Sketching and Streaming Entropy via Approximation Theory

Nicholas J. A. Harvey* Jelani Nelson† Krzysztof Onak‡

Abstract

We conclude a sequence of work by giving near-optimal sketching and streaming algorithms for estimating Shannon entropy in the most general streaming model, with arbitrary insertions and deletions. This improves on prior results that obtain suboptimal space bounds in the general model, and near-optimal bounds in the insertion-only model without sketching. Our high-level approach is simple: we give algorithms to estimate Rényi and Tsallis entropy, and use them to extrapolate an estimate of Shannon entropy. The accuracy of our estimates is proven using approximation theory arguments and extremal properties of Chebyshev polynomials, a technique which may be useful for other problems. Our work also yields the best-known and near-optimal additive approximations for entropy, and hence also for conditional entropy and mutual information.

*MIT Computer Science and Artificial Intelligence Laboratory. nickh@mit.edu. Supported by a Natural Sciences and Engineering Research Council of Canada PGS Scholarship, by NSF contract CCF-0515221 and by ONR grant N00014-05-1-0148.

†MIT Computer Science and Artificial Intelligence Laboratory. minilek@mit.edu. Supported by a National Defense Science and Engineering Graduate (NDSEG) Fellowship.

‡MIT Computer Science and Artificial Intelligence Laboratory. konak@mit.edu. Supported in part by NSF contract 0514771.

1 Introduction

Streaming algorithms have attracted much attention in several computer science communities, notably theory, databases, and networking. Many algorithmic problems in this model are now well-understood, for example, the problem of estimating frequency moments [1, 2, 10, 17, 31, 34]. More recently, several researchers have studied the problem of estimating the empirical entropy of a stream [3, 6, 7, 12, 13, 36].

Motivation. There are two key motivations for studying entropy. The first is that it is a fundamentally important quantity with useful algebraic properties (chain rule, etc.). The second stems from several practical applications in computer networking, such as network anomaly detection. Let us consider a concrete example. One form of malicious activity on the internet is *port scanning*, in which attackers probe target machines, trying to find open ports which could be leveraged for further attacks. In contrast, typical internet traffic is directed to a small number of heavily used ports for web traffic, email delivery, etc. Consequently, when a port scanning attack is underway, there is a significant change in the distribution of port numbers in the packets being delivered. It has been shown that measuring the entropy of the distribution of port numbers provides an effective means to detect such attacks. See Lakhina et al. [18] and Xu et al. [35] for further information about such problems and methods for their solution.

Our Techniques. In this paper, we give an algorithm for estimating empirical Shannon entropy while using a nearly optimal amount of space. Our algorithm is actually a sketching algorithm, not just a streaming algorithm, and it applies to general streams which allow insertions and deletions of elements. One attractive aspect of our work is its clean high-level approach: we reduce the entropy estimation problem to the well-studied frequency moment problem. More concretely, we give algorithms for estimating other notions of entropy, Rényi and Tsallis entropy, which are closely related to frequency moments. The link to Shannon entropy is established by proving bounds on the rate at which these other entropies converge toward Shannon entropy. Remarkably, it seems that such an analysis was not previously known.

There are several technical obstacles that arise with this approach. Unfortunately, it does not seem that the optimal amount of space can be obtained while using just a single estimate of Rényi or Tsallis entropy. We overcome this obstacle by using several estimates, together with approximation theory arguments and certain infrequently-used extremal properties of Chebyshev polynomials. To our knowledge, this is the first use of such techniques in the context of streaming algorithms, and it seems likely that these techniques could be applicable to many other problems.

Such arguments yield good algorithms for additively estimating entropy, but obtaining a good multiplicative approximation is more difficult when the entropy is very small. In such a scenario, there is necessarily a very heavy element, and the task that one must solve is to estimate the moment of all elements *excluding* this heavy element. This task has become known as the *residual moment* estimation problem, and it is emerging as a useful building block for other streaming problems [3, 5, 10]. To estimate the α^{th} residual moment for $\alpha \in (0, 2]$, we show that $\tilde{O}(\varepsilon^{-2} \log m)$ bits of space suffice with a random oracle and $\tilde{O}(\varepsilon^{-2} \log^2 m)$ bits without. This may be compared with existing algorithms that use $O(\varepsilon^{-2} \log^2 m)$ bits for $\alpha = 2$ [11], and $O(\varepsilon^{-2} \log m)$ for $\alpha = 1$ [10]. No non-trivial algorithms were previously known for $\alpha \notin \{1, 2\}$.

Multiplicative Entropy Estimation. Let us now state the performance of these algorithms more explicitly. We focus exclusively on single-pass algorithms unless otherwise noted. The first algorithms for entropy in the streaming model are due to Guha et al [13]; they achieved $O(\varepsilon^{-2} + \log n)$ words of space but assumed a randomly ordered stream. Chakrabarti, Do Ba and Muthukrishnan [7] then gave an algorithm for worst-case ordered streams using $O(\varepsilon^{-2} \log^2 m)$ words of space, but required two passes over the input. The algorithm of Chakrabarti, Cormode and McGregor [6] uses $O(\varepsilon^{-2} \log m)$ words of space to give a multiplicative $1 + \varepsilon$ approximation, although their algorithm cannot produce sketches and only applies to insertion-only streams. In contrast, the algorithm of Bhuvanagiri and Ganguly [3] can handle deletions but requires roughly $\tilde{O}(\varepsilon^{-3} \log^4 m)$ words[*].

Our work focuses primarily in the *strict turnstile model* (defined in Section 2), which allows deletions. Our algorithm for multiplicatively estimating Shannon entropy uses $\tilde{O}(\varepsilon^{-2} \log m)$ words of space These bounds are nearly-optimal in terms of the dependence on ε, since there is an $\tilde{\Omega}(\varepsilon^{-2})$ lower bound even for insertion-only streams. Our algorithms assume access to a random oracle. This assumption can be removed through the use of Nisan's pseudorandom generator [22], increasing the space bounds by a factor of $O(\log m)$.

Additive Entropy Estimation. Additive approximations of entropy are also useful, as they directly yield additive approximations of conditional entropy and mutual information, which cannot be approximated multiplicatively in small space [16]. Chakrabarti et al. noted that since Shannon entropy is bounded above by $\log m$, a multiplicative $(1 + (\varepsilon/\log m))$ approximation yields an additive ε-approximation. In this way, the work of Chakrabarti et al. [6] and Bhuvanagiri and Ganguly [3] yield additive ε approximations using $O(\varepsilon^{-2} \log^3 m)$ and $\tilde{O}(\varepsilon^{-3} \log^7 m)$ words of space respectively. Our algorithm yields an additive ε approximation using only $\tilde{O}(\varepsilon^{-2} \log m)$ words of space. In particular, our space bounds for multiplicative and additive approximation differ by only $\log \log m$ factors. Zhao et al. [36] give practical methods for additively estimating the so-called entropy norm of a stream. Their algorithm can be viewed as a special case of ours since it interpolates Shannon entropy using two estimates of Tsallis entropy, although this interpretation was seemingly unknown to those authors.

Other Information Statistics. We also give algorithms for approximating Rényi [25] and Tsallis [32] entropy. Rényi entropy plays an important role in expanders [14], pseudorandom generators, quantum computation [33, 37], and ecology [21, 26]. Tsallis entropy is a important quantity in physics that generalizes Boltzmann-Gibbs entropy, and also plays a role in the quantum context. Rényi and Tsallis entropy are both parameterized by a scalar $\alpha \geq 0$. The efficiency of our estimation algorithms depends on α, and is stated precisely in Section 5.

2 Preliminaries

Let $A = (A_1, \ldots, A_n) \in \mathbb{Z}^n$ be a vector initialized as $\vec{0}$ which is modified by a stream of m updates. Each update is of the form (i, v), where $i \in [n]$ and $v \in \{-M, \ldots, M\}$, and causes the change $A_i \leftarrow A_i + v$. For simplicity in stating bounds, we henceforth assume $m \geq n$ and $M = 1$; the latter can be simulated by increasing m by a factor of M and representing an update (i, v) with $|v|$ separate updates (though in actuality our algorithm can perform all $|v|$ updates simultaneously in the time it takes to do one update). The vector A gives rise to a probablity distribution $x = (x_1, \ldots, x_n)$ with $x_i = |A_i| / \|A\|_1$. Thus for each i either $x_i = 0$ or $x_i \geq 1/m$.

[*] A recent, yet unpublished improvement by the same authors [4] improves this to $\tilde{O}(\varepsilon^{-3} \log^3 m)$ words.

In the *strict turnstile model*, we assume $A_i \geq 0$ for all $i \in [n]$ at the end of the stream. In the *general update model* we make no such assumption. For the remainder of this paper, we assume the strict turnstile model and assume access to a random oracle, unless stated otherwise. Our algorithms also extend to the general update model, typically increasing bounds by a factor of $O(\log m)$. As remarked above, the random oracle can be removed, using [22], while increasing the space by another $O(\log m)$ factor.

We now define some functions commonly used in future sections. The α^{th} norm of a vector is denoted $\|\cdot\|_\alpha$. We define the α^{th} *moment* as $F_\alpha = \sum_{i=1}^n |A_i|^\alpha = \|A\|_\alpha^\alpha$. We define the α^{th} *Rényi entropy* as $H_\alpha = \log(\|x\|_\alpha^\alpha)/(1-\alpha)$ and the α^{th} Tsallis entropy as $T_\alpha = (1 - \|x\|_\alpha^\alpha)/(\alpha - 1)$. Shannon entropy $H = H_1$ is defined by $H = -\sum_{i=1}^n x_i \log x_i$. A straightforward application of l'Hôpital's rule shows that $H = \lim_{\alpha \to 1} H_\alpha = \lim_{\alpha \to 1} T_\alpha$. It will often be convenient to focus on the quantity $\alpha - 1$ instead of α itself. Thus, we often write $H(a) = H_{1+a}$ and $T(a) = T_{1+a}$.

We will often need to approximate frequency moments, for which we use the following:

Fact 2.1 (Li [19], [20]). There is an algorithm for multiplicative approximation of F_α for any $\alpha \in (0, 2]$. The algorithm needs $O(\varepsilon^{-2} \log m)$ bits of space in the general update model, and $O\left(\left(\frac{|\alpha-1|}{\varepsilon^2} + \frac{1}{\varepsilon}\right) \log m\right)$ bits of space in the strict turnstile model.

For any function $a \mapsto f(a)$, we denote its k^{th} derivative with respect to a by $f^{(k)}(a)$.

3 Estimating Shannon Entropy

3.1 Overview

We begin by describing a general algorithm for computing an additive approximation to Shannon entropy. The remainder of this paper describes and analyzes various details and incarnations of this algorithm, including extensions to give a multiplicative approximation in Section 3.4. We assume that m, the length of the stream, is known in advance. Computing $\|A\|_1$ is trivial since we assume the strict turnstile model at present.

Algorithm 1. Our algorithm for additively approximating empirical Shannon entropy.

```
Choose error parameter $\tilde{\varepsilon}$ and $k$ points $\{y_0, \ldots, y_k\}$
Process the entire stream:
  For each $i$, compute $\tilde{F}_{1+y_i}$, a $(1+\tilde{\varepsilon})$-approximation of the frequency moment $F_{1+y_i}$
For each $i$, compute $\tilde{H}(y_i) = -\log(\tilde{F}_{1+y_i}/||A||_1^{1+y_i})/y_i$ and $\tilde{T}(y_i) = \big(1 - \tilde{F}_{1+y_i}/||A||_1^{1+y_i}\big)/y_i$
Return an estimate of $H(0)$ or $T(0)$ by interpolation using the points $\tilde{H}(y_i)$ or $\tilde{T}(y_i)$
```

3.2 One-point Interpolation

The easiest implementation of this algorithm is to set $k = 0$, and estimate Shannon entropy H using a single estimate of Rényi entropy $H(y_0)$. We choose the parameters $y_0 = \tilde{\Theta}(\varepsilon/(\log n \log m))$ and $\tilde{\varepsilon} = \varepsilon \cdot y_0$. By Fact 2.1, the space required is $\tilde{O}(\varepsilon^{-3} \log n \log m)$ words. The following argument shows that this gives an additive $O(\varepsilon)$ approximation. With constant probability, $\tilde{F}_{1+y_0} = (1 \pm \tilde{\varepsilon}) F_{1+y_0}$. Then

$$\tilde{H}(y_0) = \frac{-1}{y_0} \log\left(\frac{\tilde{F}_{1+y_0}}{||A||_1^{1+y_0}}\right) = \frac{-1}{y_0} \log\left((1 \pm O(\tilde{\varepsilon})) \sum_{i=1}^n x_i^{1+y_0}\right) = H(y_0) \pm O\Big(\frac{\tilde{\varepsilon}}{y_0}\Big) = H \pm O(\varepsilon). \tag{3.1}$$

The last equality follows from the following theorem, which bounds the rate of convergence of Rényi entropy towards Shannon entropy. A proof is given in Appendix A.1.

Theorem 3.1. Let $x \in \mathbb{R}^n$ be a probability distribution whose smallest positive value is at least $1/m$, where $m \geq n$. Let $0 < \varepsilon < 1$ be arbitrary. Define $\mu = \varepsilon/(4 \log m)$, $\nu = \varepsilon/(4 \log n \log m)$, $\alpha = 1 + \mu/\big(16 \log(1/\mu)\big)$, and $\beta = 1 + \nu/\big(16 \log(1/\nu)\big)$. Then

$$1 \leq \frac{H_1}{H_\alpha} \leq 1 + \varepsilon \qquad \text{and} \qquad 0 \leq H_1 - H_\beta \leq \varepsilon.$$

3.3 Multi-point Interpolation

The algorithm of Section 3.2 is limited by the following tradeoff: if we choose the point y_0 to be close to 0, the accuracy increases, but the space usage also increases. In this section, we avoid that tradeoff by interpolating with multiple points. This allows us to obtain good accuracy without taking the points too close to 0. We formalize this using approximation theory arguments and properties of Chebyshev polynomials.

The algorithm estimates the Tsallis entropy with error parameter $\tilde{\varepsilon} = \varepsilon/(12(k+1)^3 \log m)$ using points $y_0, y_1, \ldots, y_k$, chosen as follows. First, the number of points is $k = \log(1/\varepsilon) + \log\log m$. Their values are chosen to be an affine transformation of the extrema of the k^{th} Chebyshev polynomial. Formally, set $\ell = 1/(2(k+1) \log m)$ and define the affine map $f : \mathbb{R} \to \mathbb{R}$ by

$$f(y) = \frac{(k^2 \ell) \cdot y - \ell \cdot (k^2 + 1)}{2k^2 + 1}, \qquad \text{then define} \qquad y_i = f\big(\cos(i\pi/k)\big). \tag{3.2}$$

The correctness of this algorithm is proven in Section 3.3.2. Let us now analyze the space requirements. Computing the estimate $\tilde{F}_{1+y_i}$ uses only $\tilde{O}(\tilde{\varepsilon}^{-2}/\log m)$ words of space by Fact 2.1 since $|y_i| \leq 1/(2(k+1) \log m)$ for each i. By our choice of $k = \tilde{O}(1)$ and $\tilde{\varepsilon}$, the total space required is $\tilde{O}(\varepsilon^{-2} \log m)$ words.

We argue correctness of this algorithm in Section 3.3.2. Before doing so, we must mention some properties of Chebyshev polynomials.

3.3.1 Chebyshev Polynomials

Our algorithm exploits certain extremal properties of Chebyshev polynomials. For a basic introduction to Chebyshev polynomials we refer the reader to [23, 24, 27]. A thorough treatment of these objects can be found in [28]. We now present the background relevant for our purposes.

Definition 3.2. The set $\mathcal{P}_k$ consists of all polynomials of degree at most k with real coefficients. The Chebyshev polynomial of degree k, $P_k(x)$, is defined by the recurrence

$$P_k(x) = \begin{cases} 1, & k = 0 \\ x, & k = 1 \\ 2xP_{k-1}(x) - P_{k-2}(x), & k \geq 2 \end{cases}$$

and satisfies $|P_k(x)| \leq 1$ for all $x \in [-1, 1]$. The value $|P_k(x)|$ equals 1 for exactly $k+1$ values of x in $[-1, 1]$; specifically, $P_k(\eta_{j,k}) = (-1)^j$ for $0 \leq j \leq k$, where $\eta_{j,k} = \cos(j\pi/k)$. The set $\mathcal{C}_k$ is defined as the set of all polynomials $p \in \mathcal{P}_k$ satisfying $\max_{0 \leq j \leq k} |p(\eta_{j,k})| \leq 1$.

Fact 3.3 (Extremal Growth Property). If $p \in \mathcal{C}_k$ and $|t| \geq 1$, then $|p(t)| \leq |P_k(t)|$.

Proof. See [28, Ex. 1.5.11] or Rogosinski [29]. ■

Fact 3.3 states that all polynomials which are bounded on certain "critical points" of the interval $I = [-1, 1]$ cannot grow faster than Chebyshev polynomials once leaving I.

3.3.2 Correctness

To analyze our algorithm, let us first suppose that our algorithm could exactly compute the Tsallis entropies $T(y_i)$ for $0 \leq i \leq k$. Let p be the degree-k polynomial obtained by interpolating at the chosen points, i.e., $p(y_i) = T(y_i)$ for $0 \leq i \leq k$. The algorithm uses $p(0)$ as its estimate for $T(0)$. We analyze the accuracy of this estimate using the following fact. Recall that the notation $g^{(k)}$ denotes the k^{th} derivative of a function g.

Fact 3.4 (Phillips and Taylor [24], Theorem 4.2). Let $y_0, y_1, \ldots, y_k$ be points in the interval $[a, b]$. Let $g : \mathbb{R} \to \mathbb{R}$ be such that $g^{(1)}, \ldots, g^{(k)}$ exist and are continuous on $[a, b]$, and $g^{(k+1)}$ exists on (a, b). Then, for every $y \in [a, b]$, there exists $\xi_y \in (a, b)$ such that

$$g(y) - p(y) \;=\; \left(\prod_{i=0}^{k} (y - y_i) \right) \frac{g^{(k+1)}(\xi_y)}{(k+1)!}$$

where $p(y)$ is the degree-k polynomial obtained by interpolating the points $(y_i, g(y_i))$, $0 \leq i \leq k$.

To apply this fact, a bound on $|T^{(k+1)}(y)|$ is needed. It suffices to consider the interval $[-\ell, 0)$, since the map f defined in Eq. (3.2) sends $-1 \mapsto -\ell$ and $1 \mapsto -\ell/(2k^2 + 1)$, and hence Eq. (3.2) shows that $y_i \in [-\ell, 0)$ for all i. Since $\ell = 1/(2(k+1)\log m)$, it follows from the following lemma that

$$|T^{(k+1)}(y_i)| \;\leq\; \frac{4\,\log^{k+1}(m)\,H}{k+2} \qquad \forall\, 0 \leq i \leq k. \tag{3.3}$$

Lemma 3.5. Let ε be in $(0, 1/2]$. Then, $|T^{(k)}(-\frac{\varepsilon}{(k+1)\log m})| \;\leq\; 4\,\log^k(m)\,H/(k+1)$.

By Fact 3.4 and Eq. (3.3), we have

$$\begin{aligned} |T(0) - p(0)| \;&\leq\; |\ell|^{k+1} \cdot \frac{4\,\log^{k+1}(m)\,H}{(k+1)!\,(k+2)} \\ &=\; \frac{1}{2^{k+1}\log^{k+1}(m)} \cdot \frac{4\,\log^{k+1}(m)\,H}{(k+1)!\,(k+2)} \\ &\leq\; \frac{2\varepsilon}{(k+1)!\,(k+2)} \;\leq\; \frac{\varepsilon}{2}, \end{aligned} \tag{3.4}$$

since $2^k = (\log m)/\varepsilon$ and $H \leq \log m$. This demonstrates that our algorithm computes a good approximation of $T(0) = H$, under the assumption that the values $T(y_i)$ can be computed exactly. The remainder of this section explains how to remove this assumption.

Algorithm 1 does not compute the exact values $T(y_i)$, it only computes approximations. The accuracy of these approximations can be determined as follows. Then

$$\tilde{T}(y_i) \;=\; \frac{1 - \tilde{F}_{1+y_i}/||A||_1^{1+y_i}}{y_i} \;\leq\; T(y_i) \;-\; \tilde{\varepsilon} \cdot \frac{\sum_{j=1}^{n} x_j^{1+y_i}}{y_i}. \tag{3.5}$$

Now recall that $x_j \geq 1/m$ for each i and $y_i \geq -\ell$, so that $x_i^{y_i} \leq m^{\ell} = m^{1/2(k+1)\log m} < 2$. Thus $\sum_{j=1}^{n} x_j^{1+y_i} \leq 2\sum_{j=1}^{n} x_j = 2$. Since $\tilde{\varepsilon}/\ell = \varepsilon/(6k^2)$, we have

$$T(y_i) \;\leq\; \tilde{T}(y_i) \;\leq\; T(y_i) + \varepsilon/(3k^2). \tag{3.6}$$

Now let $\tilde{p}(x)$ be the degree-k polynomial defined by $\tilde{p}(y_i) = \tilde{T}(y_i)$ for all $0 \leq i \leq k$. Then Eq. (3.6) shows that $r(x) = p(x) - \tilde{p}(x)$ is a polynomial of degree at most k satisfying $|r(y_i)| \leq \varepsilon/(3k^2)$ for all $0 \leq i \leq k$.

Let $P : \mathbb{R} \to \mathbb{R}$ be the Chebyshev polynomial of degree k, and let $Q(y) = P\big(f^{-1}(y)\big)$ be an affine transformation of P. Then the polynomial $r'(y) = (3k^2/\varepsilon) \cdot r(y)$ satisfies $|r'(y_i)| \leq |Q(y_i)|$ for all $0 \leq i \leq k$. Thus Fact 3.3 implies that $|r'(0)| \leq |Q(0)|$. By definition of Q, $Q(0) = P(f^{-1}(0)) = P(1 + 1/k^2)$. The following lemma shows that this is at most e^2.

Lemma 3.6. Let P be the k^{th} Chebyshev polynomial, $k \geq 1$, and let $x = 1 + k^{-c}$. Then

$$|P_k(x)| \;\leq\; \prod_{j=1}^{k}\left(1 + \frac{2j}{k^c}\right) \;\leq\; e^{2k^{2-c}}.$$

Thus $|r'(0)| \leq e^2$ and $|r(0)| \leq \varepsilon/2$ since $k \geq 2$. To conclude, we have shown that $|p(0) - \tilde{p}(0)| = |r(0)| \leq \varepsilon/2$. Combining this with Eq. (3.4) via the triangle inequality shows that $|\tilde{p}(0) - H| \leq \varepsilon$, as desired.

3.4 Multiplicative Approximation of Shannon Entropy

We now discuss how to extend the multi-point interpolation algorithm to obtain a multiplicative approximation of Shannon entropy. The main tool that we require is a multiplicative estimate of Tsallis entropy, rather than the additive estimates used above. Section 5 shows that the required multiplicative estimates can be efficiently computed; Section 4 provides tools for doing this.

The modifications to the multi-point interpolation algorithm are as follows. We set the number of interpolation points to be $k = \max\{5, \log(1/\varepsilon)\}$, then argue as in Eq. (3.4) to have $|T(0) - p(0)| \leq \varepsilon H/2$, where p is the interpolated polynomial of degree k. We then use Algorithm 1, but we compute $\tilde{T}(y_i)$ to be a $(1+\tilde{\varepsilon})$-multiplicative estimation of $T(y_i)$ instead of an $\tilde{\varepsilon}$-additive estimation by using Theorem 5.6. By arguing as in Eq. (3.6), we have $T(y_i) \leq \tilde{T}(y_i) \leq T(y_i) + \varepsilon T(y_i)/(3k^2) \leq T(y_i) + 4\varepsilon H/(3k^2)$. The final inequality follows from Lemma 3.5 with $k = 0$. From this point, the argument remains identical as Section 3.3.2 to show that $|p(0) - \tilde{p}(0)| \leq 4\varepsilon e^2 H/(3k^2) < \varepsilon H/2$, yielding $|\tilde{p}(0) - H| \leq \varepsilon H$ by the triangle inequality.

4 Estimating Residual Moments

To multiplicatively approximate Shannon entropy, the algorithm of Section 3.4 requires a multiplicative approximation of Tsallis entropy. Section 5 shows that the required quantities can be computed. The main tool needed is an efficient algorithm for estimating *residual moments.* That is the topic of the present section.

Define the residual α^{th} moment to be $F_\alpha^{\text{res}} := \sum_{i=2}^{n} |A_i|^\alpha = F_\alpha - |A_1|^\alpha$, where we reorder the items such that $|A_1| \geq |A_2| \geq \ldots \geq |A_n|$. In this section, we present two efficient algorithms to compute a $1+\varepsilon$ multiplicative approximation to F_α^{res} for $\alpha \in (0, 2]$. These algorithms succeed with constant probability under the assumption that a heavy hitter exists, say $|A_1| \geq \frac{4}{5}\left\|A\right\|_1$. The algorithm of Section 4.2 is valid only in the strict turnstile model. Its space usage has a complicated dependence on α; for the primary range of interest, $\alpha \in [1/3, 1)$, the bound is $O\left(\left(\frac{1}{\varepsilon^{1/\alpha}} + \frac{1-\alpha}{\varepsilon^2} + \log n\right)\log m\right)$. The algorithm of Section 4.3 is valid in the general update model and uses $\tilde{O}(\varepsilon^{-2}\log m)$ bits of space. In comparison, the result of Ganguly et al. [11], approximates only F_2^{res} using $O(\varepsilon^{-2}\log^2 m)$ bits of space, also in the general update model but without a random oracle. Another difference is that our algorithm requires that a heavy hitter exists, whereas the Ganguly et al. algorithm does not.

4.1 Finding a Heavy Element

A subroutine that is needed for both of our algorithms is to detect whether a heavy hitter exists ($|A_i| \geq \frac{4}{5}\left\|A\right\|_1$) and to find the identity of that element. We will describe a procedure for doing so in the general update model. We use the following result, which essentially follows from the count-min sketch [8]. For completeness, a self-contained proof is given in Appendix A.5.

Fact 4.1. Let $w \in \mathbb{R}_+^n$ be a weight vector on n elements so that $\sum_i w_i = 1$. There exists a family $\mathcal{H}$ of hash functions mapping the n elements to $O(1/\varepsilon)$ bins with $|\mathcal{H}| = n^{O(1)}$ such that a random $h \in \mathcal{H}$ satisfies the following two properties with probability at least 15/16.
(1) If $w_i \geq 1/2$ then the weight of elements that collide with element i is at most $\varepsilon \cdot \sum_{j \neq i} w_j$.
(2) If $\max_i w_i < 1/2$ then the weight of elements hashing to each bin is at most 3/4.

We use the hash function from Fact 4.1 with $\varepsilon = 1/10$ to partition the elements into bins, and for each bin maintain a counter of the net L_1 weight that hash to it. If there is a heavy hitter, then the net weight in its bin is more than $4/5 - \varepsilon(1/5) > 3/4$. Conversely, if there is a bin with at least 3/4 of the weight then Fact 4.1 implies then there is a heavy element.

We determine the identity of the heavy element via a group-testing type of argument: we maintain $\lceil \log_2 n \rceil$ counters, of which the i^{th} counts the number of elements which have their i^{th} bit set. Thus, if there is heavy element, we can determine its i^{th} bit by checking whether the fraction of elements with their i^{th} bit is at least 3/5.

4.2 Bucketing Algorithm

In this section, we describe an algorithm for estimating F_α^{res} that works only in the strict turnstile model. The algorithm has several cases, depending on the value of α. The third case uses an interesting technique of artifically inserting deletions into the stream.

Case 1: $\alpha = 1$. This is the simplest case for our algorithm. We use the hash function from Fact 4.1 to partition the elements into bins, and for each bin maintain a count of the number of elements that hash to it. If there is a bin with more than 3/4 elements at the end of the procedure, then there is a heavy element, and it suffices to return the total number of elements in the other bins. Otherwise, we announce that there is no heavy hitter. The correctness follows from Fact 4.1, and the space required is $O\left(\frac{1}{\varepsilon}\log m\right)$ bits.

Case 2: $\alpha = (0, \frac{1}{3}) \cup (1, 2]$. Again, we use the hash function from Fact 4.1 to partition the elements into bins. For each bin, we maintain a count of the number of elements, and a sketch of the α^{th} moment using Fact 2.1. The counts allow us to detect if there is a heavy hitter, as in Case 1. If so, we combine the moment sketches of all bins other than the one containing the heavy hitter; this gives a good estimate with constant probability. By Fact 2.1, we need only

$$O\left(\frac{1}{\varepsilon} \cdot \left(\frac{|\alpha-1|}{\varepsilon^2} + \frac{1}{\varepsilon}\right)\log m + \frac{1}{\varepsilon}\log m\right) = O\left(\left(\frac{|\alpha-1|}{\varepsilon^3} + \frac{1}{\varepsilon^2}\right)\log m\right) \text{ bits.}$$

Case 3: $\alpha = [\frac{1}{3}, 1)$**.** This is the most interesting case. This idea is to keep just one sketch of the α^{th} moment for the entire stream. At the end, we estimate F_α^{res} by artificially appending deletions to the stream which almost entirely remove the heavy hitter from the sketch.

The algorithm computes four quantities in parallel. First, $\tilde{F}_1^{\text{res}} = (1 \pm \varepsilon')F_1^{\text{res}}$ with error parameter $\varepsilon' = \varepsilon^{1/\alpha}$, using the above algorithm with $\alpha = 1$. Second, $\tilde{F}_\alpha = (1 \pm \varepsilon)F_\alpha$ using Fact 2.1. Third, F_1, which is trivial in the strict turnstile model. Lastly, we determine the identity of the heavy hitter as in Section 4.1.

Now we explain how to estimate F_α^{res}. The key observation is that $F_1 - \tilde{F}_1^{\text{res}}$ is a very good approximation to A_1 (assume this is the heavy hitter). So if we delete the heavy hitter $(F_1 - \tilde{F}_1^{\text{res}})$ times, then there are at most $A_1 \leq \varepsilon' F_1^{\text{res}}$ remaining occurrences. Define $\tilde{F}_\alpha^{\text{res}}$ to be the value of $\tilde{F}_\alpha$ after processing these deletions. Clearly $F_\alpha^{\text{res}} \geq (F_1^{\text{res}})^\alpha$, by concavity of the function $y \mapsto y^\alpha$. On the other hand, the remaining occurrences of the heavy hitter contribute at most $(\varepsilon' F_1^{\text{res}})^\alpha$. Hence, the remaining occurences of the heavy hitter inflate F_α^{res} by a factor of at most $1 + (\varepsilon' \cdot F_1^{\text{res}})^\alpha / (F_1^{\text{res}})^\alpha = 1 + \varepsilon$. Thus $\tilde{F}_\alpha^{\text{res}} = (1 + O(\varepsilon))F_\alpha^{\text{res}}$, as desired. The number of bits of space used by this algorithm is at most

$$O\left(\tfrac{1}{\varepsilon'} \log m + \left(\tfrac{1-\alpha}{\varepsilon^2} + \tfrac{1}{\varepsilon}\right) \log m + \log n \log m\right) \;=\; O\left(\left(\tfrac{1}{\varepsilon^{1/\alpha}} + \tfrac{1-\alpha}{\varepsilon^2} + \log n\right) \log m\right).$$

4.3 Geometric Mean Algorithm

This section describes an algorithm for estimating F_α^{res} in the general update model. At a high level, the algorithm uses a hash function to partition the stream elements into two substreams, then separately estimates the moment F_α for the substreams. The estimate for the substream which does not contain the heavy hitter yields a good estimate of F_α^{res}. We improve accuracy of this estimator by averaging many independent trials. Detailed description and analysis follow.

We use Li's *geometric mean estimator* [20] for estimating F_α since it is unbiased (its being unbiased will be useful later). The geometric mean estimator is defined as follows. Let k and α be parameters. We let $y = R \cdot A$, where A is the vector representing the stream and R is a $k \times n$ matrix whose entries are i.i.d. samples from an α-stable distribution. Define

$$\tilde{F}_\alpha \;=\; \frac{\prod_{j=1}^k |y_j|^{\alpha/k}}{[\frac{2}{\pi}\Gamma(\frac{\alpha}{k})\Gamma(1 - \frac{1}{k})\sin(\frac{\pi\alpha}{2k})]^k}.$$

The space required to compute this estimator is easily seen to be $O(k \cdot \log m)$ bits. Li analyzed the variance of $\tilde{F}_\alpha$ as $k \to \infty$, however for our purposes we are only interested in the case $k = 3$ and henceforth restrict to only this case (one can show $\tilde{F}_\alpha$ has unbounded variance for $k < 3$). Building on Li's analysis, we show the following result.

Lemma 4.2. There exists an absolute constant C_{GM} such that $\text{Var}\left[\tilde{F}_\alpha\right] \leq C_{GM} \cdot \text{E}\left[\tilde{F}_\alpha\right]^2$.

Let r denote the number of independent trials. For each $j \in [r]$, the algorithm picks a function $h_j : [n] \to \{0, 1\}$ uniformly at random. For $j \in [r]$ and $l \in \{0, 1\}$, define $F_{\alpha,j,l} = \sum_{i:h_j(i)=l} |A_i|^\alpha$. This is the α^{th} moment for the l^{th} substream during the j^{th} trial.

For each j and l, our algorithm computes an estimate $\tilde{F}_{\alpha,j,l}$ of $F_{\alpha,j,l}$ using the geometric mean estimator. We also run in parallel the algorithm of Section 4.1 to discover which $i \in [n]$ is the heavy hitter; henceforth assume $i = 1$. Our overall estimate for F_α^{res} is then

$$\tilde{F}_\alpha^{\text{res}} \;=\; \frac{2}{r} \sum_{j=1}^{r} \tilde{F}_{\alpha,j,1-h_j(1)}$$

The space used by our algorithm is simply the space required for r geometric mean estimators and the one heavy hitter algorithm. The latter uses $\tilde{O}(\varepsilon^{-1}\log n)$ bits of space [8, Theorem 7]. Thus the total space required is $\tilde{O}(r\log m+\varepsilon^{-1}\log n)$ bits.

We now sketch an analysis of the algorithm; a formal argument is given in Appendix A.4. The natural analysis would be to show that, for each item, the fraction of trials in which the item doesn't collide with the heavy hitter is concentrated around 1/2. A union bound over all items would require choosing the number of trials to be $\Omega(\frac{1}{\varepsilon^2}\log n)$. We obtain a significantly smaller number of trials by using a different analysis. Instead of using a concentration bound for each item, we observe that items with roughly the same weight (i.e., the value of $|A_i|$) are essentially equivalent for the purposes of this analysis. So we partition the items into classes such that all items in the a class have the same weight, up to a $(1+\varepsilon)$ factor. We then apply concentration bounds for each class, rather than separately for each item. The number of classes is only $R=O(\frac{1}{\varepsilon}\log m)$, and a union bound over classes only requires $\Theta(\frac{1}{\varepsilon^2}\log R)$ trials.

As argued, the space usage of this algorithm is $\tilde{O}(r\log m+\varepsilon^{-1}\log n)=\tilde{O}(\varepsilon^{-2}\log m)$ bits.

5 Estimation of Rényi and Tsallis Entropy

Theorem 5.1. There is an algorithm that computes an additive ε-approximation of Rényi entropy in $O\left(\frac{\log m}{|1-\alpha|\cdot\varepsilon^2}\right)$ bits of space for any $\alpha\in(0,1)\cup(1,2]$.

Theorem 5.2. There is an algorithm for additive approximation of Tsallis entropy T_α using

- $O\left(\frac{n^{2(1-\alpha)}\log m}{(1-\alpha)\varepsilon^2}\right)$ bits, for $\alpha\in(0,1)$.
- $O\left(\frac{\log m}{(\alpha-1)\varepsilon^2}\right)$ bits, for $\alpha\in(1,2]$.

Lemma 5.3. Let $x_1,x_2,\ldots,x_n$ be values in $[0,1]$ of total sum 1. There exists a positive constant C such that if $x_i\leq 5/6$ for all i then, for $\alpha\in(0,1)\cup(1,2]$,

$$\left|1-\sum_{i=1}^{n}x_i^\alpha\right|\geq C\cdot|\alpha-1|.$$

Corollary 5.4. There exists a constant C such that if the probability of each element is at most 5/6, then the Tsallis entropy is at least C for any $\alpha\in(0,1)\cup(1,2]$.

Proof. We have

$$T_\alpha=\frac{1-\sum_{i=1}^{n}x^\alpha}{\alpha-1}=\frac{|1-\sum_{i=1}^{n}x_i^\alpha|}{|\alpha-1|}\geq C.$$

■

Lemma 5.5. There is a positive constant C such that if there is an element i of probability $x_i\geq 2/3$, then the sum of a multiplicative $(1+C\cdot|1-\alpha|\cdot\varepsilon)$-approximation to $1-x_i$ and a multiplicative $(1+C\cdot|1-\alpha|\cdot\varepsilon)$-approximation to $\sum_{j\neq i}x_j^\alpha$ gives a multiplicative $(1+\varepsilon)$-approximation to $|1-\sum_i x_i^\alpha|$, for any $\alpha\in(0,1)\cup(1,2]$.

Theorem 5.6. There is a streaming algorithm for multiplicative $(1+\varepsilon)$-approximation of Tsallis entropy for any $\alpha\in(0,1)\cup(1,2]$ using $\tilde{O}\left(\log m/(|1-\alpha|\varepsilon^2)\right)$ bits of space.

Lemma 5.7. It suffices to have a multiplicative $(1+\varepsilon)$-approximation to $t-1$, where $t\in(4/9,\infty)$ to compute a multiplicative $(1+C\cdot\varepsilon)$ approximation to $\log(t)$, for some constant C.

Theorem 5.8. There is a streaming algorithm for multiplicative $(1+\varepsilon)$-approximation of Rényi entropy for any $\alpha \in (0,1) \cup (1,2]$. The algorithm uses $\tilde{O}\left(\log m/(|1-\alpha|\varepsilon^2)\right)$ bits of space.

In fact, Theorem 5.8 is tight in the sense that $(1+\varepsilon)$-multiplicative approximation of H_α for $\alpha > 2$ requires polynomial space, as seen in the following theorem.

Theorem 5.9. For any $\alpha > 2$, any randomized one-pass streaming algorithm which $(1+\varepsilon)$-approximates $H_\alpha(X)$ requires $\Omega(n^{1-2/\alpha-2\varepsilon-\gamma(\varepsilon+1/\alpha)})$ bits of space for arbitrary constant $\gamma > 0$.

6 Modifications for General Update Streams

The algorithms described in Section 3 and Section 5 are for the strict turnstile model. They can be extended to work in the general updates model with a few modifications.

First, we cannot efficiently and exactly compute $\|A\|_1 = F_1$ in the general update model. However, a $(1+\varepsilon)$-multiplicative approximation can be computed in $O(\varepsilon^{-2}\log m)$ bits of space by Fact 2.1. In Section 3.2 and Section 3.3, the value of $\|A\|_1$ is used as a normalization factor to scale the estimate of F_α to an estimate of $\sum_{i=1}^n x_i^\alpha$. (See, e.g., Eq. (3.1) and Eq. (3.5).) However,

$$\frac{\tilde{F}_\alpha}{(\tilde{F}_1)^\alpha} \;=\; \frac{(1 \pm \varepsilon)\cdot F_\alpha}{\big((1\pm\varepsilon)\cdot F_1\big)^\alpha} \;=\; \big(1 \pm O(\varepsilon)\big)\cdot \frac{F_\alpha}{F_1^\alpha},$$

so the fact that F_1 can only be approximated in the general update model affects the analysis only by increasing the constant factor that multiplies ε. A similar modification must also be applied to all algorithms in Section 5; we omit the details.

Next, the multiplicative algorithm Section 3.4 needs to compute a multiplicative estimate of $T(y_i)$ using Theorem 5.6. In the general updates model, a weaker result than Theorem 5.6 holds: we obtain a multiplicative $(1+\varepsilon)$-approximation of Tsallis entropy for any $\alpha \in (0,1)\cup(1,2]$ using $\tilde{O}\left(\log m/(|1-\alpha|\cdot\varepsilon)^2\right)$ bits of space. The proof is identical to the argument in Appendix A.6, except that the the moment estimator of Fact 2.1 uses more space, and we must use the residual moment algorithm of Section 4.3 instead of Section 4.2. Similar modifications must be made to Theorem 5.1, Theorem 5.2 and Theorem 5.8, with a commensurate increase in the space bounds.

7 Future Research

We hope that the techniques from approximation theory that we introduce may be useful for streaming and sketching other functions. For instance, consider the following function $G_{\alpha,k}(x) = \sum_i x_i^\alpha (\log n)^k$, where $k \in \mathbb{N}$ and $\alpha \in [0,\infty)$. One can show that

$$\lim_{\beta\to\alpha} \frac{G_{\alpha,k}(x) - G_{\beta,k}(x)}{\alpha-\beta} = G_{\beta,k+1}(x).$$

Note that $G_{\alpha,0}(x)$ is the α-th moment of x, and one can attempt to estimate $G_{\alpha,k+1}$ by computing $G_{\beta,k}$ for $\beta = \alpha$ and β close to α. It is not unlikely that our techniques can be generalized to estimation of functions $G_{\alpha,k}$ for $\alpha \in (0,2]$. Can one also use our techniques for approximation of other classes of functions?

Acknowledgements

We thank Piotr Indyk and Ping Li for many helpful discussions. We also thank Jonathan Kelner for some pointers to the approximation theory literature.

References

[1] Noga Alon, Yossi Matias, and Mario Szegedy. The Space Complexity of Approximating the Frequency Moments. *Journal of Computer and System Sciences*, 58(1):137–147, 1999.

[2] Ziv Bar-Yossef, T. S. Jayram, Ravi Kumar, and D. Sivakumar. An information statistics approach to data stream and communication complexity. *J. Comput. Syst. Sci.*, 68(4):702–732, 2004.

[3] Lakshminath Bhuvanagiri and Sumit Ganguly. Estimating entropy over data streams. In *Proceedings of the 14th Annual European Symposium on Algorithms*, pages 148–159, 2006.

[4] Lakshminath Bhuvanagiri and Sumit Ganguly. Hierarchical Sampling from Sketches: Estimating Functions over Data Streams, 2008. Manuscript.

[5] Lakshminath Bhuvanagiri, Sumit Ganguly, Deepanjan Kesh, and Chandan Saha. Simpler algorithm for estimating frequency moments of data streams. In *Proceedings of the 17th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 708–713, 2006.

[6] Amit Chakrabarti, Graham Cormode, and Andrew McGregor. A near-optimal algorithm for computing the entropy of a stream. In *Proceedings of the 18th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 328–335, 2007.

[7] Amit Chakrabarti, Khanh Do Ba, and S. Muthukrishnan. Estimating Entropy and Entropy Norm on Data Streams. In *Proceedings of the 23rd Annual Symposium on Theoretical Aspects of Computer Science (STACS)*, pages 196–205, 2006.

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

[9] Thomas Cover and Joy Thomas. *Elements of Information Theory.* Wiley Interscience, 1991.

[10] Sumit Ganguly and Graham Cormode. On estimating frequency moments of data streams. In *APPROX-RANDOM*, pages 479–493, 2007.

[11] Sumit Ganguly, Deepanjan Kesh, and Chandan Saha. Practical algorithms for tracking database join sizes. In *Proceedings of the 25th International Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS)*, pages 297–309, 2005.

[12] Yu Gu, Andrew McCallum, and Donald F. Towsley. Detecting anomalies in network traffic using maximum entropy estimation. In *Internet Measurment Conference*, pages 345–350, 2005.

[13] Sudipto Guha, Andrew McGregor, and Suresh Venkatasubramanian. Streaming and sublinear approximation of entropy and information distances. In *Proceedings of the 17th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 733–742, 2006.

[14] Shlomo Hoory, Nathan Linial, and Avi Wigderson. Expander graphs and their applications. *Bulletin of the American Mathematical Society*, 43(4):439–561, 2006.

[15] Piotr Indyk. Stable distributions, pseudorandom generators, embeddings, and data stream computation. *Journal of the ACM*, 53(3):307–323, 2006.

[16] Piotr Indyk and Andrew McGregor. Declaring independence via the sketching of sketches. In *Proceedings of the 19th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2008.

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

[18] A. Lakhina, M. Crovella, and C. Diot. Mining anomalies using traffic feature distributions. In *Proceedings of the ACM SIGCOMM Conference*, pages 217–228, 2005.

[19] Ping Li. Compressed counting. CoRR abs/0802.2305v2, 2008.

[20] Ping Li. Estimators and tail bounds for dimension reduction in l_p $(0 < p \leq 2)$ using stable random projections. In *Proceedings of the 19th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 10–19, 2008.

[21] Canran Liu, Robert J. Whittaker, Keping Ma, and Jay R. Malcolm. Unifying and distinguishing diversity ordering methods for comparing communities. *Population Ecology*, 49(2):89–100, 2006.

[22] Noam Nisan. Pseudorandom generators for space-bounded computation. *Combinatorica*, 12(4):449–461, 1992.

[23] George McArtney Phillips. *Interpolation and Approximation by Polynomials.* Springer-Verlag, New York, 2003.

[24] George McArtney Phillips and Peter John Taylor. *Theory and Applications of Numerical Analysis.* Academic Press, 2nd edition, 1996.

[25] Alfred Rényi. On measures of entropy and information. In *Proc. Fourth Berkeley Symp. Math. Stat. and Probability*, volume 1, pages 547–561, 1961.

[26] Carlo Ricotta, Alessandra Pacini, and Giancarlo Avena. Parametric scaling from species to growth-form diversity: an interesting analogy with multifractal functions. *Biosystems*, 65(2–3):179–186, 2002.

[27] Theodore J. Rivlin. *An Introduction to the Approximation of Functions.* Dover Publications, New York, 1981.

[28] Theodore J. Rivlin. *Chebyshev Polynomials: From Approximation Theory to Algebra and Number Theory.* John Wiley & Sons, 2nd edition, 1990.

[29] Werner Wolfgang Rogosinski. Some elementary inequalities for polynomials. *The Mathematical Gazette*, 39(327):7–12, 1955.

[30] Walter Rudin. *Principles of Mathematical Analysis.* McGraw-Hill, third edition, 1976.

[31] Michael E. Saks and Xiaodong Sun. Space lower bounds for distance approximation in the data stream model. In *Proceedings of the 34th Annual ACM Symposium on Theory of Computing (STOC)*, pages 360–369, 2002.

[32] Constantino Tsallis. Possible generalization of boltzmann-gibbs statistics. *Journal of Statistical Physics*, 52:479–487, 1988.

[33] Wim van Dam and Patrick Hayden. Renyi-entropic bounds on quantum communication. arXiv:quant-ph/0204093, 2002.

[34] David Woodruff. *Efficient and Private Distance Approximation in the Communication and Streaming Models*. PhD thesis, Massachusetts Institute of Technology, 2007.

[35] Kuai Xu, Zhi-Li Zhang, and Supratik Bhattacharyya. Profiling internet backbone traffic: behavior models and applications. In *Proceedings of the ACM SIGCOMM Conference*, pages 169–180, 2005.

[36] Haiquan Zhao, Ashwin Lall, Mitsunori Ogihara, Oliver Spatscheck, Jia Wang, and Jun Xu. A Data Streaming Algorithm for Estimating Entropies of OD Flows. In *Proceedings of the Internet Measurement Conference (IMC)*, 2007.

[37] Karol Życzkowski. Rényi Extrapolation of Shannon Entropy. *Open Systems & Information Dynamics*, 10(3):297–310, 2003.

A Proofs

A.1 Proofs from Section 3.2

Recall that $x \in \mathbb{R}^n$ is a distribution whose smallest positive value is at least $1/m$. The key technical lemma needed is as follows.

Lemma A.1. Let $\alpha > 1$, let $\xi = \xi(\alpha)$ denote $4(\alpha - 1)H_1(x)$, and let

$$e(\alpha) \;=\; 2\Big(\xi \log n \;+\; \xi \log(1/\xi)\Big).$$

Assume that $\xi(\alpha) < 1/4$. Then $H_\alpha \leq H_1 \leq H_\alpha + e(\alpha)$.

We require the following basic results.

Claim A.2. The following inequalities follow from convexity.

- Let $0 < y \leq 1$. Then $e^y < 1 + 2y$.
- Let $y > 0$. Then $1 - y \leq \log(1/y)$.
- Let $0 \leq y \leq 1/2$. Then $1/(1-y) \leq 1 + 2y$.

Claim A.3. Let $1 \leq a \leq b$ and let $x \in \mathbb{R}^n$. Then $\|x\|_b \leq \|x\|_a \leq n^{1/a - 1/b} \|x\|_b$.

Claim A.4. If $0 \leq \alpha \leq \beta$ then $H_\alpha \geq H_\beta$

Claim A.5. If $\alpha > 1$ then $\log\left(1/\|x\|_\alpha\right) < (\alpha - 1) \cdot H_1$.

Proof. $\log\left(1/\|x\|_\alpha\right) \;=\; \frac{\alpha-1}{\alpha} H_\alpha(x) \;<\; (\alpha - 1) \cdot H_\alpha(x) \;\leq\; (\alpha - 1) \cdot H_1(x).$ ■

Claim A.6. Let $y = (y_1, \ldots, y_n)$ and $z = (z_1, \ldots, z_n)$ be probability distributions such that $\|y - z\|_1 \leq 1/2$. Then

$$|H_1(y) - H_1(z)| \;\leq\; \|y - z\|_1 \cdot \log\left(\frac{n}{\|y - z\|_1}\right).$$

Proof. See Cover and Thomas [9, 16.3.2]. ■

Proof (of Lemma A.1). The first inequality follows from Claim A.4 so we focus on the second one. Define $f(\alpha) = \log \|x\|_\alpha^\alpha$ and $g(\alpha) = 1 - \alpha$, so that $H_\alpha = f(\alpha)/g(\alpha)$. The derivatives are

$$f'(\alpha) \;=\; \frac{\sum_{i=1}^n x_i^\alpha \log x_i}{\|x\|_\alpha^\alpha} \qquad \text{and} \qquad g'(\alpha) \;=\; -1,$$

so $\lim_{\alpha \to 1} f'(\alpha)/g'(\alpha)$ exists and equals $H(x)$. Since $\lim_{\alpha \to 1} f(\alpha) = \lim_{\alpha \to 1} g(\alpha) = 0$, L'Hôpital's rule implies that $\lim_{\alpha \to 1} H_\alpha = H(x)$. A stronger version of L'Hôpital's rule is as follows.

Claim A.7. Let $f : \mathbb{R} \to \mathbb{R}$ and $g : \mathbb{R} \to \mathbb{R}$ be differentiable functions such that the following limits exist

$$\lim_{\alpha \to 1} f(\alpha) = 0, \qquad \lim_{\alpha \to 1} g(\alpha) = 0, \qquad \text{and} \qquad \lim_{\alpha \to 1} f'(\alpha)/g'(\alpha) = L.$$

Let ε and δ be such that $|\alpha - 1| < \delta$ implies that $|f'(\alpha)/g'(\alpha) - L| < \varepsilon$. Then $|\alpha - 1| < \delta$ also implies that $|f(\alpha)/g(\alpha) - L| < \varepsilon$.

Proof. See Rudin [30, p.109]. □

Thus, to prove our lemma, it suffices to show that $|f'(\alpha)/g'(\alpha) - H_1| < e(\alpha)$. (In fact, we actually need $|f'(\beta)/g'(\beta) - H_1| < e(\alpha)$ for all $\beta \in (1, \alpha]$, but this follows by monotonicity of $e(\beta)$ for $\beta \in (1, \alpha]$.)

A key concept in this proof is the "perturbed" probability distribution $x(\alpha)$, defined by $x(\alpha)_i = x_i^\alpha / \|x\|_\alpha^\alpha$. We have the following relationship.

$$\begin{aligned}
\frac{f'(\alpha)}{g'(\alpha)} &= \frac{\sum_{i=1}^n x_i^\alpha \log(1/x_i)}{\|x\|_\alpha^\alpha} \\
&= \frac{\sum_{i=1}^n x_i^\alpha \big(\log(1/x_i) + \log \|x\|_\alpha - \log \|x\|_\alpha \big)}{\|x\|_\alpha^\alpha} \\
&= \frac{\Big(\sum_{i=1}^n x_i^\alpha \log(\|x\|_\alpha / x_i) \Big) \;-\; \Big(\sum_{i=1}^n x_i^\alpha \log \|x\|_\alpha \Big)}{\|x\|_\alpha^\alpha} \\
&= \frac{1}{\alpha} \sum_{i=1}^n \frac{x_i^\alpha}{\|x\|_\alpha^\alpha} \log \left(\frac{\|x\|_\alpha^\alpha}{x_i^\alpha} \right) \;-\; \log \|x\|_\alpha \\
&= \frac{H_1\big(x(\alpha)\big)}{\alpha} \;+\; \log(1/\|x\|_\alpha)
\end{aligned}$$

In summary, we have shown that

$$\left| \frac{f'(\alpha)}{g'(\alpha)} - \frac{H_1\big(x(\alpha)\big)}{\alpha} \right| \;\leq\; \log(1/\|x\|_\alpha) \;\leq\; (\alpha - 1) \cdot H_1(x), \tag{A.1}$$

the last inequality following from Claim A.5. To use this bound, we observe that:

$$\begin{aligned}
\left| \frac{f'(\alpha)}{g'(\alpha)} - H_1\big(x(\alpha)\big) \right| &= \left| \frac{f'(\alpha)}{g'(\alpha)} - \frac{H_1\big(x(\alpha)\big)}{\alpha} + \left(\frac{1}{\alpha} - 1 \right) H_1\big(x(\alpha)\big) \right| \\
&\leq \left| \frac{f'(\alpha)}{g'(\alpha)} - \frac{H_1\big(x(\alpha)\big)}{\alpha} \right| \;+\; |1/\alpha - 1| \cdot H_1\big(x(\alpha)\big)
\end{aligned}$$

We now substitute Eq. (A.1) into this expression, and use $|1/\alpha - 1| \leq \alpha - 1$ (valid since $\alpha \geq 1$). This yields:

$$\left| \frac{f'(\alpha)}{g'(\alpha)} - H_1\big(x(\alpha)\big) \right| \;\leq\; (\alpha - 1) \cdot H_1(x) \;+\; (\alpha - 1) \cdot H_1\big(x(\alpha)\big) \tag{A.2}$$

Recall that our goal is to analyze $|f'(\alpha)/g'(\alpha) - H_1(x)|$. We do this by showing that $H_1\big(x(\alpha)\big) \approx H_1(x)$, and that the right-hand side of Eq. (A.2) is at most $e(\alpha)$. This is done using Claim A.6; the key step is bounding $\|x - x(\alpha)\|_1$.

Claim A.8. Suppose that $1 < \alpha \leq 1 + 1/(2 \log n)$. Then $1/\left\|x\right\|_\alpha^\alpha < 1 + 3(\alpha - 1)H_1(x)$.

Proof. From Claim A.3 and $\|x\|_1 = 1$, we obtain $1/\left\|x\right\|_\alpha \leq n^{1-1/\alpha} < n^{\alpha-1}$. Our hypothesis on α implies that

$$\alpha \cdot \log(1/\left\|x\right\|_\alpha) \;<\; \alpha \cdot (\alpha - 1)\log n \;<\; 2 \cdot (\alpha - 1)\log n \;\leq\; 1. \tag{A.3}$$

Thus

$$\frac{1}{\left\|x\right\|_\alpha^\alpha} \;=\; e^{\alpha \log(1/\|x\|_\alpha)} \;<\; 1 + 2 \cdot \alpha \log(1/\left\|x\right\|_\alpha) \;<\; 1 + 3(\alpha - 1)H_1(x).$$

The first inequality is from Claim A.2 and Eq. (A.3), and the second from Claim A.5. □

Recall that $\xi = 4(\alpha - 1)H_1(x)$.

Claim A.9. $\|x - x(\alpha)\|_1 \leq \xi$.

Proof. To avoid the absolute values, we shall split the sum defining $\|x - x(\alpha)\|_1$ into two cases. For that purpose, let $S = \{\ i \,:\, x(\alpha)_i \geq x_i\ \}$. Then

$$\begin{aligned} \|x - x(\alpha)\|_1 \;&=\; \sum_{i \in S} \big(x(\alpha)_i - x_i\big) \;+\; \sum_{i \notin S} \big(x_i - x(\alpha)_i\big) \\ &=\; \sum_{i \in S} x_i \cdot \left(\frac{x_i^{\alpha-1}}{\left\|x\right\|_\alpha^\alpha} - 1 \right) \;+\; \sum_{i \notin S} x_i \cdot \left(1 - \frac{x_i^{\alpha-1}}{\left\|x\right\|_\alpha^\alpha} \right) \end{aligned}$$

The first sum is upper-bounded using $x_i^{\alpha-1} \leq 1$ and $\sum_{i \in S} x_i \leq 1$. The second sum is upper-bounded using $\left\|x\right\|_\alpha^\alpha \leq 1$ and $1 - x_i^{\alpha-1} \leq \log\big(1/x_i^{\alpha-1}\big)$ (see Claim A.2).

$$\begin{aligned} &\leq\; \left(\frac{1}{\left\|x\right\|_\alpha^\alpha} - 1 \right) \;+\; (\alpha - 1) \sum_{i \notin S} x_i \log(1/x_i) \\ &\leq\; 3(\alpha - 1)H_1(x) \;+\; (\alpha - 1)H_1(x), \end{aligned}$$

using Claim A.8. This completes the proof. □

Thus, by our assumption that $\xi(\alpha) < 1/4$, by Claim A.6, by Claim A.9, and by the fact that $x \mapsto x \log(1/x)$ is monotonically increasing for $x \in (0, 1/4)$, we obtain that

$$|H_1(x) - H_1(x(\alpha))| \;\leq\; \xi \log n + \xi \log(1/\xi).$$

Now we assemble the error bounds. Our result from Eq. (A.2) yields

$$\begin{aligned}
\left|\frac{f'(\alpha)}{g'(\alpha)} - H_1(x)\right| &\leq \left|\frac{f'(\alpha)}{g'(\alpha)} - H_1(x(\alpha))\right| + |H_1(x) - H_1(x(\alpha))| \\
&\leq \Big((\alpha-1)H_1(x) + (\alpha-1)H_1(x(\alpha))\Big) + |H_1(x) - H_1(x(\alpha))| \\
&\leq 2(\alpha-1)H_1(x) + \alpha \cdot |H_1(x) - H_1(x(\alpha))| \\
&\leq 2\Big(\xi \log n + \xi \log(1/\xi)\Big)
\end{aligned}$$

This completes the proof. ■

We now use Lemma A.1 to show that $H_\alpha \approx H_1$, if α is sufficiently small.

Proof (of Theorem 3.1). First we focus on the multiplicative approximation. The lower bound is immediate from Claim A.4, so we show the upper-bound. For an arbitrary $\mu \in (0,1)$, we have

$$\mu^2 < \frac{\mu}{2\log(1/\mu)} < \mu;$$

this follows since $\mu \log(1/\mu) < 1/2$ for all μ. Let $\tilde{\mu} = \mu/\big(2\log(1/\mu)\big)$. Then

$$\tilde{\mu} \log(1/\tilde{\mu}) < \mu.$$

This follows since $\mu^2 < \tilde{\mu} \implies 1/\tilde{\mu} < 1/\mu^2 \implies \log(1/\tilde{\mu}) < 2\log(1/\mu)$.

The hypotheses of Theorem 3.1 give $\alpha = 1 + \tilde{\mu}/8$. Hence,

$$\begin{aligned}
e(\alpha) &= 8(\alpha-1)H_1\Big[\log n + \log\Big(1/\big(4(\alpha-1)H_1\big)\Big)\Big] \\
&\leq \tilde{\mu}H_1\Big[\log n + \log\big(2/(\tilde{\mu}H_1)\big)\Big]
\end{aligned}$$

Since $H_1 \geq (\log m)/m$ for any distribution satisfying our hypotheses, this is at most

$$\begin{aligned}
&\leq \tilde{\mu}H_1\Big(\log n + \log(1/\tilde{\mu}) + \log m\Big) \\
&\leq (\log m)\mu H_1 < (\varepsilon/2)H_1,
\end{aligned}$$

since our hypotheses give $\mu = \varepsilon/(4\log m)$. Applying Lemma A.1, we obtain that

$$\begin{aligned}
H_1 - H_\alpha &\leq (\varepsilon/2)H_1 \\
\implies (1-\varepsilon/2)H_1 &\leq H_\alpha \\
\implies \frac{H_1}{H_\alpha} &\leq \frac{1}{1-\varepsilon/2} \leq 1+\varepsilon,
\end{aligned}$$

the last inequality following from Claim A.2. This establishes the multiplicative approximation.

Let us now consider the above argument, replacing μ with $\nu = \varepsilon/(4\log n \log m)$. We obtain

$$e(\alpha) \leq (\log m)\nu H_1 \leq \varepsilon/4,$$

since $H_1 \leq \log n$. Thus, the additive approximation follows directly. ■

A.2 Proofs from Section 3.3

Our first task is to prove Lemma 3.5. We require a definition and two preliminary technical results. For any integer $k \geq 0$ and real number $a \geq -1$, define

$$G_k(a) \;=\; \sum_{i=1}^{n} x_i^{1+a} \log^k(x_i),$$

so $G_0(a) = F_{1+a}/||A||_1^{1+a}$. Note that $G_k^{(1)}(a) = G_{k+1}(a)$ for $k \geq 0$, and $T(a) = (1 - G_0(a))/a$.

Claim A.10. The k^{th} derivative of the Tsallis entropy has the following expression.

$$T^{(k)}(a) \;=\; \frac{(-1)^k\, k!\, \big(1 - G_0(a)\big)}{a^{k+1}} \;-\; \left(\sum_{j=1}^{k} \frac{(-1)^{k-j}\, k!\, G_j(a)}{a^{k-j+1} j!}\right)$$

Proof. The proof is by induction, the case $k = 0$ being trivial. So assume $k \geq 1$. Taking the derivative of the expression for $T^{(k)}(a)$ above, we obtain:

$$\begin{aligned}
&T^{(k+1)}(a)\\
&= \left(\sum_{j=1}^{k} \frac{k!(k-j+1)(-1)^{(k+1)-j}G_j(a)}{a^{(k+1)-j+1}j!} + \frac{k!(-1)^{k-j}G_{j+1}(a)}{a^{k-j+1}j!}\right)\\
&\quad + \frac{(-1)^{k+1}(k+1)!(G_0(a)-1)}{a^{k+2}} + \frac{(-1)^k k! G_1(a)}{a^{k+1}}\\
&= \left(\sum_{j=1}^{k} \frac{k!(-1)^{(k+1)-j}G_j(a)}{a^{(k+1)-j+1}(j-1)!}\left(1 + \frac{k-j+1}{j}\right)\right) + \frac{G_{k+1}(a)}{a} + \frac{(-1)^{k+1}(k+1)!(G_0(a)-1)}{a^{k+2}}\\
&= \left(\sum_{j=1}^{k+1} \frac{(k+1)!(-1)^{(k+1)-j}G_j(a)}{a^{(k+1)-j+1}j!}\right) + \frac{(-1)^{k+1}(k+1)!(G_0(a)-1)}{a^{k+2}}
\end{aligned}$$

as claimed. ■

Claim A.11. Define $S_k(a) = a^{k+1}T^{(k)}(a)$. Then, for $1 \leq j \leq k+1$,

$$S_k^{(j)}(a) = \sum_{i=0}^{j-1} \binom{j-1}{i} \frac{k!}{(k-j+i+1)!} a^{k-j+i+1} G_{k+1+i}(a)$$

In particular, for $1 \leq j \leq k$, we have

$$\lim_{a \to 0} S_k^{(j)}(a) = 0 \qquad \text{and} \qquad \lim_{a \to 0} S_k^{(k+1)}(a) = k!\, G_{k+1}(0) \qquad \text{so that} \qquad \lim_{a \to 0} T^{(k)}(a) = \frac{G_{k+1}(0)}{k+1}.$$

Proof. We prove the claim by induction on j. First, note

$$S_k(a) = (-1)^k k!(1 - G_0(a)) - \left(\sum_{j=1}^{k} \frac{a^j(-1)^{k-j}k!G_j(a)}{j!}\right)$$

so that

$$\begin{aligned} S_k^{(1)}(a) &= (-1)^{k-1}k!G_1(a) - \left(\sum_{j=1}^{k} -\frac{a^{(j+1)-1}(-1)^{k-(j+1)}k!G_{j+1}(a)}{((j+1)-1)!} + \frac{a^{j-1}(-1)^{k-j}k!G_j(a)}{(j-1)!} \right) \\ &= a^k G_{k+1}(a) \end{aligned}$$

Thus, the base case holds. For the inductive step with $2 \le j \le k+1$, we have

$$\begin{aligned} S_k^{(j)}(a) &= \frac{\partial}{\partial a} \left(\sum_{i=0}^{j-2} \binom{j-2}{i} \frac{k!}{(k-j+i+2)!} a^{k-j+i+2} G_{k+1+i}(a) \right) \\ &= \sum_{i=0}^{j-2} \left(\binom{j-2}{i} \frac{k!}{(k-j+i+1)!} a^{k-j+i+1} G_{k+1+i}(a) \right. \\ &\quad \left. + \binom{j-2}{i} \frac{k!}{(k-j+(i+1)+1)!} a^{k-j+(i+1)+1} G_{k+1+(i+1)}(a) \right) \\ &= \sum_{i=0}^{j-1} \binom{j-1}{i} \frac{k!}{(k-j+i+1)!} a^{k-j+i+1} G_{k+1+i}(a) \end{aligned}$$

The final equality holds since $\binom{j-2}{0} = \binom{j-1}{0} = 1$, $\binom{j-2}{j-2} = \binom{j-1}{j-1} = 1$, and by Pascal's formula $\binom{j-2}{i} + \binom{j-2}{i+1} = \binom{j-1}{i+1}$ for $0 \le i \le j-3$.

For $1 \le j \le k$, every term in the above sum is well-defined for $a = 0$ and contains a power of a which is at least 1, so $\lim_{a \to 0} S_k^{(j)}(a) = 0$. When $j = k+1$, all terms but the first term contain a power of a which is at least 1, and the first term is $k!G_{k+1}(a)$, so $\lim_{a \to 0} S_k^{(k+1)}(a) = k!G_{k+1}(0)$. The claim on $\lim_{a \to 0} T(k)(a)$ thus follows by writing $T^{(k)}(a) = S_k(a)/a^{k+1}$ then applying l'Hôpital's rule $k+1$ times. ■

Proof (of Lemma 3.5). We will first show that

$$\left| T^{(k)} \left(-\frac{\varepsilon}{(k+1)\log m} \right) - \frac{G_{k+1}(0)}{k+1} \right| \le \frac{6\varepsilon \log^k(m) H(x)}{k+1}$$

Let $S_k(a) = a^{k+1}T^{(k)}(a)$ and note $T^{(k)}(a) = S_k(a)/a^{k+1}$. By Claim A.10, $\lim_{a \to 0} S_k(a) = 0$. Furthermore, $\lim_{a \to 0} S_k^{(j)} = 0$ for all $1 \le j \le k$ by Claim A.11. Thus, when analyzing $\lim_{a \to 0} S_k^{(j)}(a)/(a^{k+1})^{(j)}$ for $0 \le j \le k$, both the numerator and denominator approach 0 and we can apply l'Hôpital's rule (here $(a^{k+1})^{(j)}$ denotes the jth derivative of the function a^{k+1}). By $k+1$ applications of l'Hôpital's rule, we can thus say that $T^{(k)}(a)$ converges to its limit at least as quickly as $S_k^{(k+1)}(a)/(a^{k+1})^{(k+1)} = S_k^{(k+1)}(a)/(k+1)!$ does (using Claim A.7). We note that $G_j(a)$ is nonnegative for j even and nonpositive otherwise. Thus, for negative a, each term in the summand of the expression for $S_k^{(k+1)}(a)$ in Claim A.11 is nonnegative for odd k and nonpositive for even k. As the analyses for even and odd k are nearly identical, we focus below on odd k, in which case every term in the summand is nonnegative. For odd k, $S_k^{(k+2)}(a)$ is nonpositive so that $S_k^{(k+1)}(a)$ is monotonically decreasing. Thus, it suffices to show that $S_k^{(k+1)}(-\varepsilon/((k+1)\log m))/(k+1)!$ is not much larger than its limit.

$$\begin{aligned}
\frac{S_k^{(k+1)}\left(-\frac{\varepsilon}{(k+1)\log m}\right)}{(k+1)!} &= \frac{\sum_{i=0}^{k}\binom{k}{i}\frac{k!}{i!}\left(-\frac{\varepsilon}{(k+1)\log m}\right)^i G_{k+1+i}\left(-\frac{\varepsilon}{(k+1)\log m}\right)}{(k+1)!} \\
&\leq \frac{1+2\varepsilon}{k+1}\sum_{i=0}^{k}\binom{k}{i}\left(\frac{\varepsilon}{(k+1)\log m}\right)^i |G_{k+1+i}(0)| \\
&\leq \frac{1+2\varepsilon}{k+1}\sum_{i=0}^{k}k^i\left(\frac{\varepsilon}{(k+1)\log m}\right)^i |G_{k+1+i}(0)| \\
&\leq \frac{1+2\varepsilon}{k+1}\sum_{i=0}^{k}\left(\frac{\varepsilon}{\log m}\right)^i |G_{k+1+i}(0)| \\
&\leq \frac{1+2\varepsilon}{k+1}\sum_{i=0}^{k}\varepsilon^i |G_{k+1}(0)| \\
&\leq \frac{(1+2\varepsilon)|G_{k+1}(0)|}{k+1} + \frac{1+2\varepsilon}{k+1}\sum_{i=1}^{k}\varepsilon^i |G_{k+1}(0)| \\
&\leq \frac{(1+2\varepsilon)|G_{k+1}(0)|}{k+1} + \frac{2}{k+1}\sum_{i=1}^{k}\varepsilon^i \log^k(m)H(x) \\
&\leq \frac{|G_{k+1}(0)|}{k+1} + \frac{6\varepsilon\log^k(m)H(x)}{k+1}
\end{aligned}$$

The first inequality holds since $x_i \geq 1/m$ for each i, so that $x_i^{-\varepsilon/((k+1)\log m)} \leq m^{\varepsilon/((k+1)\log m)} \leq m^{\varepsilon/\log m} \leq e^{\varepsilon} \leq 1+2\varepsilon$ for $\varepsilon \leq 1/2$. The final inequality above holds since $\varepsilon \leq 1/2$.

The lemma follows since $|G_{k+1}(0)| \leq \log^k(m)H(x)$. ■

Proof (of Lemma 3.6). Let P_j denote the j^{th} Chebyshev polynomial. We will prove for all $j \geq 1$ that

$$P_{j-1}(x) \leq P_j(x) \leq P_{j-1}(x)\left(1+\frac{2j}{k^c}\right).$$

For the first inequality, we observe $P_{j-1} \in \mathcal{C}_j$, so we apply Fact 3.3 together with the fact that $P_j(y)$ is strictly positive for $y > 1$ for all j.

For the second inequality, we induct on j. For the sake of the proof define $P_{-1}(x) = 1$ so that the inductive hypothesis holds at the base case $d = 0$. For the inductive step with $j \geq 1$, we use the recurrence definition of $P_j(x)$ and we have

$$\begin{aligned}
P_{j+1}(x) &= P_j(x)\left(1+\frac{2}{k^c}\right) + (P_j(x) - P_{j-1}(x)) \\
&\leq P_j(x)\left(1+\frac{2}{k^c}\right) + P_{j-1}(x)\frac{2j}{k^c} \\
&\leq P_j(x)\left(1+\frac{2}{k^c}\right) + P_j(x)\frac{2j}{k^c} \\
&= P_j(x)\left(1+\frac{2(j+1)}{k^c}\right)
\end{aligned}$$

■

A.3 Proofs from Section 4

Fact A.12. For any real $z > 0$, $\Gamma(z+1) = z\Gamma(z)$.

Fact A.13. For any real $z \geq 0$, $\sin(z) \leq z$.

Fact A.14 (Euler's Reflection Formula). For any real z, $\Gamma(z)\Gamma(1-z) = \pi/\sin(\pi z)$.

Fact A.15. $\frac{\partial}{\partial z}\Gamma(z) = \Gamma(z)\psi_0(z)$, where $\psi_0(\cdot)$ is the digamma function, which is continuous on $\mathbb{R}_+$.

Definition A.16. The function $V : \mathbb{R}^+ \to \mathbb{R}$ is defined by

$$V(\alpha) = \frac{\left[\frac{2}{\pi}\Gamma(\frac{2\alpha}{3})\Gamma(\frac{1}{3})\sin(\frac{\pi\alpha}{3})\right]^3}{\left[\frac{2}{\pi}\Gamma(\frac{\alpha}{3})\Gamma(\frac{2}{3})\sin(\frac{\pi\alpha}{6})\right]^6} - 1$$

Lemma A.17.

$$\lim_{\alpha\to 0} V(\alpha) = \frac{\Gamma\left(\frac{1}{3}\right)^3}{\Gamma\left(\frac{2}{3}\right)^6}$$

Proof. Define $u(\alpha) = \Gamma(2\alpha/3)(\pi\alpha/3) = \Gamma(2\alpha/3)(2\alpha/3)(\pi/2) = \Gamma((2\alpha/3)+1)(\pi/2)$ by Fact A.12. By the continuity of $\Gamma(\cdot)$ on $\mathbb{R}_+$, $\lim_{\alpha\to 0} u(\alpha) = \Gamma(1)\pi/2 = \pi/2$. Define $f(\alpha) = \Gamma(2\alpha/3)\sin(\pi\alpha/3)$. Then $f(\alpha) \leq u(\alpha)$ for all $\alpha \geq 0$ by Fact A.13, and thus $\lim_{\alpha\to 0} f(\alpha) \leq \pi/2$. Now define $\ell_\delta(\alpha) = \Gamma(2\alpha/3)(1-\delta)(\pi\alpha/3)$. By the definition of the derivative and the fact that the derivative of $\sin(\alpha)$ evaluated at $\alpha = 1$ is 1, it follows that $\forall\delta > 0\ \exists\varepsilon > 0$ s.t. $0 \leq \alpha < \varepsilon \Rightarrow \sin(\alpha) \geq (1-\delta)\alpha$. Thus, $\forall\delta > 0\ \exists\varepsilon > 0$ s.t. $0 \leq \alpha < \varepsilon \Rightarrow \ell_\delta(\alpha) \leq f(\alpha)$, and so $\forall\delta > 0$ we have that $\lim_{\alpha\to 0} f(\alpha) \geq \lim_{\alpha\to 0} \ell_\delta(\alpha) = (1-\delta)\pi/2$. Thus, $\lim_{\alpha\to 0} f(\alpha) \geq \pi/2$, implying $\lim_{\alpha\to 0} f(\alpha) = \pi/2$. Similarly we can define $g(\alpha) = \Gamma(\alpha/3)\sin(\pi\alpha/6)$ and show $\lim_{\alpha\to 0} g(\alpha) = \pi/2$.

Now,

$$V(\alpha) = \frac{\left[\frac{2}{\pi}\Gamma\left(\frac{1}{3}\right) f(\alpha)\right]^3}{\left[\frac{2}{\pi}\Gamma\left(\frac{2}{3}\right) g(\alpha)\right]^6}$$

Thus $\lim_{\alpha\to 0} V(\alpha) = \Gamma(1/3)^3/\Gamma(2/3)^6$ as claimed. ■

Proof (of Lemma 4.2). Li shows in [20] that the variance of the geometric mean estimator with $k = 3$ is $V(\alpha)F_\alpha^2$. As $\Gamma(z)$ and $\sin(z)$ are continuous for $z \in \mathbb{R}_+$, so is $V(\alpha)$. Futhermore Lemma A.17 shows that $\lim_{\alpha\to 0} V(\alpha)$ exists (and equals $(\Gamma(1/3)^3/\Gamma(2/3)^6) - 1$). We define $V(0)$ to be this limit. Thus $V(\alpha)$ is continuous on $[0, 2]$, and the extreme value theorem implies there exists a constant C_{GM} such that $V(\alpha) \leq C_{GM}$ on $[0, 2]$. ■

A.4 Detailed Analysis of Geometric Mean Residual Moments Algorithm

Formally, define $R = \left\lceil \log_{1+\frac{\varepsilon}{c_1}} m \right\rceil$, and let $I_z = \left\{ i : (1+\frac{\varepsilon}{c_1})^z \leq |A_i| < (1+\frac{\varepsilon}{c_1})^{z+1} \right\}$ for $0 \leq z \leq R$. Let z^* satisfy $(1+\frac{\varepsilon}{c_1})^{z^*} \leq |A_1| < (1+\frac{\varepsilon}{c_1})^{z^*+1}$. For $1 \leq j \leq r$ and $0 \leq z \leq R$, define $X_{j,z} = \sum_{i\in I_z} \mathbf{1}_{h_j(i)\neq h_j(1)}$. We now analyze the j^{th} trial.

Claim A.18. $\mathrm{E}\left[2 \cdot F_{\alpha,j,1-h_j(1)}\right] = \big(1 + O(\varepsilon)\big) \cdot F_\alpha^{\text{res}}$.

Proof. We have

$$\begin{aligned}
\mathrm{E}\left[\, 2 \cdot F_{\alpha,j,1-h_j(1)} \,\right] &= 2 \cdot \mathrm{E}\left[\sum_i |A_i|^\alpha \cdot \mathbf{1}_{h_j(i) \neq h_j(1)} \right] \\
&= 2 \cdot \sum_z \mathrm{E}\left[\sum_{i \in I_z} |A_i|^\alpha \cdot \mathbf{1}_{h_j(i) \neq h_j(1)} \right] \\
&= 2 \cdot \sum_z \mathrm{E}\left[\sum_{i \in I_z} \big((1 \pm \varepsilon)(1+\varepsilon)^z \big)^\alpha \cdot \mathbf{1}_{h_j(i) \neq h_j(1)} \right] \\
&= (1 \pm \varepsilon)^\alpha \cdot \sum_z (1+\varepsilon)^{z\alpha}\, \mathrm{E}\left[\, 2X_{j,z} \,\right].
\end{aligned}$$

Clearly $\mathrm{E}\left[\, 2 \cdot X_{j,z} \,\right]$ is $|I_z| - 1$ if $z = z^*$ and $|I_z|$ otherwise. Thus

$$\sum_z (1+\varepsilon)^{z\alpha}\, \mathrm{E}\left[\, 2 \cdot X_{j,z} \,\right] \;=\; \sum_{i \geq 2} \big((1 \pm \varepsilon)|A_i| \big)^\alpha \;=\; (1 \pm \varepsilon)^\alpha \cdot F_\alpha^{\mathrm{res}}.$$

Since $\alpha < 2$, $(1 \pm \varepsilon)^\alpha = 1 \pm O(\varepsilon)$, so this shows the desired result. ■

We now show concentration for $X_z := \frac{1}{r} \sum_{1 \leq j \leq r} X_{j,z}$. By independence of the h_j's, Chernoff bounds show that $X_z = (1 \pm \varepsilon)\, \mathrm{E}\left[\, X_z \,\right]$ with probability at least $1 - \exp(-\Theta(\varepsilon^2 r))$. This quantity is at least $1 - \frac{1}{8(R+1)}$ if we choose $r = c_2 \left\lceil \varepsilon^{-2} (\log\log ||A||_1 + \log(c_3/\varepsilon)) \right\rceil$. The *good event* is the event that, for all z, $X_z = (1 \pm \varepsilon)\, \mathrm{E}\left[\, X_z \,\right]$; a union bound shows that this occurs with probability at least 7/8. So suppose that the good event occurs. Then a calculation analogous to Claim A.18 shows that

$$\begin{aligned}
\sum_j \frac{2}{r} \cdot F_{\alpha,j,1-h_j(1)} &= (1 \pm \varepsilon)^\alpha \cdot \sum_z (1+\varepsilon)^{z\alpha} \cdot 2X_z \\
&= (1 \pm \varepsilon)^\alpha \cdot \sum_z (1+\varepsilon)^{z\alpha} \cdot (1 \pm \varepsilon)\, \mathrm{E}\left[\, 2X_z \,\right] \\
&= \big(1 \pm O(\varepsilon) \big) \cdot F_\alpha^{\mathrm{res}}. \qquad \text{(A.4)}
\end{aligned}$$

Recall that $\tilde{F}_\alpha^{\mathrm{res}} = \sum_{j=1}^r \frac{2}{r} \tilde{F}_{\alpha,j,1-h_j(1)}$. Since the geometric mean estimator is unbiased, we also have that

$$\mathrm{E}\left[\, \tilde{F}_\alpha^{\mathrm{res}} \,\right] \;=\; \mathrm{E}\left[\sum_j \frac{2}{r} F_{\alpha,j,1-h_j(1)} \right]. \qquad \text{(A.5)}$$

We conclude the analysis by showing that the random variable $\tilde{F}_\alpha^{\mathrm{res}}$ is concentrated. By Lemma 4.2 applied to each substream, and properties of variance, we have

$$\mathrm{Var}\left[\, \tilde{F}_\alpha^{\mathrm{res}} \,\right] \;=\; \frac{4}{r^2} \sum_{j=1}^r \mathrm{Var}\left[\, \tilde{F}_{\alpha,j,1-h_j(1)} \,\right] \;\leq\; \frac{4\, C_{GM}}{r} \cdot \mathrm{E}\left[\, \tilde{F}_{\alpha,j,1-h_j(1)} \,\right]^2 \;\leq\; \frac{C_{GM}}{r} \cdot \mathrm{E}\left[\, \tilde{F}_\alpha^{\mathrm{res}} \,\right]^2.$$

Chebyshev's inequality therefore shows that

$$\Pr\left[\; \tilde{F}_\alpha^{\mathrm{res}} = (1 \pm \varepsilon)\, \mathrm{E}\left[\, \tilde{F}_\alpha^{\mathrm{res}} \,\right] \;\right] \;\geq\; 1 - \frac{\mathrm{Var}\left[\, \tilde{F}_\alpha^{\mathrm{res}} \,\right]}{(\varepsilon \cdot \mathrm{E}\left[\, \tilde{F}_\alpha^{\mathrm{res}} \,\right])^2} \;\geq\; 1 - \frac{C_{GM}}{\varepsilon^2\, r} \;>\; 6/7,$$

by appropriate choice of constants. This event and the good event both occur with probability at least 3/4. When this holds, we have

$$\tilde{F}_\alpha^{\text{res}} \;=\; (1 \pm \varepsilon)\,\mathrm{E}\left[\tilde{F}_\alpha^{\text{res}}\right] \;=\; (1 \pm \varepsilon)\,\mathrm{E}\left[\sum_j \frac{2}{r} F_{\alpha,j,1-h_j(1)}\right] \;=\; \big(1 \pm O(\varepsilon)\big) \cdot F_\alpha^{\text{res}},$$

by Eq. (A.5) and Eq. (A.4).

A.5 Proofs from Section 4.2

Proof (of Fact 4.1). Let $B = \lceil 20/\varepsilon \rceil$ be the number of bins. Let $\mathcal{H}$ be a pairwise independent family of hash functions, each function mapping $[n]$ to $[B]$. Standard constructions yield such a family with $|\mathcal{H}| = n^{O(1)}$. We will let h be a randomly chosen hash function from $\mathcal{H}$.

For notational simplicity, suppose that $x_1 = \max_i x_i$. Let $\mathcal{E}_{i,j}$ be the indicator variable for the event that $h(i) = j$, so that $\mathrm{E}\left[\,\mathcal{E}_{i,j}\,\right] = 1/B$ and $\mathrm{Var}\left[\,\mathcal{E}_{i,j}\,\right] < 1/B$. Let X_j be the random variable denoting the weight of the items that hash to bin j, i.e., $X_j = \sum_i x_i \cdot \mathcal{E}_{i,j}$. Since $\sum_i x_i = 1$, we have $\mathrm{E}\left[\,X_j\,\right] = 1/B$ and $\mathrm{Var}\left[\,X_j\,\right] < \|x\|_2^2/B$.

Suppose that $x_1 \geq 1/2$. Let Y be the fraction of mass that hashes to x_1's bin, excluding x_1 itself. That is, $Y = \sum_{i \geq 2} x_i \cdot \mathcal{E}_{i,h(1)}$. Note that $\mathrm{E}\left[\,Y\,\right] = (\sum_{i\geq 2} x_i)/B < (\varepsilon/20) \cdot (\sum_{i \geq 2} x_i)$. By Markov's inequality,

$$\Pr\left[\,Y \geq \varepsilon \cdot (\textstyle\sum_{i\geq 2} x_i)\,\right] \;\leq\; \Pr\left[\,Y \geq 16\,\mathrm{E}\left[\,Y\,\right]\,\right] \;\leq\; 1/16.$$

Suppose that $x_1 < 1/2$. This implies, by convexity, that $\|x\|_2^2 < 1/2$. Let $\beta = \sqrt{2/3} < 5/6$. Then

$$\Pr\left[\,|X_j - 1/B| \geq \beta\,\right] \;\leq\; \frac{\mathrm{Var}\left[\,X_j\,\right]}{\beta^2} \;<\; \frac{3}{4B}.$$

Thus, by a union bound,

$$\Pr\left[\,\exists j \text{ such that } X_j \geq \beta + 1/B\,\right] \;\leq\; \frac{3}{4}.$$

Suppose we want to test if $x_1 \geq 1/2$ by checking if there's a bin of mass at least 5/6. As argued above, the failure probability of one hash function is at most 3/4. If we choose ten independent hash functions and check that all of them have a bin of at least 5/6, then the failure probability decreases to less than 1/16. ■

A.6 Proofs from Section 5

Proof (of Theorem 5.1). Let m_i be the number of times the i-th element appears in the stream. Recall that m is the length of the stream. By computing a $(1+\varepsilon')$-approximation to the α^{th} moment (as in Fact 2.1) and dividing by $||A||_1^\alpha$, we get a multiplicative approximation to $F_\alpha/||A||_1^\alpha = ||x||_\alpha^\alpha$. We can thus compute the value

$$\frac{1}{1-\alpha}\log\left((1 \pm \varepsilon')\sum_{i=1}^n x_i^\alpha\right) = \frac{1}{1-\alpha}\log\left(\sum_{i=1}^n x_i^\alpha\right) + \frac{\log(1 \pm \varepsilon')}{1-\alpha} = H_\alpha(X) \pm \frac{\varepsilon'}{1-\alpha}.$$

Setting $\varepsilon' = \varepsilon \cdot |1-\alpha|$, we obtain an additive approximation algorithm using

$$O\left(\left(\frac{|1-\alpha|}{\varepsilon^2 \cdot |\alpha-1|^2} + \frac{1}{\varepsilon \cdot |\alpha-1|}\right)\log m\right) \;=\; O(\log m/(|1-\alpha| \cdot \varepsilon^2))$$

bits, as claimed. ■

Proof (of Theorem 5.2). If $\alpha \in (0,1)$, then because the function x^α is concave, we get by Jensen's inequality

$$\sum_{i=1}^{n} {x_i}^\alpha \leq n \cdot \left(\frac{1}{n}\right)^\alpha = n^{1-\alpha}.$$

If we compute a multiplicative $(1 + (1-\alpha) \cdot \varepsilon \cdot n^{\alpha-1})$-approximation to the α^{th} moment, we obtain an additive $(1-\alpha) \cdot \varepsilon$-approximation to $(\sum_{i=1}^n x_i^\alpha) - 1$. This in turn gives an additive ε-approximation to T_α. By Fact 2.1,

$$O\left(\left(\frac{1-\alpha}{((1-\alpha)\cdot\varepsilon\cdot n^{\alpha-1})^2} + \frac{1}{(1-\alpha)\cdot\varepsilon\cdot n^{\alpha-1}}\right)\log m\right) \;=\; O(n^{2(1-\alpha)}\log m/((1-\alpha)\varepsilon^2))$$

bits of space suffice to achieve the required approximation to the α^{th} moment.

For $\alpha > 1$, the value $F_\alpha/||A||_1^\alpha$ is at most 1, so it suffices to approximate F_α to within a factor of $1 + (\alpha-1)\cdot\varepsilon$. For $\alpha \in (1,2]$, again using Fact 2.1, we can achieve this using $O(\log m/((\alpha-1)\varepsilon^2))$ bits of space. ■

Proof (of Lemma 5.3). Consider first $\alpha \in (0,1)$. For $x \in (0, 5/6]$,

$$\frac{x^\alpha}{x} = x^{\alpha-1} \geq \left(\frac{5}{6}\right)^{\alpha-1} \geq 1 + C_1 \cdot (1-\alpha),$$

for some positive constant C_1. The last equality follows from convexity of $(5/6)^y$ as a function of y. Hence,

$$\sum_{i=1}^{n} x_i^\alpha \geq \sum_{i=1}^{n}(1 + C_1(1-\alpha))x_i = 1 + C_1(1-\alpha),$$

and furthermore,

$$\left|1 - \sum_{i=1}^{n} x_i^\alpha\right| = \left(\sum_{i=1}^{n} x_i^\alpha\right) - 1 \geq C_1 \cdot (1-\alpha) = C_1 \cdot |\alpha - 1|$$

When $\alpha \in (1,2]$, then for $x \in (0, 5/6]$,

$$\frac{x^\alpha}{x} = x^{\alpha-1} \leq \left(\frac{5}{6}\right)^{\alpha-1} \leq 1 - C_2 \cdot (\alpha-1),$$

for some positive constant C_2. This implies that

$$\sum_{i=1}^{n} x_i^\alpha \leq \sum_{i=1}^{n} x_i(1 - C_2\cdot(\alpha-1)) = 1 - C_2 \cdot (\alpha-1),$$

and

$$\left|1 - \sum_{i=1}^{n} x_i^\alpha\right| = 1 - \sum_{i=1}^{n} x_i^\alpha \geq C_2 \cdot (\alpha-1) = C_2 \cdot |\alpha-1|.$$

To finish the proof of the lemma, we set $C = \min\{C_1, C_2\}$. ■

Proof (of Lemma 5.5). We first argue that a multiplicative approximation to $|1 - x_i^\alpha|$ can be obtained from a multiplicative approximation to $1 - x_i$. Let $g(y) = 1 - (1 - y)^\alpha$. Note that $g(1 - x_i) = 1 - x_i^\alpha$. Since $1 - x_i \in [0, 1/3]$, we restrict the domain of g to $[0, 1/3]$. The derivative of g is $g'(y) = \alpha(1 - y)^{\alpha-1}$. Note that g is strictly increasing for $\alpha \in (0, 1) \cup (1, 2]$. For $\alpha \in (0, 1)$, the derivative is in the range $[\alpha, \frac{3}{2}\alpha]$. For $\alpha \in (1, 2]$, it always lies in the range $[\frac{2}{3}\alpha, \alpha]$. In both cases, a $(1 + \frac{2}{3}\varepsilon)$-approximation to y suffices to compute a $(1 + \varepsilon)$-approximation to $g(y)$.

We now consider two cases:

- Assume first that $\alpha \in (0, 1)$. For any $x \in (0, 1/3]$, we have

  $$\frac{x^\alpha}{x} \geq \left(\frac{1}{3}\right)^{\alpha-1} = 3^{1-\alpha} \geq 1 + C_1(1 - \alpha),$$

  for some positive constant C_1. The last inequality follows from the convexity of the function $3^{1-\alpha}$. This means that if $x_i < 1$, then

  $$\frac{\sum_{j\neq i} x_j^\alpha}{1 - x_i} \geq \frac{\sum_{j\neq i} x_j(1 + C_1(1 - \alpha))}{1 - x_i} = \frac{(1 - x_i)(1 + C_1(1 - \alpha))}{1 - x_i} = 1 + C_1(1 - \alpha).$$

  Since $x_i \leq x_i^\alpha < 1$, we also have

  $$\frac{\sum_{j\neq i} x_j^\alpha}{1 - x_i^\alpha} \geq \frac{\sum_{j\neq i} x_j^\alpha}{1 - x_i} \geq 1 + C_1(1 - \alpha).$$

  This implies that if we compute a multiplicative $1 + (1 - \alpha)\varepsilon/D_1$-approximations to both $1 - x_i^\alpha$ and $\sum_{j\neq i} x_j^\alpha$, for sufficiently large constant D_1, we compute a multiplicative $(1 + \varepsilon)$-approximation of $(\sum_{j=1}^n x_j^\alpha) - 1$.

- The case of $\alpha \in (1, 2]$ is similar. For any $x \in (0, 1/3]$, we have

  $$\frac{x^\alpha}{x} \leq \left(\frac{1}{3}\right)^{\alpha-1} \leq 1 - C_2(\alpha - 1),$$

  for some positive constant C_2. Hence,

  $$\frac{\sum_{j\neq i} x_j^\alpha}{1 - x_i} \leq \frac{\sum_{j\neq i} x_j(1 - C_2(\alpha - 1))}{1 - x_i} = \frac{(1 - x_i)(1 - C_2(\alpha - 1))}{1 - x_i} = 1 - C_2(\alpha - 1),$$

  and because $x_i^\alpha \leq x_i$,

  $$\frac{\sum_{j\neq i} x_j^\alpha}{1 - x_i^\alpha} \leq \frac{\sum_{j\neq i} x_j^\alpha}{1 - x_i} \leq 1 - C_2(\alpha - 1).$$

  This implies that if we compute a multiplicative $1 + (\alpha - 1)\varepsilon/D_2$-approximations to both $1 - x_i^\alpha$ and $\sum_{j\neq i} x_j^\alpha$, for sufficiently large constant D_2, we can compute a multiplicative $(1 + \varepsilon)$-approximation to $1 - \sum_{j=1}^n x_j^\alpha$.

■

Proof (of Theorem 5.6). We run the algorithm of Section 4.1 to find out if there is a very heavy element. This only requires $O(\log n)$ words of space.

If there is no heavy element, then by Lemma 5.3 there is a constant $C \in (0,1)$ such that $|1-\sum_i x_i^\alpha| \geq C|\alpha-1|$. We want to compute a multiplicative approximation to $|1-\sum_i x_i^\alpha|$. We know that the difference between $\sum_i x_i^\alpha$ and 1 is large. Therefore, if we compute a multiplicative $(1+\frac{1}{2}|\alpha-1|C\varepsilon)$-approximation to $\sum_i x_i^\alpha$, we obtain an additive $(\frac{1}{2}|\alpha-1|C\varepsilon\sum_i x_i^\alpha)$-approximation to $\sum_i x_i^\alpha$. If $\sum_i x_i^\alpha \leq 2$, then

$$\frac{\frac{1}{2}|\alpha-1|C\varepsilon\sum_i x_i^\alpha}{|1-\sum_i x_i^\alpha|} \leq \frac{|\alpha-1|C\varepsilon}{C|\alpha-1|} = \varepsilon.$$

If $\sum_i x_i^\alpha \geq 2$, then

$$\frac{\frac{1}{2}|\alpha-1|C\varepsilon\sum_i x_i^\alpha}{|1-\sum_i x_i^\alpha|} \leq \frac{1}{2}|\alpha-1|C\varepsilon\cdot 2 \leq \varepsilon.$$

In either case, we obtain a multiplicative $(1+\varepsilon)$-approximation to $|1-\sum_i x_i^\alpha|$, which in turn yields a multiplicative approximation to the Tsallis entropy. We now need to bound the amount of space we use in this case. We use the estimator of Fact 2.1, which uses $O(\log m/(|\alpha-1|\varepsilon^2))$ bits in our case.

Let us focus now on the case when there is a heavy element. By Lemma 5.5 it suffices to approximate F_1^{res} and F_α^{res}, which we can do using the algorithm of Section 4.2. The number of bits required is

$$O\left(\frac{\log m}{\varepsilon\cdot|\alpha-1|}\right) + \tilde{O}\left(\frac{|\alpha-1|\cdot\log m}{(\varepsilon\cdot|\alpha-1|)^2}\right) = \tilde{O}\left(\frac{\log m}{\varepsilon^2\cdot|\alpha-1|}\right).$$

■

Proof (of Lemma 5.7). For $t \in [4/9, 1]$, the derivative of the logarithm function lies in the range $[a,b]$, where a and b are constants such that $0 < a < b$. This implies that in this case, a $(1+\varepsilon)$-approximation to $t-1$ gives a $1+\frac{b}{a}\varepsilon$ approximation to $\log(t)$. We are given $y \in [1-t, (1+\varepsilon)(1-t)]$, and we can assume that $y \in [1-t, \min\{5/9, (1+\varepsilon)(1-t)\}]$. We have

$$-\log(t) \leq -\log(1-y),$$

and

$$\begin{aligned}
\frac{-\log(1-y)}{-\log(t)} &\leq \frac{-\log(1-(1+\varepsilon)(1-t))}{-\log(t)} = \frac{-\log(t-\varepsilon(1-t))}{-\log(t)} \\
&\leq \frac{-\log(t)+(-\log(t-\varepsilon(1-t))+\log(t))}{-\log(t)} \\
&\leq 1+\frac{-\log(t-\varepsilon(1-t))+\log(t)}{-\log(t)} \\
&\leq 1+\frac{\varepsilon(1-t)\cdot\max_{z\in[\max\{t-\varepsilon(1-t),4/9\},t]}(\log(z))'}{(1-t)\cdot\min_{z\in[4/9,1]}(\log(z))'} \\
&\leq 1+\frac{\varepsilon(1-t)\cdot\max_{z\in[t,1]}(\log(z))'}{(1-t)\cdot\min_{z\in[4/9,1]}(\log(z))'} \\
&\leq 1+\frac{\varepsilon(1-t)\cdot b}{(1-t)\cdot a} = 1+\frac{b}{a}\varepsilon.
\end{aligned}$$

Consider now $t > 1$. We are given $y \in [t-1, (1+\varepsilon)(t-1)]$, and we have

$$\log(t) \leq \log(y+1) \leq \log((1+\varepsilon)(t-1)+1).$$

Furthermore,

$$
\begin{aligned}
\frac{\log((1+\varepsilon)(t-1)+1)}{\log(t)} &\leq \frac{\log(t)+\log((1+\varepsilon)(t-1)+1)-\log(t)}{\log(t)} \\
&= 1+\frac{\log(t+(t-1)\varepsilon)-\log(t)}{\log(t)} \\
&= 1+\frac{\int_t^{t+(t-1)\varepsilon}(\log(z))'dz}{\int_1^t(\log(z))'dz} \\
&\leq 1+\frac{(t-1)\varepsilon\max_{z\in[t,t+(t-1)\varepsilon]}(\log(z))'}{(t-1)\max_{z\in[1,t]}(\log(z))'} \\
&\leq 1+\frac{(t-1)\varepsilon}{t-1}=1+\varepsilon.
\end{aligned}
$$

Hence, we get a good multiplicative approximation to $\log(t)$. ■

Proof (of Theorem 5.8). We use the algorithm of Section 4.1 to check if there is a single element of high frequency. This only requires $O(\log m)$ bits of space.

If there is no element of frequency greater than 5/6, then the Rényi entropy for any α is greater than the min-entropy $H_\infty = -\log\max_i x_i \geq \log(6/5)$. Therefore, in this case it suffices to run the additive approximation algorithm with $\varepsilon' = \log(6/5)\varepsilon$ to obtain a sufficiently good estimate. To run that algorithm, we use $O\left(\frac{\log m}{|1-\alpha|\varepsilon^2}\right)$ bits of space.

Let us consider the other case, when there is an element of frequency at least 2/3. For $\alpha \in (1,2]$, we have

$$\left(\frac{2}{3}\right)^2 \leq \sum_i x_i^\alpha \leq 1,$$

and for $\alpha \in (0,1)$, $\sum_{i=1}^n x_i^\alpha \geq 1$. Therefore, by Lemma 5.7, it suffices to compute a multiplicative approximation to $|1-\sum_i x_i^\alpha|$, which we can do by Lemma 5.5. By algorithms from Section 4.3 and Section 4.2, we can compute the multiplicative $(1+\Theta(|1-\alpha|\varepsilon))$-approximations required by Lemma 5.5 with the same space complexity as for the approximation of Tsallis entropy (see the proof of Theorem 5.6). ■

Proof (of Theorem 5.9). The proof is nearly identical to that of Theorem 3.1 in [2]. We need merely observe that if $\tilde{H}_\alpha$ is a $(1+\varepsilon)$-approximation to H_α, then $m^{\alpha(1+\varepsilon)}2^{(1-\alpha)\tilde{H}_\alpha}$ is a multiplicative $m^{\alpha\varepsilon}$-approximation to F_α. From here, we set $t = cm^\varepsilon n^{1/\alpha}$ and argue identically as in [2] via a reduction from t-party disjointness; we omit the details. ■