbb{E}_{N_X|N,X}\left[\frac{1}{N_X}\right]\right)\right]. \tag{32}$$

To establish non-negativity of $\mathbb{E}\left(\log \frac{N_X}{N \cdot p_X}\right)$, it therefore suffices to show that $\mathbb{E}_{N_X|N,X=x}\left[\frac{1}{N_X}\right] \leq \frac{1}{p_x \cdot N}$. To that end, recall that given $X = x$ and N, we have $N_X \sim \mathrm{Bin}(N-1, p_x) + 1$. Thus, we indeed have

$$\begin{aligned}
\mathbb{E}_{N_X|N,X=x}\left[\frac{1}{N_X}\right] &= \sum_{m=0}^{N-1} \frac{1}{m+1}\binom{N-1}{m} p_x^m (1-p_x)^{N-m-1} \\
&= \sum_{m=0}^{N-1} \frac{1}{p_x \cdot N}\binom{N}{m+1} p_x^{m+1} (1-p_x)^{N-m-1} \\
&= \frac{1-(1-p_x)^N}{p_x \cdot N} \\
&\leq \frac{1}{p_x \cdot N}. \qquad (33)
\end{aligned}$$

To upper bound $\mathbb{E}\left(\log \frac{N_X}{N \cdot p_X}\right)$, we use Jensen's inequality and the concavity of $t \mapsto \log t$, to obtain

$$\begin{aligned}
\mathbb{E}_{N_X|N,X=x}\left(\log \frac{N_X}{N \cdot p_X}\right) &\leq \log\left(\frac{\mathbb{E}_{N_X|N,X=x}[N_X]}{N \cdot p_x}\right) \qquad (34) \\
&= \log\left(1 + \frac{1-p_x}{N \cdot p_x}\right) \qquad (35) \\
&\leq \frac{1}{N \cdot p_x}. \qquad (36)
\end{aligned}$$

Thus, overall,

$$\begin{aligned}
\mathbb{E}\left[\log \frac{N_X}{N \cdot p_X}\right] &\leq \mathbb{E}_{N,X}\left[\frac{1}{N \cdot p_X}\right] \\
&= \mathbb{E}_X\left[\frac{1}{p_X}\right] \mathbb{E}_N\left[\frac{1}{N}\right] \\
&= n \cdot \mathbb{E}_N\left[\frac{1}{N}\right]
\end{aligned}$$

and appealing to Lemma 2 with $\alpha = 1$, we have $\mathbb{E}\left[\log \frac{N_X}{N \cdot p_X}\right] \leq (e+1) n^{-(c-1)+v_n(1)}$. □

Lemma 5. *It holds that*

$$\mathbb{E}(\gamma_{N_X}) \leq 10^{-5} + \min\{1, C \cdot n^{-\frac{1}{2}\cdot(c-1)+v_n(1/2)}\}. \qquad (37)$$

Proof. Note that $\mathbb{E}(g(\log N_x)) \leq 10^{-5}$ is explicit in Theorem 4 for any $x \in [n]$, and in particular, $\mathbb{E}(g(\log N_X)) \leq 10^{-5}$. Thus, it remains to upper bound $\mathbb{E}(\phi(N_X))$. It is straightforward to verify that $\phi(x) \leq \min\left\{1, \frac{2\cdot 10^8}{\sqrt{x}}\right\}$ for all $x \geq 1$, and consequently,

$$\mathbb{E}(\phi(N_X)) \leq \mathbb{E}\left[\min\left\{1, \frac{2 \cdot 10^8}{\sqrt{N_X}}\right\}\right] \leq \min\left\{1, 2 \cdot 10^8\, \mathbb{E}\left[\sqrt{\frac{1}{N_x}}\right]\right\}. \qquad (38)$$

From Jensen's inequality, concavity of $t \mapsto \sqrt{t}$, and equation (33),

$$\mathbb{E}\left[\sqrt{\frac{1}{N_X}}\right] = \mathbb{E}_{N,X}\left[\mathbb{E}_{N_X|N,X}\left[\sqrt{\frac{1}{N_X}}\right]\right] \tag{39}$$

$$\leq \mathbb{E}_{N,X}\left[\sqrt{\mathbb{E}_{N_X|N,X}\left[\frac{1}{N_X}\right]}\right] \tag{40}$$

$$\leq \mathbb{E}_{N,X}\left[\sqrt{\frac{1}{p_X \cdot N}}\right]$$

$$= \mathbb{E}_N\left[\sqrt{\frac{1}{N}}\right]\mathbb{E}_X\left[\sqrt{\frac{1}{p_X}}\right]. \tag{41}$$

Note that, again using Jensen's inequality and concavity of $t \mapsto \sqrt{t}$, we have

$$\mathbb{E}_X\left[\sqrt{\frac{1}{p_X}}\right] = \sum_{x=1}^{n}\sqrt{p_x} \leq n\sqrt{\frac{1}{n}\sum_{x=1}^{n}p_x} = \sqrt{n}. \tag{42}$$

Appealing to Lemma 2 with $\alpha = 0.5$, we have

$$\mathbb{E}(N^{-0.5}) \leq (e+1)n^{-\frac{c}{2}+v_n(1/2)}. \tag{43}$$

Thus, substituting (42) and (43) into (41) and then into (38), and recalling that $C = 2(e+1)10^8$, we obtain the claimed result. □

Lemma 6. *Define the estimator $\hat{H} = M(\hat{\theta} - a)$, where $\hat{\theta}$ is the output of the bias estimation machine with $S_{bias} = \frac{M^2}{\beta^2\delta} + 1$ states, to which we feed a $\mathsf{Bern}(\theta_{N_X})$ sample at each iteration of our approximate counter. Then,*

$$\Pr(|\hat{H} - (H(p) + b)| > \beta) \leq \delta, \tag{44}$$

where $|b| \leq 10^{-5} + \psi_c(n)$.

Proof. Averaging over X, we have that the overall bias of the binary random variable we feed to the bias estimation machine is

$$\theta \triangleq \mathbb{E}(\theta_{N_X}) = \mathbb{E}\left(-\frac{C_{N_X}^{\text{centered}}}{M} + a\right) = \frac{H(p) + b}{M} + a, \tag{45}$$

where $|b| \leq 10^{-5} + \psi_c(n)$ is an unknown offset that arises from Lemma 3. Recalling that $\mathbb{E}(\hat{\theta} - \theta)^2 \leq \frac{1}{S_{\text{bias}}-1}$,

$$\mathbb{E}(\hat{H} - (H(p) + b))^2 = M^2 \cdot \mathbb{E}(\hat{\theta} - \theta)^2 \leq \beta^2\delta, \tag{46}$$

thus we have from Chebyshev's inequality,

$$\Pr(|\hat{H} - (H(p) + b)| > \beta) \leq \frac{\mathbb{E}(\hat{H} - (H(p) + b))^2}{\beta^2} \leq \delta. \tag{47}$$

Note that our upper bound on the additive error in estimation of $H(p)$ is $\beta + |b| \leq \beta + 10^{-5} + \psi_c(n)$, which limits our results to estimation error $\varepsilon > 10^{-5} + \psi_c(n)$. □

D. Sample complexity of our algorithm

In our algorithm, the number of observed samples is unbounded. In practice we only need to observe $O(t_{\mathsf{mix}}(p))$ samples, where $t_{\mathsf{mix}}(p)$ is the mixing time of our machine whenever the input is $\mathrm{Bern}(p)$ samples, i.e., the minimal time it takes for the total variation distance between the marginal distribution and the limiting distribution to be

small. In our case the bias estimation machine is only incremented after an iteration of the first Morris counter is completed, and the run time of each iteration is a random variable that is only bounded in expectation. However, we note that this in fact implies the existence of a good algorithm that has a bounded sample complexity; namely, running our entropy estimation algorithm on L samples is equivalent to running the bias estimation machine from [10] on a random number of samples $k = k(L)$ times with $p = \theta = \mathbb{E}(\theta_{N_X})$. The randomness in $k(L)$ follows since the runtime N_i of each iteration of the Morris counter procedure is a random variable. Below, we use Chernoff's bound to upper bound the probability that $k(L)$ is small. This event is considered as an error in our analysis. We now upper bound the mixing time of the bias estimation machine from [10]. Whenever $k(L)$ is greater than this mixing time, the error of our algorithms with L samples is close to its asymptotic value.

To upper bound the mixing time, we use the *coupling method*. Recall that the transition matrix P of a Markov process $\{X_t\}_{t=0}^{\infty}$ supported on $\mathcal{X}$ is a matrix whose elements are $\Pr(X_{t+1} = x'|X_t = x) = P(x, x')$, for any $x, x' \in \mathcal{X} \times \mathcal{X}$. We define a coupling of Markov chains with transition matrix P to be a process $\{X_t, Y_t\}_{t=0}^{\infty}$ with the property that both $\{X_t\}_{t=0}^{\infty}$ and $\{Y_t\}_{t=0}^{\infty}$ are Markov chains with transition matrix P, although the two chains may be correlated and have different initial distributions. Given a Markov chain on $\mathcal{X}$ with transition matrix P, a *Markovian coupling* of two P-chains is a Markov chain $\{X_t, Y_t\}_{t=0}^{\infty}$ with state space $\mathcal{X} \times \mathcal{X}$, which satisfies, for all x, y, x', y',

$$\Pr(X_{t+1} = x'|X_t = x, Y_t = y) = P(x, x') \tag{48}$$

$$\Pr(Y_{t+1} = y'|X_t = x, Y_t = y) = P(y, y'). \tag{49}$$

Let $P^t(x_0)$ be the marginal distribution of the chain at time t when initiated at x_0, and let π be the unique stationary distribution. Define the δ-mixing time as

$$t_{\delta}^{*} \triangleq \min\{t : d_{\mathsf{TV}}(P^t(x_0), \pi) \leq \delta\}, \tag{50}$$

and $t_{\mathsf{mix}} \triangleq t_{1/4}^{*}$. We now show that the bias estimation machine with S states mixes in $\Theta(S \log S)$ time, uniformly for all $p \in (0, 1]$.

Theorem 5. *Let $t_{\mathsf{mix}}(p)$ denote the mixing time of the bias estimation machine with S states when the input is i.i.d $\mathsf{Bern}(p)$, and define the worst-case mixing time to be $t^* = \max_{p\in(0,1]} t_{\mathsf{mix}}(p)$. Then*

$$\ln(2) \cdot (S-1)\log(S-1) \leq t^* \leq 4S \log S. \tag{51}$$

Proof. The transition probabilities of the bias estimation machine of Figure 2 are given, for $1 < k < S$, as

$$X_{t+1}|_{X_t=k} = \begin{cases} k+1, & \textit{w.p. } \frac{S-k}{S-1} \cdot p, \\ k, & \textit{w.p. } \frac{k-1}{S-1} \cdot p + \frac{S-k}{S-1} \cdot q, \\ k-1, & \textit{w.p. } \frac{k-1}{S-1} \cdot q, \end{cases} \tag{52}$$

and for the extreme states $\{1, S\}$ as

$$X_{t+1}|_{X_t=1} = \begin{cases} 2, & \textit{w.p. } p, \\ 1, & \textit{w.p. } q, \end{cases} \quad X_{t+1}|_{X_t=S} = \begin{cases} S, & \textit{w.p. } p, \\ S-1, & \textit{w.p. } q. \end{cases} \tag{53}$$

We construct a Markovian coupling in which the two chains stay together at all times after their first simultaneous visit to a single state, that is

$$\text{if } X_s = Y_s \text{ then } X_t = Y_t \text{ for all } t \geq s. \tag{54}$$

The following lemma, due to [35](Theorem 5.4), will give us an upper bound on the mixing time using this coupling.

Lemma 7. *Let* $\{(X_t, Y_t)\}$ *be a Markovian coupling satisfying* (54)*, for which* $X_0 = x_0$ *and* $Y_0 = y_0$*. Let* τ_{couple} *be the coalescence time of the chains, that is,*

$$\tau_{\mathsf{couple}} \triangleq \min\{t : X_t = Y_t\}. \tag{55}$$

Then

$$t_{\mathsf{mix}} \leq 4 \max_{x_0, y_0 \in \mathcal{X}} \mathbb{E}(\tau_{\mathsf{couple}}). \tag{56}$$

Assume w.l.o.g. that $x_0 < y_0$ and let U_t be an i.i.d sequence drawn according to the $\mathrm{Unif}(0,1)$ distribution. We construct a coupling on (X_t, Y_t) such that, at each time point $t < \tau_{\mathsf{couple}}$, X_t and Y_t are incremented in the following manner:

$$X_{t+1}|_{X_t=i} = \begin{cases} i+1, & \text{if } U_t \leq \frac{S-i}{S-1} \cdot p, \\ i, & \text{if } \frac{S-i}{S-1} \cdot p \leq U_t \leq 1 - \frac{i-1}{S-1} \cdot q, \\ i-1, & \text{if } 1 - \frac{i-1}{S-1} \cdot q \leq U_t \leq 1, \end{cases} \tag{57}$$

and

$$Y_{t+1}|_{Y_t=j} = \begin{cases} j+1, & \text{if } U_t \leq \frac{S-j}{S-1} \cdot p, \\ j, & \text{if } \frac{S-j}{S-1} \cdot p \leq U_t \leq 1 - \frac{j-1}{S-1} \cdot q, \\ j-1, & \text{if } 1 - \frac{j-1}{S-1} \cdot q \leq U_t \leq 1. \end{cases} \tag{58}$$

One can validate that the transition probabilities are the correct ones, for example

$$\Pr(X_{t+1} = i | X_t = i) = \Pr\left(\frac{S-i}{S-1} \cdot p \leq U_t \leq 1 - \frac{i-1}{S-1} \cdot q\right) \tag{59}$$

$$= 1 - \frac{i-1}{S-1} \cdot q - \frac{S-i}{S-1} \cdot p \tag{60}$$

$$= \frac{i-1}{S-1} \cdot p + \frac{S-i}{S-1} \cdot q, \tag{61}$$

and, similarly, $\Pr(Y_{t+1} = j | Y_t = j) = \frac{j-1}{S-1} \cdot p + \frac{S-j}{S-1} \cdot q$. The other transition probabilities are easily calculated. Note that $i < j$ implies $\frac{S-j}{S-1} < \frac{S-i}{S-1}$, thus Y_t cannot move right unless X_t moves right and X_t cannot move left unless Y_t moves left. Moreover, since $x_0 < y_0$, we have $i < j$ for all $t < \tau_{\mathsf{couple}}$. This follows from construction, since $\frac{S-i}{S-1} \cdot p$ is always smaller than $1 - \frac{j-1}{S-1} \cdot q$, implying that X_t cannot jump over Y_t when they are one-state apart. Thus, the *distance* process $D_t \triangleq Y_t - X_t$, is a non-increasing function of t, with initial state $D_0 = y_0 - x_0$, that can only decrease by one unit at a time or stay unchanged. We have

$$\Pr(D_{t+1} = D_t - 1) = \Pr(X_{t+1} = X_t + 1, Y_{t=1} = Y_t) + \Pr(Y_{t+1} = Y_t - 1, X_{t=1} = X_t) \tag{62}$$

$$= \Pr\left(\frac{S-Y_t}{S-1} \cdot p \leq U_t \leq \frac{S-X_t}{S-1} \cdot p\right) + \Pr\left(1 - \frac{Y_t-1}{S-1} \cdot q \leq U_t \leq 1 - \frac{X_t-1}{S-1} \cdot q\right) \tag{63}$$

$$= \frac{Y_t - X_t}{S-1} \cdot p + \frac{Y_t - X_t}{S-1} \cdot q \tag{64}$$

$$= \frac{D_t}{S-1}. \tag{65}$$

The expected coupling time is now the expected time it takes for D_t to decrease from D_0 to D_t, thus in order to

maximize it under the given coupling, we need to maximize D_0, which corresponds to setting $X_0 = 1, y_0 = S$. For $D_0 = S-1$, consider the process $M_t \triangleq D_0 - D_t$, which is a non-decreasing function of t that goes from 0 to $S-1$ and has $\Pr(M_{t+1} = M_t + 1) = \Pr(D_{t+1} = D_t - 1) = \frac{D_t}{S-1} = 1 - \frac{M_t}{S-1}$. Then this process is no other than the *Coupon Collector* process with $S-1$ coupons, and the expected coupling time in our chain in identical to the expected number of coupons collected until the set contains all $S-1$ types, which according to [35], Proposition 2.3., is

$$\mathbb{E}(\tau_{\mathsf{couple}}) = (S-1) \cdot \sum_{k=1}^{S-1} \frac{1}{k} \leq (S-1)(\ln(S-1)+1) \leq S\log(S). \tag{66}$$

To show that this upper bound is indeed tight, consider the case of $p = 1$. In this case, the chain of Figure 2 is simply the Coupon Collector process with $S-1$ coupons, thus, letting τ be the (random) time it takes to collect all coupons, we have

$$\mathbb{E}(\tau) = (S-1) \cdot \sum_{k=1}^{S-1} \frac{1}{k} \geq \ln(2) \cdot (S-1)\log(S-1). \tag{67}$$

□

From [35], Eq. (4.34), we have that the δ-mixing time t^*_δ can be upper bounded in terms on the mixing time by

$$t^*_\delta \leq \left\lceil \log\left(\frac{1}{\delta}\right) \right\rceil \cdot t_{\mathsf{mix}}. \tag{68}$$

Let

$$k \triangleq 8\log\left(\frac{1}{\delta}\right) \frac{(\log 4n^c)^2}{\beta^2\delta} \log\left(\frac{(\log 4n^c)^2}{\beta^2\delta}\right), \tag{69}$$

and note that from equation (68), Theorem 5, and substituting $S_{\text{bias}} = \frac{M^2}{\beta^2\delta} + 1$, we have that the δ-mixing time of the bias estimation machine is at most k. Let $N_1, N_2, \ldots, N_k$ be the first k i.i.d. Morris counter running times, which are all distributed as N in the analysis from Section IV. Lemma 8 uses the concentration of N to show that, with probability $1-\delta$, the number of samples we need to observe until the bias machine mixes is not large.

Lemma 8. *Let* $m = 4n^c \cdot \ln\left(\frac{5k}{\delta}\right)$. *Then*

$$\Pr\left(\sum_{i=1}^{k} N_i > k \cdot m\right) \leq \delta. \tag{70}$$

Proof. Recall that for all $i \in \mathbb{N}$, we have

$$\mathbb{M}_{N_i}(s) = \prod_{j=1}^{M-1} \frac{1}{1 + 2^j(e^{-s} - 1)}, \tag{71}$$

which is defined for all $s < -\ln(1 - 2^{-(M-1)}) = -\ln(1 - 1/T)$. As $T \leq 2n^c$, $\mathbb{M}_{N_i}(t)$ is well defined for

$s = -\ln(1 - 1/4n^c)$, thus we have from Chernoff bound,

$$\Pr(N_i > m) \le e^{-sm} \cdot \mathbb{M}_{N_i}(s) \tag{72}$$

$$= \left(1 - \frac{1}{4n^c}\right)^m \cdot \prod_{j=1}^{M-1} \frac{1}{1 - \frac{2^j}{4n^c}} \tag{73}$$

$$\le \left(1 - \frac{1}{4n^c}\right)^m \cdot \prod_{j=1}^{M-1} \frac{1}{1 - 2^{-j}} \tag{74}$$

$$\le 5 \exp\left\{-\frac{m}{4n^c}\right\} = \frac{\delta}{k} \tag{75}$$

where in (74) we used the fact that $\frac{2^j}{4n^c} \le \frac{2^j}{2T} = 2^{j-M}$, and in (75) we used the bound $\prod_{j=1}^{M}(1-2^{-j}) \ge \frac{1}{4} + \frac{1}{2^{M+1}}$, which can be proved via induction. Consequently, the probability that at least one of the random variables $N_1, \ldots, N_k$ is greater than m is at most $1 - \left(1 - \frac{\delta}{k}\right)^k \le \delta$, which implies the statement of the lemma. □

We conclude with the following lemma, which connects Theorem 5 and Lemma 8 to show that our entropy estimator performs well even if the number of input samples is $\tilde{O}(n^c/\delta)$.

Lemma 9. *Let the algorithm of Theorem 1 run on $L = k \cdot m$ samples, and output the estimate $\hat{H}_{M_L}$. Then with probability at least $1 - 3\delta$, $\hat{H}_{M_L}$ is within ε-additive error from $H(p)$.*

Proof. Lemma 8 implies that, with probability at least $1 - \delta$, after observing $k \cdot m$ samples, the bias estimation machine has been incremented at least k times. Recall that, by definition, after $t \ge t^*_\delta$ increments of the bias estimation machine, we have that $d_{\mathsf{TV}}(P^t(x_0), \pi) \le \delta$, and that our S-states entropy estimator has $\sum_{i \in \hat{H}_{\bar{\varepsilon}}} \pi_i < \delta$, where $\hat{H}_{\bar{\varepsilon}} = \{i \in [S] : |\hat{H}_i - H(p)| > \varepsilon\}$. Thus, from a union bound, a fraction of 2δ of the distribution $P^t(x_0)$ (at most) is supported on $\hat{H}_{\bar{\varepsilon}}$. Putting it all together, we have that a finite-time algorithm that outputs an estimate $\hat{H}(M_L)$ after

$$L = k \cdot m = \Omega\left(\frac{n^c \cdot \mathrm{poly}(\log n)}{\delta} \cdot \mathrm{poly}(\log(1/\delta))\right) \tag{76}$$

will be ε-far from the correct entropy with probability at most 3δ. □

V. Lower Bounds

In this section we prove Theorem 2. The $\Omega(n)$ bound is proved via reduction to uniformity testing. For the $\frac{\log n}{2\varepsilon}$ bound, we use a simple quantization argument. Assume that $S < \frac{\log n}{2\varepsilon}$. Then there must be two consecutive estimate values $\hat{H}_1, \hat{H}_2 \in [0, \log n]$ such that $\hat{H}_2 - \hat{H}_1 > 2\varepsilon$. This implies that $H = (\hat{H}_1 + \hat{H}_2)/2$ has $|H - \hat{H}_1| = |H - \hat{H}_2| > \varepsilon$. Thus, for this value of the entropy, we have $\Pr(|\hat{H}(M_t) - H| > \varepsilon) = 1$ for all $t \in \mathbb{N}$.

A. Proof of the $(1 - 2\sqrt{\varepsilon \ln 2})n$ bound

An (ε, δ) uniformity tester can distinguish (with probability $0 < \delta < 1/2$) between the case where p is uniform and the case where p is ε-far from uniform in total variation. Assume we have an (ε, δ) entropy estimator. Then we can obtain an $(\tilde{\varepsilon} = \sqrt{\varepsilon \ln 2}, \delta)$ uniformity tester using the following protocol: the tester declares that p is uniform if $\hat{H} > \log n - \varepsilon$, and that p is $\tilde{\varepsilon}$-far from uniform if $\hat{H} < \log n - \varepsilon$. We now argue that this is indeed an $(\tilde{\varepsilon}, \delta)$ uniformity tester, in which case the $(1 - 2\tilde{\varepsilon})n$ lower bound will follow immediately from the lower bound on uniformity testing of [7]. If $p = u$, where u is the uniform distribution over $[n]$, then $H(p) = \log n$ and $\hat{H} > \log n - \varepsilon$ with probability

at least $1-\delta$, so our tester will correctly declare "uniform" with probability at least $1-\delta$. If $d_{\mathsf{TV}}(p,u) > \sqrt{\varepsilon \ln 2}$, then from Pinsker's inequality ([36], Lemma 11.6.1),

$$2\varepsilon < \frac{2}{\ln 2} d_{\mathsf{TV}}(p,u)^2 \le D(p||u) = \log n - H(p), \tag{77}$$

which implies $H(p) < \log n - 2\varepsilon$ and $\hat{H} < \log n - \varepsilon$ with probability at least $1-\delta$. Thus, our tester will correctly declare "far from uniform" with probability at least $1-\delta$.

VI. Memory Complexity of Mutual Information Estimation

We extend our results to the problem of mutual information estimation. The upper bound follows by a slight tweaking of our entropy estimation machine, and the lower bound follows by noting the close relation between mutual information and joint entropy, and lower bounding the memory complexity of the latter.

A. Sketch of Upper Bound achieving algorithm

1) Let nature randomly draw some $(X,Y) = (x,y) \in [n] \times [m]$ according to p_{XY}, and keep that value of (x,y) for the following stages. This requires $n \cdot m$ states.
2) We use *four* Morris counters - one that approximates a clock, one that approximates a count for x values, one that approximates a count for y values, and one that approximates a count for the pair (x,y):
   - The first counter has $M = \log T + 1$ states, where T is the next power of 2 after $\lceil (n \cdot m)^c \rceil$. The random stopping time of the counter is N. This counter approximates a clock that counts until $\lceil (n \cdot m)^c \rceil$ samples from the distribution are observed. This takes at most $c \log(n \cdot m) + 2$ states.
   - The second, third and fourth counters run in parallel to the first one and approximate a counter for x, a counter for y, and a counter for the pair (x,y), and we denote their outputs as C_{N_x}, C_{N_y} and $C_{N_{xy}}$, respectively. Each of these counters contains M states.
3) Let $C_{\text{MI}} = C_{N_x} + C_{N_y} - C_{N_{xy}}$, and define $C_{\text{MI}}^{\text{centered}} = C_{\text{MI}} - \mu - \mathbb{E}\log N$. For the reasons outlined in Section IV, the expectation of $C_{\text{MI}}^{\text{centered}}$ is equal to $-\log p_x - \log p_y + \log p_{xy}$ plus some bias
$$|b_{\text{MI}}| \le 3 \cdot 10^{-5} + O\left(n^{-\frac{1}{2}\cdot(c-1)} \vee m^{-\frac{1}{2}\cdot(c-1)} \right). \tag{78}$$
4) We simulate the expectation operation using a bias estimation machine with $S_{\text{bias}} = \frac{9M^2}{\beta^2\delta} + 1$ states. Let $\theta_{N_{xy}} = -\frac{C_{\text{MI}}^{\text{centered}}}{3M} + a$, where $a \triangleq \frac{2M - \mathbb{E}(\log N) - \mu}{3M}$ is a known offset that we add to guarantee that $\theta_{N_{xy}} \in [0,1)$ with probability 1. Averaging over $(X,Y) = (x,y)$, our bias is $\theta_{\text{MI}} = \mathbb{E}(\theta_{N_{XY}}) = \frac{I(X;Y) + b_{\text{MI}}}{3M} + a$. After subtraction of a and multiplication by $3M$, that is, setting $\hat{I} = 3M(\hat{\theta}_{\text{MI}} - a)$ we get an (almost) unbiased estimator for $I(X;Y)$. As we have
$$\mathbb{E}(\hat{I} - (I(X;Y) + b_{\text{MI}}))^2 = 9M^2 \cdot \mathbb{E}(\hat{\theta}_{\text{MI}} - \theta_{\text{MI}})^2 \le \beta^2\delta, \tag{79}$$
Chebyshev's inequality implies that $\Pr(|\hat{I} - (I(X;Y) + b_{\text{MI}})| > \beta) \le \delta$. The total number of states is thus the product of the number of states at each step,
$$S = nm \cdot M^4 \cdot \left(\frac{9M^2}{\beta^2\delta} + 1 \right) \le \frac{9nm \cdot (c+1)^6 (\log nm)^6}{\beta^2\delta}, \tag{80}$$
for $\beta > 0$, $\varepsilon = 3 \cdot 10^{-5} + \beta + O\left(n^{-\frac{1}{2}\cdot(c-1)} \vee m^{-\frac{1}{2}\cdot(c-1)} \right)$.

B. Lower Bound

For simplicity of proof, let $\varepsilon, \delta \geq \frac{1}{100}$, and recall that $\varepsilon < \frac{1}{12 \ln 2}$. Our lower bound from Theorem 2 implies that for joint entropy estimation of $H(X,Y)$ where $(X,Y) \in [n] \times [m]$, the memory complexity is $\Omega(n \cdot m)$. Assume that we have a mutual information estimation machine that returns an estimate of $I(X;Y)$ with additive error at most ε with probability at least $1-\delta$ using $S^*_{\text{MI}}(n,m,\varepsilon,\delta)$ states. We show below an algorithm that uses this machine as a black box and estimates $H(X,Y) = H(X) + H(Y) - I(X;Y)$ with additive error of at most 3ε with probability at least $1-3\delta$ using $S^*_{\text{MI}} \cdot O(\log^3 n \cdot \log^3 m)$ states. Since estimation of $H(X,Y)$ requires $S^*(n \cdot m, 3\varepsilon, 3\delta) = \Omega(n \cdot m)$, this must imply that

$$S^*_{\text{MI}}(n,m,\varepsilon,\delta) > \Omega\left(\frac{n \cdot m}{\log^3 n \cdot \log^3 m}\right). \tag{81}$$

We now describe such an algorithm. The algorithm has 3 modes. It starts in mode 1, in which $H(X)$ is estimated. It then moves to mode 2, in which $H(Y)$ is estimated, and finally it moves to mode 3 in which $I(X;Y)$ is estimated. The estimation of these 3 quantities is done using

$$\tilde{S} = \max\{S^*(n,\varepsilon,\delta), S^*(m,\varepsilon,\delta), S^*_{\text{MI}}(n,m,\varepsilon,\delta)\} \tag{82}$$

states. Those states are "reused" once the algorithm switches its mode of operation. The current mode is stored using $S_1 = 3$ states.

In mode 1, the algorithm estimates $H(X)$ using the Morris-counter based machine we introduced in Section IV, which uses $S^*(n,\varepsilon,\delta)$ states. After a long enough time, the estimate will be accurate enough, and we will then stop the machine and store that value. In order to decide if enough time has passed, we must ensure that the bias estimation machine with $S_{\text{Bias}} = O(\log^2 n)$ number of states, to which we feed $\text{Ber}(\theta_{N_X})$ samples, is sufficiently mixed. From Theorem 5, we have that the mixing time of the bias estimation machine is at most $4S_{\text{Bias}} \log S_{\text{Bias}} \leq O(\log^3 n)$. Thus, we will let the machine run for $\log^k n$ samples of independent $\text{Ber}(\theta_{N_X})$ random variables for $k >> 3$, and then stop it, which would guarantee it is sufficiently mixed. To determine when to stop the machine without using a memory consuming clock, we will use another Morris counter with $S_2 = O(\log \log^k n) = O(\log \log n)$ states. We will also store the state of the bias estimation machine, which corresponds to our estimate $\hat{H}(X)$ of $H(X)$, using $S_3 = S_{\text{Bias}} = O(\log^2 n)$ states. At this point, the algorithm switches to mode 2, and estimates $H(Y)$ with the algorithm from Section IV, which uses $S^*(m,\varepsilon,\delta)$ states. As in mode 1, we use a Morris counter of $S_4 = O(\log \log m)$ states to determine when the machine is sufficiently mixed and can be stopped, and store the state of the bias estimation machine, which corresponds to the estimate $\hat{H}(Y)$ of $H(Y)$, using $S_5 = O(\log^2 m)$ states. The process then moves to state 3 where $I(X;Y)$ is estimated using the black-box machine with $S^*_{\text{MI}}(n,m,\varepsilon,\delta)$ states. From this time onward, the machine estimates $H(X,Y)$ as $\hat{H}(X) + \hat{H}(Y) - \hat{I}(X;Y)$, where $\hat{I}(X;Y)$ is the current estimate of the black box machine. All in all, this algorithm produces a $(3\varepsilon, 3\delta)$ (recall that we assumed $\delta, \varepsilon \geq 1/100$) estimate of $H(X,Y)$ using

$$S \leq \tilde{S} \prod_{i=1}^{5} S_i = \tilde{S} \cdot O(\log^3 n \cdot \log^3 m), \tag{83}$$

which implies that

$$\tilde{S} = \Omega\left(\frac{S}{\log^3 n \log^3 m}\right) = \Omega\left(\frac{S^*(n,m,3\varepsilon,3\delta)}{\log^3 n \log^3 m}\right) = \Omega\left(\frac{n \cdot m}{\log^3 n \log^3 m}\right). \tag{84}$$

Finally, since Theorem 1 states that $S^*(n,\varepsilon,\delta) = O(n \cdot \log^4 n)$ and $S^*(m,\varepsilon,\delta) = O(m \cdot \log^4 m)$, and we assumed

that $\frac{n}{\log^3 n} = \Omega(\log^7 m)$ and $\frac{m}{\log^3 m} = \Omega(\log^7 n)$, we must therefore have that

$$S^*_{\mathrm{MI}}(n, m, \varepsilon, \delta) = \Omega\left(\frac{n \cdot m}{\log^3 n \cdot \log^3 m}\right). \tag{85}$$

ACKNOWLEDGEMENTS

This work was supported by the ISF under Grants 1641/21 and 1766/22.

REFERENCES

[1] J. Acharya, S. Bhadane, P. Indyk, and Z. Sun, "Estimating entropy of distributions in constant space," *arXiv preprint arXiv:1911.07976*, 2019.

[2] M. Aliakbarpour, A. McGregor, J. Nelson, and E. Waingarten, "Estimation of entropy in constant space with improved sample complexity," *arXiv preprint arXiv:2205.09804*, 2022.

[3] R. Morris, "Counting large numbers of events in small registers," *Communications of the ACM*, vol. 21, no. 10, pp. 840–842, 1978.

[4] F. Leighton and R. Rivest, "Estimating a probability using finite memory," *IEEE Transactions on Information Theory*, vol. 32, no. 6, pp. 733–742, 1986.

[5] T. Berg, O. Ordentlich, and O. Shayevitz, "Deterministic finite-memory bias estimation," in *Conference on Learning Theory*, pp. 566–585, PMLR, 2021.

[6] A. Chakrabarti, K. Do Ba, and S. Muthukrishnan, "Estimating entropy and entropy norm on data streams," *Internet Mathematics*, vol. 3, no. 1, pp. 63–78, 2006.

[7] T. Berg, O. Ordentlich, and O. Shayevitz, "On the memory complexity of uniformity testing," in *Conference on Learning Theory*, pp. 3506–3523, PMLR, 2022.

[8] T. M. Cover *et al.*, "Hypothesis testing with finite statistics," *The Annals of Mathematical Statistics*, vol. 40, no. 3, pp. 828–835, 1969.

[9] M. E. Hellman and T. M. Cover, "Learning with finite memory," *The Annals of Mathematical Statistics*, pp. 765–782, 1970.

[10] F. J. Samaniego, "Estimating a binomial parameter with finite memory," *IEEE Transactions on Information Theory*, vol. 19, no. 5, pp. 636–643, 1973.

[11] J. Steinhardt and J. Duchi, "Minimax rates for memory-bounded sparse linear regression," in *Conference on Learning Theory*, pp. 1564–1587, 2015.

[12] J. Steinhardt, G. Valiant, and S. Wager, "Memory, communication, and statistical queries," in *Conference on Learning Theory*, pp. 1490–1516, 2016.

[13] R. Raz, "Fast learning requires good memory: A time-space lower bound for parity learning," *Journal of the ACM (JACM)*, vol. 66, no. 1, p. 3, 2018.

[14] Y. Dagan and O. Shamir, "Detecting correlations with little memory and communication," in *Conference On Learning Theory*, pp. 1145–1198, 2018.

[15] A. Jain and H. Tyagi, "Effective memory shrinkage in estimation," in *2018 IEEE International Symposium on Information Theory (ISIT)*, pp. 1071–1075, IEEE, 2018.

[16] Y. Dagan, G. Kur, and O. Shamir, "Space lower bounds for linear prediction in the streaming model," in *Conference on Learning Theory*, pp. 929–954, 2019.

[17] V. Sharan, A. Sidford, and G. Valiant, "Memory-sample tradeoffs for linear regression with small error," in *Symposium on Theory of Computing (STOC)*, 2019.

[18] S. Garg, P. K. Kothari, P. Liu, and R. Raz, "Memory-sample lower bounds for learning parity with noise," in *24th International Conference on Approximation Algorithms for Combinatorial Optimization Problems, APPROX 2021 and 25th International Conference on Randomization and Computation, RANDOM 2021*, p. 60, Schloss Dagstuhl-Leibniz-Zentrum fur Informatik GmbH, Dagstuhl Publishing, 2021.

[19] A. Pensia, V. Jog, and P.-L. Loh, "Communication-constrained hypothesis testing: Optimality, robustness, and reverse data processing inequalities," *arXiv preprint arXiv:2206.02765*, 2022.

[20] G. P. Basharin, "On a statistical estimate for the entropy of a sequence of independent random variables," *Theory of Probability & Its Applications*, vol. 4, no. 3, pp. 333–336, 1959.

[21] A. Antos and I. Kontoyiannis, "Convergence properties of functional estimates for discrete distributions," *Random Structures & Algorithms*, vol. 19, no. 3-4, pp. 163–193, 2001.

[22] L. Paninski, "Estimation of entropy and mutual information," *Neural computation*, vol. 15, no. 6, pp. 1191–1253, 2003.

[23] L. Paninski, "Estimating entropy on m bins given fewer than m samples," *IEEE Transactions on Information Theory*, vol. 50, no. 9, pp. 2200–2203, 2004.

[24] G. Valiant and P. Valiant, "A clt and tight lower bounds for estimating entropy.," in *Electron. Colloquium Comput. Complex.*, vol. 17, p. 179, 2010.

[25] G. Valiant and P. Valiant, "Estimating the unseen: an n/log (n)-sample estimator for entropy and support size, shown optimal via new clts," in *Proceedings of the forty-third annual ACM symposium on Theory of computing*, pp. 685–694, 2011.

[26] P. Valiant and G. Valiant, "Estimating the unseen: Improved estimators for entropy and other properties.," in *NIPS*, pp. 2157–2165, 2013.
[27] G. Valiant and P. Valiant, "The power of linear estimators," in *2011 IEEE 52nd Annual Symposium on Foundations of Computer Science*, pp. 403–412, IEEE, 2011.
[28] J. Jiao, K. Venkat, Y. Han, and T. Weissman, "Minimax estimation of functionals of discrete distributions," *IEEE Transactions on Information Theory*, vol. 61, no. 5, pp. 2835–2885, 2015.
[29] Y. Wu and P. Yang, "Minimax rates of entropy estimation on large alphabets via best polynomial approximation," *IEEE Transactions on Information Theory*, vol. 62, no. 6, pp. 3702–3720, 2016.
[30] A. Lall, V. Sekar, M. Ogihara, J. Xu, and H. Zhang, "Data streaming algorithms for estimating entropy of network traffic," *ACM SIGMETRICS Performance Evaluation Review*, vol. 34, no. 1, pp. 145–156, 2006.
[31] S. Guha, A. McGregor, and S. Venkatasubramanian, "Sublinear estimation of entropy and information distances," *ACM Transactions on Algorithms (TALG)*, vol. 5, no. 4, pp. 1–16, 2009.
[32] S. Chien, K. Ligett, and A. McGregor, "Space-efficient estimation of robust statistics and distribution testing.," in *ICS*, pp. 251–265, Citeseer, 2010.
[33] P. Flajolet, "Approximate counting: a detailed analysis," *BIT Numerical Mathematics*, vol. 25, no. 1, pp. 113–134, 1985.
[34] J. Nelson and H. Yu, "Optimal bounds for approximate counting," *arXiv preprint arXiv:2010.02116*, 2020.
[35] D. A. Levin and Y. Peres, *Markov chains and mixing times*, vol. 107. American Mathematical Soc., 2017.
[36] T. M. Cover and J. A. Thomas, *Elements of information theory*. John Wiley & Sons, 2012.