Supplementary Material for: Estimating the Unseen: Improved Estimators for Entropy and Other Properties

GREGORY VALIANT, Stanford University
PAUL VALIANT, Brown University

This Supplemental Material contains the following: (A) a self-contained treatment of the two distribution setting, containing a proof of Theorem 5.6; (B) some additional empirical results showing that the performance of Algorithm 1 is robust to natural variations and the choice of parameters; and (C) a Matlab implementation of Algorithm 1, which is also available from our websites.

A PROPERTIES OF PAIRS OF DISTRIBUTIONS

Our general approach for constructing constant-factor optimal estimators for symmetric properties of distributions can be extended to yield constant-factor optimal estimators for symmetric properties of *pairs* of distributions, including total variation distance (ℓ_1 distance). In analogy with the single–distribution setting, given a pair of distributions over a common domain, a property of the pair of distributions is symmetric if its value is invariant to permutations of the domain.

For properties of pairs of distributions, an estimator receives two samples as input, one drawn from the first distribution and one drawn from the second distribution. As with the analysis of estimators for properties of a single distribution, we begin by extending our definitions of *fingerprints* and *histograms* to this two–distribution setting.

Definition A.1. The *fingerprint* $\mathcal{F}$ of a sample of size n_1 from distribution p_1 and a sample of size n_2 from distribution p_2 is a $n_1 \times n_2$ matrix, whose entry $\mathcal{F}(i,j)$ is given by the number of domain elements that are seen exactly i times in the sample from p_1 and exactly j times in the sample from p_2.

Definition A.2. The *histogram* $h_{p_1,p_2} : [0,1]^2 \setminus \{(0,0)\} \to \mathbb{N} \cup 0$ of a pair of distributions p_1, p_2 is defined by letting $h_{p_1,p_2}(x,y)$ be the number of domain elements that occur with probability x in distribution p_1 and probability y in distribution p_2.

Thus in any two-dimensional histogram h corresponding to a pair of distributions, we have

$$\sum_{x,y:h(x,y)\neq 0} x \cdot h(x,y) = \sum_{x,y:h(x,y)\neq 0} y \cdot h(x,y) = 1,$$

and $\sum_{x,y:h(x,y)\neq 0} h(x,y) \leq 2k$, as we take k to be a bound on the support size of each distribution. In our analysis, it will prove convenient to also consider "generalized histograms" in which the entries need not be integral, and for which the "probability masses" $\sum_{x,y:h(x,y)\neq 0} x \cdot h(x,y)$ and $\sum_{x,y:h(x,y)\neq 0} y \cdot h(x,y)$ do not necessarily equal 1.

As in the case with symmetric properties of single distributions, symmetric properties of pairs of distributions are functions of only the histogram of the pair of distributions, and given any estimator that takes as input the actual pair of samples, there is an estimator of equivalent performance that takes as input the fingerprint $\mathcal{F}$ derived from such a pair of samples.

Both total variation distance (ℓ_1 distance), and Kullback-Leibler divergence are symmetric properties:

https://doi.org/10.1145/3125643

Example A.3. Consider a pair of distributions p_1, p_2 with histogram h:

- The total variation distance (ℓ_1 distance) is given by
$$D_{tv}(p_1, p_2) = \frac{1}{2} \sum_{(x,y):h(x,y)\neq 0} h(x,y) \cdot |x - y|.$$
- The Kullback-Leibler divergence is given by
$$D_{KL}(p_1||p_2) = \sum_{(x,y):h(x,y)\neq 0} h(x,y) \cdot x \log \frac{x}{y}.$$

We will use the following two-dimensional earthmover metric on the set of two-dimensional generalized histograms. Note that it does not make sense to define a strict analog of the relative earthmover distance of Definition 1.6, since a given histogram entry $h(x,y)$ does not correspond to a single quantity of probability mass—it corresponds to $xh(x,y)$ mass in one distribution, and $yh(x,y)$ mass in the other distribution. Thus the following metric is in terms of moving *histogram entries* rather than probability mass.

Definition A.4. Given two two-dimensional generalized histograms h_1, h_2, their *histogram distance*, denoted $W(h_1, h_2)$, is defined to be the minimum over all schemes of moving the histogram values in h_1 to yield h_2, where the cost of moving histogram value c at location x, y to location x', y' is $c\left(|x - x'| + |y - y'|\right)$. To ensure that such a scheme always exists, in the case that $\sum_{x,y:x+y>0} h_1(x,y) < \sum_{x,y:x+y>0} h_2(x,y)$, one proceeds as if
$$h_1(0,0) = \sum_{x,y:x+y>0} h_2(x,y) - \sum_{x,y:x+y>0} h_1(x,y),$$
and analogously for the case in which h_2 contains fewer histogram entries.

We provide an example of the above definitions:

Example A.5. Define distributions $p_1 = Unif[k]$, and $p_2 = Unif[k/2]$, where the $k/2$ support elements of distribution p_2 are contained in the support of p_1. The corresponding histogram h_{p_1,p_2}, is defined as $h_{p_1,p_2}(\frac{1}{k}, \frac{2}{k}) = \frac{k}{2}$, $h_{p_1,p_2}(\frac{1}{k}, 0) = \frac{k}{2}$, and $h_{p_1,p_2}(x,y) = 0$ for all other values of x, y.

Considering a second pair of distributions, $q_1 = q_2 = Unif[k/4]$, with histogram $h_{q_1,q_2}(\frac{4}{k}, \frac{4}{k}) = \frac{k}{4}$, we have
$$W(h_{p_1,p_2}, h_{q_1,q_2}) = \frac{k}{4}\left(\left|\frac{1}{k} - \frac{4}{k}| + |\frac{2}{k} - \frac{4}{k}\right|\right) + \frac{k}{4}\left(\left|\frac{1}{k} - 0| + |\frac{2}{k} - 0\right|\right)$$
$$+\frac{k}{2}\left(\left|\frac{1}{k} - 0| + |0 - 0\right|\right) = \frac{5}{2},$$
since the optimal scheme is to move $k/4$ histogram entries in h_{p_1,p_2} from $(1/k, 2/k)$ to location $(4/k, 4/k)$, and all the remaining histogram entries must be moved to $(0,0)$ to yield histogram h_{q_1,q_2}.

We note that ℓ_1 distance is 1-Lipschitz with respect to the above distance metric:

Fact 4. *For any pair of two-dimensional generalized histograms, h, h'*
$$W(h, h') \geq \left| \sum_{x,y:h(x,y)\neq 0} h(x,y)|x - y| - \sum_{x,y:h'(x,y)\neq 0} h'(x,y)|x - y| \right|.$$

Hence if $h = h_{p_1,p_2}$ and $h' = h_{q_1,q_2}$ are histograms corresponding to pairs of distributions, then $W(h_{p_1,p_2}, h_{q_1,q_2}) \geq |D_{tv}(p_1,p_2) - D_{tv}(q_1,q_2)|$.

Both our algorithm for estimating properties of pairs of distributions, and its analysis parallel their analogs in the one-distribution setting. For simplicity, we restrict our attention to the setting in which one obtains samples of size n from both distributions—though our approach extends naturally to the setting in which one obtains samples of different sizes from the two distributions.

THEOREM A.6 (5.6). *There exist absolute constants $\alpha, \gamma > 0$ such that for any $c > 0$, for sufficiently large k, given two samples of size $n = c\frac{k}{\log k}$ consisting of independent draws from each of two distributions, $p, q \in \mathcal{D}^k$ with a two-dimensional histogram $h_{p,q}$, with probability at least $1 - e^{-n^\alpha}$ over the randomness in the selection of the sample, our algorithm returns a two-dimensional generalized histogram g_{LP} such that*

$$W(g_{LP}, h_{p,q}) \leq \frac{\gamma}{\sqrt{c}}.$$

Together with Fact 4, this immediately implies Theorem 1.13, which we restate for convenience: Theorem 1.13. *There exist absolute positive constants α, γ such that for any positive $\epsilon < 1$, there exists k_ϵ such that for any $k > k_\epsilon$, given a pair of samples of size $n = \frac{\gamma}{\epsilon^2}\frac{k}{\log k}$ drawn, respectively, from $p, q \in \mathcal{D}^k$, our estimator will output a number $\hat{d}$ such that with probability at least $1 - e^{-k^\alpha}$*

$$|\hat{d} - D_{tv}(p,q)| \leq \epsilon,$$

where $D_{tv}(p,q) = \sum_i \frac{1}{2}|p(i) - q(i)|$ is half the ℓ_1 distance between distributions p and q.

A.1 Proof of Theorem 5.6

We begin by formally describing our algorithm for recovering an estimate of the histogram corresponding to a pair of distributions. As in the one-distribution setting of Section 6, we state the algorithm in terms of three positive constants, $\mathcal{B}, C$, and $\mathcal{D}$, which can be defined arbitrarily provided the following inequalities hold:

$$\mathcal{B} > C > \mathcal{B}\left(\frac{1}{2} + \mathcal{D}\right) > \frac{\mathcal{B}}{2} > \mathcal{D} > 0 \quad \text{and} \quad 2\mathcal{B} + \mathcal{D} < 0.2.$$

ALGORITHM 3. ESTIMATE UNSEEN–TWO DISTRIBUTIONS

Input: Two-dimensional fingerprint $\mathcal{F}$, derived from two samples of size n, and an upper bound on the support sizes of the two distributions, k:
Output: Generalized two-dimensional histogram g_{LP}.

- Let $c_1 := \min\{i : i \in [n^{\mathcal{B}}, 2 \cdot n^{\mathcal{B}}] \text{ and } \sum_{j=i-n^C}^{i+2n^C} \sum_{\ell \geq 0} (j+\ell)\mathcal{F}(j,\ell) \leq nk^{1-\mathcal{B}+C}\}$.
- Let $c_2 := \min\{i : i \in [n^{\mathcal{B}}, 2 \cdot n^{\mathcal{B}}] \text{ and } \sum_{j=i-n^C}^{i+2n^C} \sum_{\ell \geq 0} (j+\ell)\mathcal{F}(\ell,j) \leq 6n^{1-\mathcal{B}+C}\}$.
- Let $v = (\ldots, v_{x_i,y_j}, \ldots)$ be the solution to Linear Program 5, on input $\mathcal{F}, c_1, c_2$, and k.
- Let g_{LP} be the generalized histogram formed by setting $g_{LP}(x_i, y_j) = v_{x_i,y_j}$ for all i, j, and then for all pairs i, j with either $i \geq c_1 + n^C$ or $j \geq c_2 + n^C$, incrementing $g_{LP}(\frac{i}{n}, \frac{j}{n})$ by $\mathcal{F}(i,j)$.

Linear Program 5.

Given a two-dimensional fingerprint $\mathcal{F}$, derived from two samples of size n, an upper bound on the support sizes of the two distributions, k, and two integers c_1, c_2:

- Define the sets
$$X := \left\{0, \frac{1}{nk}, \frac{2}{nk}, \ldots, \frac{c_1 + n^C/2}{n}\right\}, \text{ and } Y := \left\{0, \frac{1}{nk}, \frac{2}{nk}, \ldots, \frac{c_2 + n^C/2}{n}\right\}.$$
- For each pair $(x, y) \neq (0, 0)$ with $x \in X$ and $y \in Y$ define the associated LP variable $v_{x,y}$.

The linear program is defined as follows:

$$\text{Minimize} \sum_{i \in [c_1], j \in [c_2] : i+j \neq 0} \left| \mathcal{F}(i,j) - \sum_{x \in X, y \in Y} poi(nx, i) poi(ny, j) v_{x,y} \right|,$$

Subject to:

- $\sum_{x \in X, y \in Y} x \cdot v_{x,y} + \sum_{i=c_1+n^C}^{n} \sum_{j \geq 0} \frac{i}{n} \mathcal{F}(i,j) = 1$ (prob. mass = 1.)
- $\sum_{x \in X, y \in Y} y \cdot v_{x,y} + \sum_{j=c_2+n^C}^{n} \sum_{i \geq 0} \frac{j}{n} \mathcal{F}(i,j) = 1$ (prob. mass = 1.)
- $\sum_{x \in X, y \in Y} v_{x,y} \leq 2(n+k)$ (support size is not too big)
- $\forall x \in X, y \in Y, v_{x,y} \geq 0$ (histogram entries are non-negative)

The structure of the proof of Theorem 5.6 is very similar to that of its one-distribution analog, Theorem 1.11. The main difference is distance metrics—in the one-distribution setting, we used relative earthmover distance, and in this two-distribution setting we are using a histogram-moving metric. The second difference is that in the two-distribution setting we must be slightly more delicate in the intermediate region between the "frequently occurring" portion of the distribution (for which we simply use the empirical distribution of the samples), and the "infrequently occurring" region of the distribution for which we use the linear programming approach. In contrast to the one-distribution setting for which we fixed the location of this transition region obliviously, in the two-distribution setting, we choose the location of this transition region using the samples to guarantee that there is relatively little probability mass near this transition region. Finally, instead of using the one-dimensional Chebyshev bumps of Definition 6.7, we define two-dimensional analogs of those bumps, though we can reuse much of the same machinery and lemmas.

As was done in the one-distribution setting, we begin our proof by compartmentalizing the probabilistic component of our theorem by defining what it means for a pair of samples to be "faithful." We will then show that a pair of samples is "faithful" with high probability, and that our algorithm is successful whenever it is given a "faithful" pair of samples as input.

Definition A.7. A pair of samples of size n drawn, respectively, from distributions p, q with histogram $h = h_{p,q}$, with two-dimensional fingerprint $\mathcal{F}$, is said to be *faithful* if the following conditions hold:

- For all i, j,
$$\left| \mathcal{F}(i,j) - \sum_{x,y : h(x,y) \neq 0} h(x,y) \cdot poi(nx, i) poi(ny, j) \right| \leq n^{\frac{1}{2}+\mathcal{D}}.$$

- For all domain elements i, the number of times i occurs in the sample from p differs from its expectation of $n \cdot p(i)$ by at most
$$\max\left\{(n \cdot p(i))^{\frac{1}{2}+\mathcal{D}}, n^{\mathcal{B}(\frac{1}{2}+\mathcal{D})}\right\}.$$
Analogously for the number of times i occurs in the sample from q.
- Defining c_1, c_2 as in Algorithm 3,
$$\sum_{i \geq c_1+n^C \text{ or } j \geq c_2+n^C} \frac{i}{n}\mathcal{F}(i,j) + \sum_{x \leq \frac{c_1+n^C/2}{n}, y \leq \frac{c_2+n^C/2}{n}} x \cdot h(x,y) \leq 1 + n^{-\frac{1}{2}+\mathcal{D}}$$
and
$$\sum_{i \geq c_1+n^C \text{ or } j \geq c_2+n^C} \frac{j}{n}\mathcal{F}(i,j) + \sum_{x \leq \frac{c_1+n^C/2}{n}, y \leq \frac{c_2+n^C/2}{n}} y \cdot h(x,y) \leq 1 + n^{-\frac{1}{2}+\mathcal{D}}.$$
- Additionally,
$$\sum_{x \in [\frac{c_1-n^C}{n}, \frac{c_1+2n^C}{n}], y \geq 0} x \cdot h(x,y) \leq 13n^{-\mathcal{B}+C}$$
and
$$\sum_{x \geq 0, y \in [\frac{c_2-n^C}{n}, \frac{c_2+2n^C}{n}]} y \cdot h(x,y) \leq 13n^{-\mathcal{B}+C}.$$

The proof of the following lemma follows from basic tail bounds on Poisson random variables, and Chernoff bounds, and is nearly identical to that of Lemma 6.2.

Lemma A.8. *There is a constant $\gamma > 0$ such that for sufficiently large n, the probability that a pair of samples of size n consisting of independent draws from two distribution is "faithful" is at least $1 - e^{-n^\gamma}$.*

Proof. The proof that the first three conditions hold with the claimed probability is identical to the proof of the corresponding conditions in the one-distribution setting—Lemma 6.2—modulo an extra union bound over all possible choices of c_1, c_2. For the final condition, we show that it is implied by the first condition. For any $x \in [\frac{c_1-n^C}{n}, \frac{c_1+2n^C}{n}]$,
$$\mathrm{E}\left[I_{[c_1-n^C, c_1+2n^C]}(Poi(x))\right] \geq \frac{x}{2} - o(1/n),$$
where $I_{[c_1-n^C, c_1+2n^C]}(y)$ is the function that is equal to y if $y \in [c_1 - n^C, c_1 + 2n^C]$, and is 0 otherwise. Let $h(x)$ denote the histogram of the first distribution; assuming for the sake of contradiction that $\sum_{x \in [\frac{c_1-n^C}{n}, \frac{c_1+2n^C}{n}]} xh(x) > 13n^{C-\mathcal{B}}$, then $\sum_{i=c_1-n^C}^{c_1+2n^C} \mathrm{E}[\mathcal{F}_i] > \frac{13}{2}n^{1+C-\mathcal{B}} - o(1)$. On the other hand from the definition of c_1, $\sum_{i=c_1-n^C}^{c_1+2n^C} \mathcal{F}_i \leq 6n^{1+C-\mathcal{B}}$, yet the disparity between these $3n^C$ fingerprints and expected fingerprints, by the first condition of "faithful," is bounded by $3n^C n^{\frac{1}{2}+\mathcal{D}} = o(n^{1+C-\mathcal{B}})$, yielding the contradiction. The analogous statement holds for the second distribution.

Lemma A.9. *Given two distributions of support size at most k with histogram h, and a "faithful" pair of samples of size n drawn from each distribution with two-dimensional fingerprint $\mathcal{F}$, if c_1, c_2 are chosen as prescribed in Algorithm 3 then Linear Program 5 has a feasible point v' with objective value at most $O(n^{\frac{1}{2}+2\mathcal{B}+\mathcal{D}})$, and which is close to the true histogram h is the following sense:*

$$W(h, h_{v'}) \le O\left(n^{\mathcal{B}(-\frac{1}{2}+\mathcal{D})} + n^{-\mathcal{B}+C} + n^{-\frac{1}{2}+\mathcal{D}}\right) = O\left(\frac{1}{n^{\Omega(1)}}\right),$$

where $h_{v'}$ is the generalized histogram that would be returned by Algorithm 3 if v' were used in place of the solution to the linear program, v.

Proof. We explicitly define v' as a function of the true histogram h and fingerprint of the samples, $\mathcal{F}$, as follows:

- Define h' such that $h'(x, y) = h(x, y)$ for all x, y satisfying $x \le \frac{c_1+n^C/2}{n}$ and $y \le \frac{c_2+n^C/2}{n}$, and for all other x, y set $h'(x, y) = 0$, where c_1, c_2 are as defined in Algorithm 3.
- Initialize v' to be identically 0, and for each pair x, y with either $x \ge 1/nk$ or $y \ge 1/nk$ such that $h'(x, y) \ne 0$ increment $v'_{\bar{x},\bar{y}}$ by $h'(x, y)$, where $\bar{x}, \bar{y}$ are defined to be x, y rounded down to the closest elements of the set $Z = \{0, 1/nk, 2/nk, \ldots\}$.
- Let $m_1 := \sum_{x,y\in Z} x v'_{x,y} + m_{1,\mathcal{F}}$ and $m_2 := \sum_{x,y\in Z} y v'_{x,y} + m_{2,\mathcal{F}}$, where

$$m_{1,\mathcal{F}} := \sum_{i\ge c_1+n^C \text{ or } j\ge c_2+n^C} \frac{i}{n}\mathcal{F}(i, j) \text{ and } m_{2,\mathcal{F}} := \sum_{i\ge c_1+n^C \text{ or } j\ge c_2+n^C} \frac{j}{k}\mathcal{F}(i, j).$$

  If $m_1 > 1$, then decrease the probability mass in the first distribution by arbitrarily moving quantities of histogram from $v'_{x,y}$ to $v'_{0,y}$ until $m_1 = 1$; note that this does not alter the probability mass in the second distribution. If $m_2 > 1$, then perform the analogous operation. If $m_1 < 1$, then increase $v'_{x,0}$ by $(1 - m_1)/x$, where $x = \frac{c_1+n^C/2}{n}$. If $m_2 < 1$, then increase $v'_{0,y}$ by $(1 - m_2)/y$, where $y = \frac{c_2+n^C/2}{n}$.

To see that v' is a feasible point of the linear program, note that by construction, the first, second, and fourth conditions of the linear program are satisfied. The third condition of the linear program is satisfied, because each of the true distributions has support at most k, and, crudely, in the final step of the construction of v', we increment v' by less than $2n$— with one n corresponding to the increment we make for each of the two distributions.

We now consider the objective function value of v'. Note that $\sum_{j\le c_i} poi(c_i + n^C/2, j) = o(1/n)$, hence the fact that we are truncating $h(x, y)$ at probability $x \le \frac{c_1+n^C/2}{n}$ and $y \le \frac{c_2+n^C/2}{n}$ in the first step in our construction of v', has little effect on the "expected fingerprints" $\mathcal{F}(i, j)$ for $i \le c_1, j \le c_2$: specifically, for all such i, j,

$$\sum_{x,y:h(x,y)\ne 0} (h'(x, y) - h(x, y))\, poi(nx, i) poi(ny, j) = o(1).$$

Together with the first condition of the definition of faithful, by the triangle inequality, for each such i, j,

$$\left|\mathcal{F}(i, j) - \sum_{x,y:h'(x,y)\ne 0} h'(x, y) poi(kx, i) poi(ny, j)\right| \le n^{\frac{1}{2}+\mathcal{D}} + o(1).$$

We now bound the contribution of the discretization to the objective function value. As in the proof of Lemma 6.3, $|\frac{d}{dx} poi(nx, i)| \le n$, and hence we have

$$\left| \sum_{x,y:h'(x,y\neq 0)} h'(x,y)poi(nx,i)poi(ny,j) - \sum_{x,y\in X} v'_{x,y}poi(nx,i)poi(ny,j) \right| \leq 4k\frac{n}{kn},$$

where the factor of 4 arises, because the sum of the histogram entries is at most $2k$ (k for each of the two distributions), and hence discretizing the support in two stages, by first discretizing the x component and then discretizing the y component, each yields a contribution of at most $2n\frac{n}{kn}$.

In the final adjustment of mass in the creation of v', if any mass is added to v', then this added mass alters the objective function value by a negligible $o(1)$, again because $\sum_{j\leq c_i} poi(c_i + n^C/2, j) = o(1/n)$. In the case that mass must be removed, by the third condition of "faithful," and the fact that h' is generated from h by rounding the support down, which only decreases the amount of probability mass, the removal of this mass will decrease the expected fingerprints by at most $2n \cdot n^{-\frac{1}{2}+\mathcal{D}} = 2n^{\frac{1}{2}+\mathcal{D}}$. Thus, putting together the above pieces, the objective function value associated to v' is bounded by

$$c_1c_2\left(n^{\frac{1}{2}+\mathcal{D}} + 4 + o(1)\right) + 2n^{\frac{1}{2}+\mathcal{D}} \leq 5n^{\frac{1}{2}+2\mathcal{B}+\mathcal{D}},$$

for sufficiently large n.

We now turn to analyzing the *histogram distance* $W(h, h_{v'})$, where $h_{v'}$ is the generalized histogram obtained by appending the empirical fingerprint entries $\mathcal{F}(i,j)$ for $i \geq c_1 + n^C$ or $j \geq c_2 + n^C$ to v'. Our scheme for moving the histogram entries of $h_{v'}$ to yield h will have three stages. In the first stage, we consider the portion of $h_{v'}$ consisting of the empirical fingerprint—namely, $h_{v'}(\frac{i}{n}, \frac{j}{n})$, where either $i \geq c_1 + n^C$ or $j \geq c_2 + n^C$. In the second stage, we consider the portions corresponding to probability $x < \frac{c_1+n^C/2}{n}, y < \frac{c_2+n^C/2}{n}$, and in the third stage we consider the intermediate region (corresponding to the region of the fingerprint in which is relatively little probability mass, by the choice of c_1, c_2 and the final condition of "faithful").

For the first stage, for each domain element α contributing to histogram entry $h_{v'}(\frac{i}{n}, \frac{j}{n})$, with $i \geq c_1 + n^C$ or $j \geq c_2 + n^C$, we move one histogram entry in $h_{v'}$ from $(i/n, j/n)$ to location (x, y), where x, y are the true probabilities with which α occurs, respectively, in the two distributions. Let h' denote the histogram obtained after this movement. By the second condition of "faithful," the total histogram distance incurred by this process is bounded by assuming that all the weight in the histograms is at probability $n^{\mathcal{B}-1}$, and the discrepancy between the expected and actual number of occurrences of each domain element are maximal (given the second condition of "faithful"), namely $\frac{(n^{\mathcal{B}})^{\frac{1}{2}+\mathcal{D}}}{n}$. Thus the cost of this portion of the scheme is at most

$$2 \cdot \frac{n}{n^{\mathcal{B}}} \cdot \frac{2(n^{\mathcal{B}})^{\frac{1}{2}+\mathcal{D}}}{n} = 4n^{\mathcal{B}(-\frac{1}{2}+\mathcal{D})},$$

where the first factor of two is due to the two cases that either $i \geq c_1 + n^C$ or $j \geq c_2 + n^C$, the second factor of two is that for each domain element, we are considering the sum of discrepancy in the number of times it occurs in teach of the two distributions, and the factor of $\frac{n}{n^{\mathcal{B}}}$ is a bound on the number of such domain elements that can occur. Finally, note that after this phase of histogram moving, again by the second condition of "faithful," $h(x, y) = h'(x, y)$ for all x, y where either $x \geq \frac{c_1+2n^C}{k}$ or $y \geq \frac{c_2+2n^C}{n}$.

For the second stage of our histogram moving scheme, we transform h into g so the small histogram region with $x < \frac{c_1+n^C/2}{n}$ and $y < \frac{c_2+n^C/2}{n}$ of g and h' are identical. First, note that the rounding of the support of h to yield $h_{v'}$ has a cost per histogram entry of at most $\frac{1}{nk}$. There are at most $2k$ histogram entries, and thus the total cost, neglecting the extra mass that might be added or removed in the final step of constructing v', is at most $\frac{2}{n}$. By the third condition of "faithful," in

the final step of creating v' in which the total amount of mass is adjusted, at most $n^{-\frac{1}{2}+\mathcal{D}}$ units of mass will be removed from each distribution, which could contribute to the histogram distance an additional cost of at most $2n^{\frac{-1}{2}+\mathcal{D}}$; this is because the movement of q histogram units from location (x, y) to location $(0, y)$ decreases the probability mass by qx and also incurs this same amount of histogram distance cost, hence the removal of at most $n^{\frac{-1}{2}+\mathcal{D}}$ probability mass in each distribution augments the histogram distance by at most the claimed amount.

Thus after the first two histogram-moving stages, we have histograms h' and g such that $h'(x, y)$ and $g(x, y)$ are equal everywhere, except for (x, y) such that $x \leq \frac{c_1+2n^C}{n}$ and $y \leq \frac{c_2+2n^C}{n}$ and either $x \geq \frac{c_1+n^C/2}{n}$ or $y \geq \frac{c_2+n^C/2}{n}$. Now, we use the fact that there is relatively little histogram mass in this region; by our choice of c_1, c_2 and the final condition of "faithful", there are at most $(9+1)n^{1-2\mathcal{B}+C}$ histogram entries in either h' or g in this region, where the 9 is from the final condition of "faithful", and the 1 is a crude upper bound on the contribution from the adjustment in histogram mass in the final stage of the construction of $h_{v'}$. These entries can be moved to equalize the histogram entries in this region at a per-histogram entry cost of at most $4\frac{n^{\mathcal{B}}}{n}$, where the factor of 4 is because $x, y \leq 2n^{\mathcal{B}}$, and the cost is at most $x + y$, as these histogram entries, at worst, will be sent to $(0, 0)$. Hence the contribution towards the cost is at most $O(\frac{n^{\mathcal{B}}}{n} \cdot n^{1-2\mathcal{B}+C}) = O(n^{-\mathcal{B}+C})$. Summing up these bounds on the costs of the above three stages of a histogram-moving scheme yields the lemma. □

We now define the two-dimensional analog of the earthmoving schemes of Section 6.3. As we are working with a distance metric between two-dimensional generalized histograms that is in terms of the histogram entries, rather than the probability mass, our scheme will describe a manner of moving histogram entries. We repurpose much of the "Chebyshev bump" machinery of Section 6.3.

Definition A.10. For a given n, a *β-bump histogram-moving scheme* is defined by a sequence of pairs of positive real numbers $\{(r_1^i, r_2^i)\}$, the *bump centers*, and a sequence of corresponding functions $\{f_i\} : [0, 1]^2 \to \mathbb{R}$ such that $\sum_{i=0}^{\infty} f_i(x, y) = 1$ for all x, y, and each function f_i may be expressed as a linear combination of products of Poisson functions, $f_i(x, y) = \sum_{j,\ell=0}^{\infty} a_{ij\ell} poi(kx, j) poi(nx, \ell)$, such that $\sum_{j,\ell=0}^{\infty} |a_{ij\ell}| \leq \beta$.

Given a generalized histogram h, the scheme works as follows: for each x, y such that $h(x, y) \neq 0$, and each integer $i \geq 0$, move $h(x, y) \cdot f_i(x, y)$ histogram entries from (x, y) to the corresponding bump center (r_1^i, r_2^i). We denote the histogram resulting from this scheme by $(r, f)(h)$.

Definition A.11. A bump histogram-moving scheme (r, f) is $[\epsilon, k]$-*good* if for any generalized histogram h corresponding to a pair of distributions each of which has support size at most k, the histogram distance $W(h, (r, f)(h)) \leq \epsilon$.

The histogram-moving scheme we describe will use a rectangular mesh of bump centers, and thus it will prove convenient to index the bump centers, and corresponding functions via two subscripts. Thus a bump center will be denoted (r_1^{ij}, r_2^{ij}), and the corresponding function will be denoted f_{ij}.

Definition A.12. Let $s = 0.1 \log k$, and let $B_i(x)$ denote the (one dimensional) Chebyshev bumps of Definition 6.6, corresponding to $s = 0.1 \log n$ (as opposed to $0.2 \log n$ as in Definition 6.6). We define functions f_{ij} for $i, j \in [s-1] \cup \{0\}$, by

$$f_{ij}(x, y) = B_i(x) B_j(y).$$

Definition A.13. The *Chebyshev histogram-moving scheme* is defined in terms of n as follows: let $s = 0.1 \log n$. For $i \geq s$ or $j \geq s$, define the i, jth bump function $f_{ij}(x, y) = poi(nx, i) poi(ny, j)$

and associated bump center $(r_1^{ij}, r_2^{ij}) = (\frac{i}{n}, \frac{j}{n})$. For $i, j < s$ let $f_{i,j}(x, y) = B_i(x)B_j(y)$ and define their associated bump centers $(r_1^{ij}, r_2^{ij}) = (\frac{2s}{n}(1 - \cos(\frac{i\pi}{s})), \frac{2s}{n}(1 - \cos(\frac{j\pi}{s})))$. For ease of notation, let $r_i = \frac{2s}{n}(1 - \cos(\frac{i\pi}{s}))$, and hence for $i, j < s$ we have $(r_1^{ij}, r_2^{ij}) = (r_i, r_j)$.

The following lemma follows relatively easily from the corresponding lemmas in the one-dimensional setting (Lemmas 6.10 and 6.9), and shows that the above bump scheme is a $4n^{0.3}$-bump histogram-moving scheme.

Lemma A.14. *Each $f_{ij}(x, y)$ may be expressed as*

$$f_{ij}(x, y) = \sum_{\ell, m=0}^{\infty} a_{ij,\ell,m} poi(nx, \ell) poi(ky, m)$$

for coefficients satisfying $\sum_{\ell,m=0}^{\infty} |a_{ij,\ell,m}| \le 4n^{0.3}$. Additionally, for any x, y

$$\sum_{i,j \ge 0} f_{ij}(x, y) = 1.$$

Proof. To prove the first claim, recall that in the proof of Lemma 6.10, we showed that $B_i = \sum_{j=0}^{\infty} a_{ij} poi(nx, j)$ with $\sum_{j \ge 0} |a_{ij}| \le 2e^{\frac{3}{2}s}$. Thus in our setting, as $s = 0.1n$, we have that $\sum_{\ell,m=0}^{\infty} |a_{ij,\ell,m}| \le (2e^{\frac{3}{2}s})^2 = 4n^{0.3}$, as desired.

To prove the second claim, by Lemma 6.9, we have the following: For $i \ge s$, we have $\sum_{j \ge 0} f_{ij}(x, y) = poi(nx, i) \sum_{j \ge 0} poi(ny, j) = poi(nx, i)$. For $i < s$,

$$\begin{aligned}\sum_{j \ge 0} f_{ij}(x, y) &= \sum_{j < s} f_{ij}(x, y) + \sum_{j \ge s} f_{ij}(x, y) \\ &= \left(B_i(x) \sum_{j=0}^{s-1} poi(ny, j)\right) + \left(poi(nx, i) \sum_{j \ge s} poi(ny, j)\right).\end{aligned}$$

Summing the above expression over all i, and noting that $\sum_{i \ge 0} B_i(x) = 1$, and $\sum_{i \ge 0} poi(nx, i) = 1$, we conclude that

$$\sum_{i,j \ge 0} f_{ij}(x, y) = 1 \cdot \sum_{j < s} poi(nx, j) + 1 \cdot \sum_{j \ge s} poi(nx, j) = 1.$$

We now show that the scheme is $[O(\sqrt{\delta}), k]$-good, where $k = \delta n \log n$, and $\delta \ge \frac{1}{\log n}$. As in the one-distribution setting, the proof relies on the "skinnyness" of the Chebyshev bumps, as shown in Lemma 6.11, together with the bound on the support size.

Lemma A.15. *The Chebyshev histogram-moving scheme of Definition A.13 is $[O(\sqrt{\delta}), k]$-good, where $k = \delta n \log n$, and $\delta \ge \frac{1}{\log n}$.*

Proof. We begin by analyzing the contribution towards the cost of $h(x, y)$ for $x, y \le \frac{s}{n}$. Note that we can decompose the cost of moving the histogram entry at (x, y) to the bump centers (r_i, r_j) into the component due to the movement in each direction. For the skinny bumps, the per-histogram-entry cost of movement in the x direction is bounded by $\sum_{i=0}^{s-1} B_i(x)|x - r_i|$, which from Lemma 6.11 as employed in the proof of Lemma 6.12, is bounded by $O(\sqrt{\frac{x}{ns}})$. As $k = \delta n \log n$, and $\sum_{x,y} x \cdot h(x, y) = 1$, and $\sum_{x,y} h(x, y) \le 2k$, by the Cauchy-Schwarz inequality,

$$\sum_{x,y} \sqrt{x} h(x, y) = \sum_{x,y} \sqrt{x \cdot h(x, y)} \sqrt{h(x, y)} \le \sqrt{2k}$$

and hence the total cost of the skinny bumps is thus bounded by $O(\frac{\sqrt{k}}{\sqrt{ns}}) = O(\frac{1}{\sqrt{\delta}})$. For the wide bumps, the per-histogram entry cost is bounded by the following telescoping sum:

$$\sum_{i \geq s} poi(nx, i)\left(\left|\frac{i}{n} - x\right|\right) = \sum_{i \geq s} poi(nx, i)\frac{i}{n} - \sum_{i \geq s} poi(nx, i+1)\frac{i+1}{n} = poi(nx, s)\frac{s}{n}.$$

And hence the total cost is at most $\sup_{x \leq s/n}(\frac{1}{x} poi(nx, s)\frac{s}{n}) = O(1/\sqrt{s})$.

For (x, y) such that either $x > \frac{s}{n}$ or $y > \frac{s}{k}$, by the analysis of the skinny bumps above, the contribution to the cost from the skinny bumps is trivially seen to be $O(1/\sqrt{s})$. For the wider bumps, as above we have the following telescoping sum

$$\begin{aligned}\sum_{i \geq nx} poi(nx, i)\left(\left|\frac{i}{n} - x\right|\right) &= \sum_{i \geq nx}^{\infty} poi(nx, i)\frac{i}{n} - \sum_{i \geq nx}^{\infty} poi(nx, i+1)\frac{i+1}{n} \\ &= poi(nx, \lceil nx \rceil)\frac{\lceil nx \rceil}{n}.\end{aligned}$$

Similarly,

$$\sum_{i < nx} poi(nx, i)\left(\left|\frac{i}{n} - x\right|\right) = poi(nx, \lfloor nx \rfloor)\frac{\lfloor nx \rfloor}{n}.$$

Thus the cost of the wide bumps, per histogram entry, is at most $O(\sqrt{x/n})$. From our lower bounds on either x or y, the histogram entry at (x, y) can be at most n/s, and hence the total cost of this portion of the histogram moving scheme is at most $O(\frac{n}{s}\sqrt{s/n^2}) = O(1/\sqrt{s})$, as desired. □

We are now equipped to assemble the pieces and prove the performance guarantee of our ℓ_1 distance estimator. The proof mirrors that of Theorem 1.11; we leverage the fact that each Chebyshev bump can be expressed as a low-weight linear combination of Poisson functions, and hence, given two generalized histograms corresponding to feasible points of Linear Program 5 that have low objective function, after applying the Chebyshev histogram-moving scheme, the resulting generalized histograms will be extremely similar. Together with Lemma A.9 showing the existence of a feasible point that is close to the true histogram, all generalized histograms corresponding to solutions to the linear program (with low objective function) will be close to the true histogram, and in particular, will have similar ℓ_1 distance.

Proof of Theorem 5.6. Let h denote the histogram of the pair of distributions from which the samples were drawn. Let g_1 denote the generalized histogram whose existence is guaranteed by Lemma A.9, satisfying $W(g_1, h) \leq n^{-\Omega(1)}$, corresponding to a feasible point of the linear program with objective function at most $\alpha \leq O(n^{\frac{1}{2}+2\mathcal{B}+\mathcal{D}})$. Let g_2 denote the generalized histogram output by Algorithm 3, and hence corresponds to a solution to the linear program with objective function at most α. Let g_1', g_2' denote the generalized histograms that result from applying the Chebyshev histogram-moving scheme of Definition A.13 to g_1 and g_2, respectively. By Lemma A.15, $W(g_i, g_i') = O(\sqrt{\delta})$. We now show that $W(g_1', g_2') = O(n^{-\mathcal{B}+C})$. Given this, the triangle inequality yields that

$$W(h, g_2) \leq W(h, g_1) + W(g_1, g_1') + W(g_1', g_2') + W(g_2', g_2) \leq O(\sqrt{\delta}).$$

The analysis of $W(g_1', g_2')$ is nearly identical to the analogous component of Theorem 1.11: we argue that for all pairs i, j except those in the intermediate zone defined by $i \in [c_1, c_1 + n^C]$ or $j \in [c_2, c_2 + n^C]$ and both $i < c_1 + n^C$ and $j < c_2 + n^C$, in the case that $g_1(r_1^{ij}, r_2^{ij}) > g_2(r_1^{ij}, r_2^{ij})$ we can move this discrepancy $g_1(r_1^{ij}, r_2^{ij}) - g_2(r_1^{ij}, r_2^{ij})$ from $g_1(r_1^{ij}, r_2^{ij})$ to location $(0, 0)$ incurring little histogram distance; analogously in the case that $g_1(r_1^{ij}, r_2^{ij}) < g_2(r_1^{ij}, r_2^{ij})$. After this histogram-moving scheme is implemented, we conclude by noting that the total number of histogram entries

in this intermediate zone is relatively small, because of our choice of c_1, c_2 and our bounds on the objective function value associated to g_1, g_2, and thus the discrepancy in this region can also be moved to $(0,0)$ at small histogram–moving cost, thus bounding $W(g_1', g_2')$.

We now quantify the cost of this approach—the analysis closely mirrors that of the one-distribution setting. As in the one-dimensional setting, for each of the "skinny" Chebyshev bumps with centers (r_i, r_j), $|g_1'(r_i, r_j) - g_2'(r_i, r_j)| \leq O(\alpha n^{0.3})$, and hence the cost of equalizing the discrepancy for all s^2 such pairs of centers is bounded by $O(\alpha n^{0.3} s^2 \frac{s}{n})$, where the final factor of $\frac{s}{n}$ is because the per-histogram-entry histogram-moving cost of moving from (x, y) to $(0,0)$ is $x + y = O(\frac{s}{n})$.

Similarly, the contribution from bump centers (r_1^{ij}, r_2^{ij}) with $i \leq c_1$, and $j \leq c_2$, not including the already counted bumps with $i, j \leq s$ is bounded by $O(c_1 c_2 \alpha \frac{c_1+c_2}{n}) = O(n^{3\mathcal{B}-1}\alpha)$. The contribution from (r_1^{ij}, r_2^{ij}) for either $i \geq c_1 + n^C$ or $j \geq c_2 + n^C$ is $o(1/n)$ as g_1 and g_2 are identical in this region, and the $o(1/n)$ is due to the contribution to the discrepancy in g_1', g_2' in this region from the discrepancy between $g_1(x, y)$ and $g_2(x, y)$ for $x \leq \frac{c_1+n^{C/2}}{n}, y \leq \frac{c_2+n^{C/2}}{n}$, bounded via Poisson tail bounds.

To conclude, we bound the contribution from the intermediate zone corresponding to bump centers (r_1^{ij}, r_2^{ij}) with $i \in [c_1, c_1 + n^C]$ and $j \leq c_2 + n^C$, or with $j \in [c_2, c_2 + n^C]$ and $i \leq c_1 + n^C$. To show this, we will argue that g_1 and g_2 can have at most $O(n^{-\mathcal{B}+C})$ probability mass in this intermediate zone. We prove this by recalling that c_1 and c_2 were chosen so the probability mass in the empirical distribution of the fingerprint in a slightly larger region containing this intermediate zone, is small. Thus if g_i had too much mass in this region, it means that g_i has too little mass in the low-probability region (as the high probability region is fixed to be the empirical distribution of the fingerprint, and the total mass in each distribution is fixed to be 1 by the linear program). We then argue that having too little mass in the low-probability region would induce a large objective function value, contradicting the bounds on the objective values of g_i (from Lemma A.9). Thus we will conclude that g_i has little mass in this region, and hence g_i' will have little mass in a (slightly smaller) corresponding region.

We give the proof in the case that $i \in [c_1, c_1 + n^C]$; the proof in the other case is analogous, with the roles of i and j swapped. Since all but $o(1/n)$ of the mass in this interval in g_i' comes from the slightly larger region $x \in [\frac{c_1-n^{C/2}}{n}, \frac{c_1+n^C}{n}]$, $y \leq \frac{c_2+n^C}{n}$ of g_i, we will bound the total mass that can be in this region of g_i.

Assume for the sake of contradiction that

$$\sum_{x \in [\frac{c_1-n^{C/2}}{n}, \frac{c_1+n^C}{n}], y \geq 0} x \cdot g_1(x, y) > 7n^{-\mathcal{B}+C}.$$

From the definition of c_1 and the fact that we are drawing samples of size n from each distribution,

$$\sum_{i \in [c_1-n^C, c_1+n^C], j \geq 0} \frac{i}{n} \mathcal{F}(i, j) \leq 6n^{-\mathcal{B}+C}.$$

Since $\mathcal{F}$ scaled by $1/n$ agrees with $g_1(x, y)$ for $x \geq \frac{c_1+n^C}{n}$, and both $\mathcal{F}$ scaled by $1/n$ and g_1 have total probability mass of 1 (in each component), it follows from the above two inequalities that

$$\sum_{i \leq c_1-n^C, j \geq 0} \frac{i}{n} \mathcal{F}(i, j) - \sum_{x < \frac{c_1-n^{C/2}}{n}, y \geq 0} x \cdot g_1(x, y) \geq n^{-\mathcal{B}+C}.$$

From which it immediately follows that

$$\sum_{i\le c_1-n^C, j\ge 0} i\mathcal{F}(i,j) - n\sum_{i\le c_1-n^C-1}\sum_{x<\frac{c_1-n^C/2}{n}, y\ge 0} x\cdot poi(nx,i)\cdot g_1(x,y) \ge n^{1-\mathcal{B}+C}.$$

And hence, since $x\cdot poi(xn,i) = poi(xn,i+1)\frac{i+1}{n}$, we have that

$$\sum_{i\le c_1-n^C, j\ge 0} i\mathcal{F}(i,j) - \sum_{i\le c_1-n^C}\sum_{x<\frac{c_1-n^C/2}{n}, y\ge 0} i\cdot poi(nx,i)\cdot g_1(x,y) \ge n^{1-\mathcal{B}+C}.$$

Poisson tail bounds yield that we can extend the sum over x to cover all $x\ge 0$, adding a $o(1)$ term. Hence

$$\begin{aligned} n^{1-\mathcal{B}+C} - o(1) &\le \sum_{i\le c_1-n^C, j\ge 0} i\mathcal{F}(i,j) - \sum_{i\le c_1-n^C}\sum_{x,y\ge 0} i\cdot poi(nx,i)\cdot g_1(x,y) \\ &\le \sum_{i\le c_1-n^C, j\ge 0} i\left|\mathcal{F}(i,j) - \sum_{x,y\ge 0} poi(nx,i)\cdot g_1(x,y)\right|. \end{aligned}$$

Since $i = O(n^{\mathcal{B}})$, we can replace i in the above expression by this value, yielding that the linear program objective function value corresponding to g_1 would be at least $O(n^{1-\mathcal{B}+C}/n^{\mathcal{B}}) = O(n^{1-2\mathcal{B}+C})$, contradicting the fact that the objective function value is bounded by $O(n^{\frac{1}{2}+2\mathcal{B}+\mathcal{D}})$ (by definition of g_1 and Lemma A.9). An identical argument applies to g_2, hence this intermediate region has at most $7n^{-\mathcal{B}+C}$) units of mass, in either g_1 or g_2, and thus the discrepancy between g_1' and g_2' in the (smaller) intermediate region with $x\in[\frac{c_1}{n}, \frac{c_1+n^C}{n}]$, or $y\in[\frac{c_2}{n}, \frac{c_2+n^C}{n}]$, is bounded by $O(n^{-\mathcal{B}+C})$. Each unit of this mass can give rise to at most $\frac{n}{n^{\mathcal{B}}}$ histogram entries, since $c_i\ge n^{\mathcal{B}}$. Additionally, each these histogram entries can be moved to $(0,0)$ at a cost of at most $c_1+c_2 = O(\frac{n^{\mathcal{B}}}{n})$. Thus the contribution to the histogram distance of this intermediate region is bounded by

$$O\left(n^{-\mathcal{B}+C}\frac{n}{n^{\mathcal{B}}}\frac{n^{\mathcal{B}}}{n}\right) = O(n^{-\mathcal{B}+C}).$$

Thus we conclude that

$$\begin{aligned} W(g_1',g_2') &= O\left(\alpha n^{0.3}\frac{\log^3 n}{n} + n^{3\mathcal{B}-1}\alpha + n^{-\mathcal{B}+C}\right), \\ &= O\left(\frac{n^{0.8+2\mathcal{B}+\mathcal{D}}\log^3 n}{n} + \frac{n^{\frac{1}{2}+5\mathcal{B}+\mathcal{D}}}{n} + n^{-\mathcal{B}+C}\right), \\ &= O(n^{-\gamma}), \end{aligned}$$

for some constant $\gamma>0$. Hence

$$W(h,g_2)\le W(h,g_1)+W(g_1,g_1')+W(g_1',g_2')+W(g_2',g_2) = O(\sqrt{\delta}),$$

as the $W(g_i,g_i')$ terms in the above expression are the dominant terms.

B ROBUSTNESS TO MODIFYING PARAMETERS

In this section, we give strong empirical evidence for the robustness of our approach. Specifically, we show that the performance of our estimator remains essentially unchanged over large ranges of the two parameters of our estimator: The choice of mesh points of the interval $(0,1]$ which

correspond to the variables of the linear programs, and the parameter α of the second linear program that dictates the additional allowable discrepancy between the expected fingerprints of the returned histogram and the observed fingerprints.

Additionally, we also consider the variant of the second linear program that is based on a slightly different interpretation of Occam's Razor: Instead of minimizing the support size of the returned histogram, we now minimize the *entropy* of the returned histogram. Note that this is still a *linear* objective function, and hence can still be solved by a linear program. Formally, recall that the linear programs have variables $h'_1, \ldots, h'_\ell$ corresponding to the histogram values at corresponding fixed grid points $x_1, \ldots, x_\ell$. Rather than having the second linear program minimize $\sum_{j=1}^{\ell} h'_j$, we consider replacing the objective function by

$$\text{Minimize: } \sum_{j=1}^{\ell} h'_j \cdot \log \frac{1}{x_j}.$$

Note that the quantity $\sum_{j=1}^{\ell} h'_j \cdot \log \frac{1}{x_j}$ is precisely the entropy corresponding to the histogram defined by $h(x_i) = h'_i$ and $h(x) = 0$ for all $x \notin \{x_1, \ldots, x_\ell\}$. Additionally, this expression is still a linear function (of the variables h'_j), and hence we still have a linear program.

Figure 5 depicts the performance of our estimator with five different sets of parameters, as well as the performance of the estimator with the entropy minimization objective, as described in the previous paragraph.

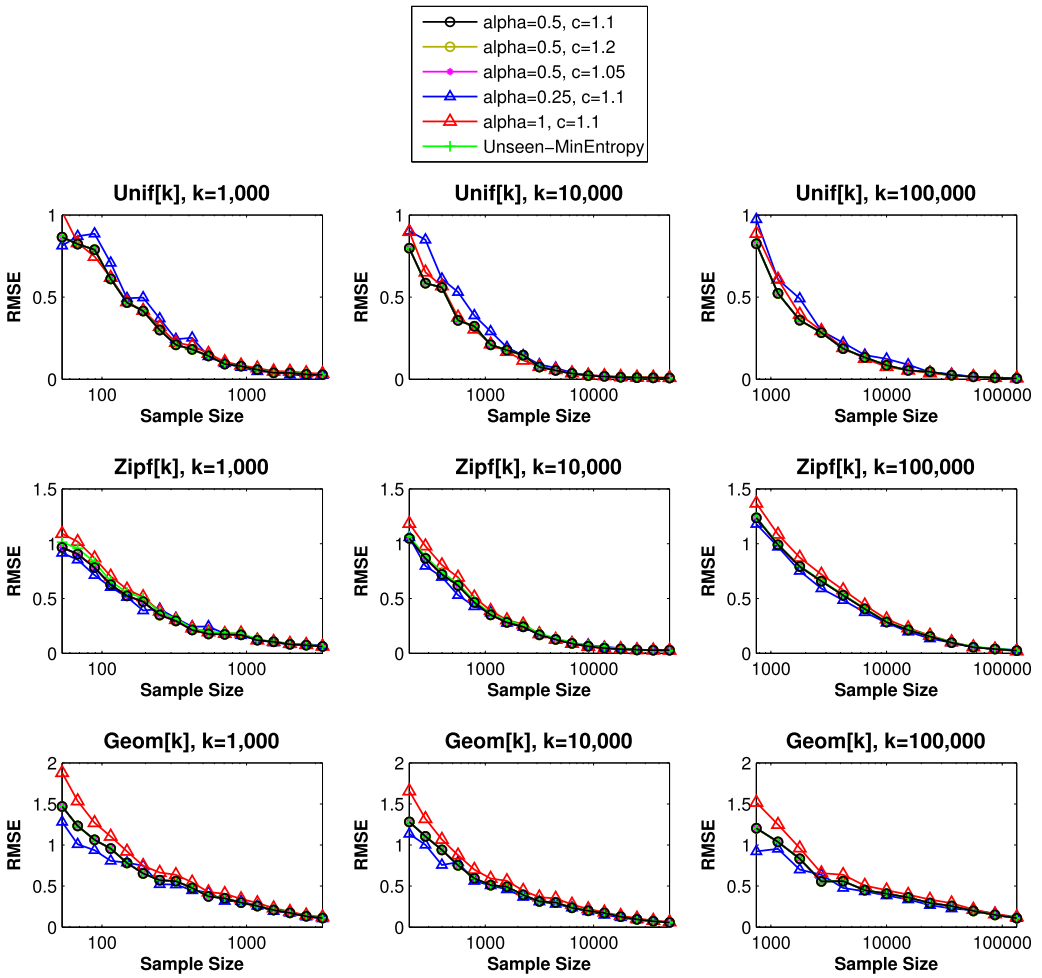

Fig. 5. Plots depicting the RMSE of each entropy estimator over 100 trials, plotted as a function of the sample size. The samples are drawn from a uniform distribution $Unif[k]$ (top row), a Zipf distribution $Zipf[k]$ (middle row), and a geometric distribution $Geom[k]$ (bottom row), for $k = 1{,}000$ (left column), $k = 10{,}000$ (middle column), and $k = 100{,}000$ (right column). The unseen estimator with parameters α, c corresponds to setting the error parameter α of Algorithm 1 and the mesh corresponding to the linear program variables to be a geometrically spaced grid with geometric ratio c; namely, $X = \{\frac{1}{n^2}, \frac{c}{n^2}, \frac{c^2}{n^2}, \frac{c^3}{n^2}, \ldots, \}$, where n is the sample size. Note that the performance of the different variants of the *unseen* estimator perform nearly identically. In particular, the performance is essentially unchanged if one makes the granularity of the grid spacing of the mesh of probabilities used in the linear programs more fine, or slightly more coarse. The performance is also essentially identical if one changes the objective function of Linear Program 2 to minimize the entropy of the returned histogram ("Unseen-MinEntropy" in the above plot), rather than minimizing the support size. The performance varies slightly when the error parameter α is changed, though is reasonably robust to increasing or decreasing α by factors of up to 2.

C MATLAB CODE

Below is our Matlab implementation of Algorithm 1. Our implementation uses the *linprog* command for solving the linear programs, which requires Matlab's Optimization toolkit. This code is also available from our websites.

```
function [histx,x] = unseen(f)
% Input: fingerprint f, where f(i) represents number of elements that
% appear i times in a sample.  Thus sum_i i*f(i) = sample size.
% File makeFinger.m transforms a sample into the associated fingerprint.
%
% Output: approximation of 'histogram' of true distribution.  Specifically,
% histx(i) represents the number of domain elements that occur with
% probability x(i).   Thus sum_i x(i)*histx(i) = 1, as distributions have
% total probability mass 1.
%
% An approximation of the entropy of the true distribution can be computed
% as:    Entropy = (-1)*sum(histx.*x.*log(x))

f=f(:)';
k=f*(1:size(f,2))';  %total sample size


%%%%%%% algorithm parameters %%%%%%%%%%
gridFactor = 1.1;     % the grid of probabilities will be geometric, with ...
    this ratio.
% setting this smaller may slightly increase accuracy, at the cost of speed
alpha = .5; %the allowable discrepancy between the returned solution and ...
    the "best" (overfit).
% 0.5 worked well in all examples we tried, though the results were nearly ...
    indistinguishable
% for any alpha between 0.25 and 1.  Decreasing alpha increases the ...
    chances of overfitting.
xLPmin = 1/(k*max(10,k));
min_i=min(find(f>0));
if min_i > 1
    xLPmin = min_i/k;
end% minimum allowable probability.
% a more aggressive bound like 1/k^1.5 would make the LP slightly faster,
% though at the cost of accuracy
maxLPIters = 1000;    % the 'MaxIter' parameter for Matlab's 'linprog' LP ...
    solver.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


% Split the fingerprint into the 'dense' portion for which we
% solve an LP to yield the corresponding histogram, and 'sparse'
% portion for which we simply use the empirical histogram
x=0;
histx = 0;
fLP = zeros(1,max(size(f)));
for i=1:max(size(f))
    if f(i)>0
        wind = [max(1,i-ceil(sqrt(i))),min(i+ceil(sqrt(i)),max(size(f)))];
        if sum(f(wind(1):wind(2)))<sqrt(i)% 2*sqrt(i)
            x=[x, i/k];
```

```
            histx=[histx,f(i)];
            fLP(i)=0;
        else
            fLP(i)=f(i);
        end
    end
end

% If no LP portion, return the empirical histogram
fmax = max(find(fLP>0));
if min(size(fmax))==0
    x=x(2:end);
    histx=histx(2:end);
    return;
end

% Set up the first LP
LPmass = 1 - x*histx'; %amount of probability mass in the LP region

fLP=[fLP(1:fmax), zeros(1,ceil(sqrt(fmax)))];
szLPf=max(size(fLP));

xLPmax = fmax/k;
xLP=xLPmin*gridFactor.^(0:ceil(log(xLPmax/xLPmin)/log(gridFactor)));
szLPx=max(size(xLP));

objf=zeros(szLPx+2*szLPf,1);
objf(szLPx+1:2:end)=1./(sqrt(fLP+1));  % discrepancy in ith fingerprint ...
    expectation
objf(szLPx+2:2:end)=1./(sqrt(fLP+1));  % weighted by 1/sqrt(f(i) + 1)

A = zeros(2*szLPf,szLPx+2*szLPf);
b=zeros(2*szLPf,1);
for i=1:szLPf
    A(2*i-1,1:szLPx)=poisspdf(i,k*xLP);
    A(2*i,1:szLPx)=(-1)*A(2*i-1,1:szLPx);
    A(2*i-1,szLPx+2*i-1)=-1;
    A(2*i,szLPx+2*i)=-1;
    b(2*i-1)=fLP(i);
    b(2*i)=-fLP(i);
end

Aeq = zeros(1,szLPx+2*szLPf);
Aeq(1:szLPx)=xLP;
beq = LPmass;


options = optimset('MaxIter', maxLPIters,'Display','off');
for i=1:szLPx
    A(:,i)=A(:,i)/xLP(i);   %rescaling for better conditioning
    Aeq(i)=Aeq(i)/xLP(i);
end
[sol, fval, exitflag, output] = linprog(objf, A, b, Aeq, beq, ...
    zeros(szLPx+2*szLPf,1), Inf*ones(szLPx+2*szLPf,1),[], options);
if exitflag==0
        'maximum number of iterations reached--try increasing maxLPIters'
end
```

```
if exitflag<0
    'LP1 solution was not found, still solving LP2 anyway...'
    exitflag
end

% Solve the 2nd LP, which minimizes support size subject to incurring at most
% alpha worse objective function value (of the objective function in the
% previous LP).
objf2=0*objf;
objf2(1:szLPx) = 1;
A2=[A;objf'];          % ensure at most alpha worse obj value
b2=[b; fval+alpha];    % than solution of previous LP
for i=1:szLPx
    objf2(i)=objf2(i)/xLP(i);    %rescaling for better conditioning
end
[sol2, fval2, exitflag2, output] = linprog(objf2, A2, b2, Aeq, beq, ...
    zeros(szLPx+2*szLPf,1), Inf*ones(szLPx+2*szLPf,1),[], options);

if not(exitflag2==1)
    'LP2 solution was not found'
    exitflag2
end


%append LP solution to empirical portion of histogram
sol2(1:szLPx)=sol2(1:szLPx)./xLP';    %removing the scaling
x=[x,xLP];
histx=[histx,sol2'];
[x,ind]=sort(x);
histx=histx(ind);
ind = find(histx>0);
x=x(ind);
histx=histx(ind);
```

```
function f=makeFinger(v)

% Input:  vector of integers, v
% Output: vector of fingerprints, f where f(i) = |{j: |{k:v(k)=j}|=i }|
%         i.e. f(i) is the number of elements that occur exactly i times
%         in the vector v

h1 = hist(v,min(v):max(v));
f=hist(h1,0:max(h1));
f=f(2:end);
f=f(:);
```

Example of how to invoke the unseen" estimator:

```
% Generate a sample of size 10,000 from the uniform distribution of ...
    support 100,000
n=200; k=10000;
samp = randi(n,k,1);

% Compute corresponding 'fingerprint'
f = makeFinger(samp);


% Estimate distribution from which sample was drawn
[h,x,en]=entropy_est(f);


%output entropy of the true distribution, Unif[n]
trueEntropy = log(n)

%output entropy of the empirical distribution of the sample
empiricalEntropy = -f'*(((1:max(size(f)))/k).*log(((1:max(size(f)))/k)))' ...
    + sum(f)/(2*k)

%output entropy of the recovered histogram, [h,x]
estimatedEntropy = -h*(x.*log(x))'




%output support size (# species) of the recovered distribution
suppSz = sum(h)
```