Instance Optimal Learning

Gregory Valiant∗
Stanford University
valiant@stanford.edu

Paul Valiant†
Brown University
pvaliant@gmail.com

April 22, 2015

Abstract

We consider the following basic learning task: given independent draws from an unknown distribution over a discrete support, output an approximation of the distribution that is as accurate as possible in $\ell_1$ distance (equivalently, total variation distance, or “statistical distance”). Perhaps surprisingly, it is often possible to “de-noise” the empirical distribution of the samples to return an approximation of the true distribution that is significantly more accurate than the empirical distribution, without relying on any prior assumptions on the distribution. We present an instance optimal learning algorithm which, up to an additive sub-constant factor, optimally performs this de-noising for every distribution for which such a de-noising is possible. More formally, given $n$ independent draws from a distribution $p$, our algorithm returns a labelled vector whose expected distance from $p$ is equal to the minimum possible expected error that could be obtained by any algorithm that knows the true unlabeled vector of probabilities of distribution $p$ and simply needs to assign labels, up to an additive subconstant term that is independent of $p$ and depends only on the number of samples, $n$. This somewhat surprising result has several conceptual implications, including the fact that, for any large sample, Bayesian assumptions on the “shape” or bounds on the tail probabilities of a distribution over discrete support are not helpful for the task of learning the distribution.

1 Introduction

Given independent draws from an unknown distribution over an unknown discrete support, what is the best way to aggregate those samples into an approximation of the true distribution? This is, perhaps, the most fundamental learning problem. The most obvious and most widely employed approach is to simply output the empirical distribution of the sample. To what extent can one improve over this naive approach? To what extent can one “de-noise” the empirical distribution, without relying on any assumptions on the structure of the underlying distribution?

Perhaps surprisingly, there are many settings in which de-noising can be done without a priori assumptions on the distribution. We begin by presenting two motivating examples illustrating rather different settings in which significant de-noising of the empirical distribution is possible.

Example 1. Suppose you are given 100,000 independent draws from some unknown distribution, and you find that there are roughly 1,000 distinct elements, each of which appears roughly 100

∗This work is supported in part by NSF CAREER Award CCF-1351108.
†This work is supported in part by a Sloan Research Fellowship.
times. Furthermore, suppose you compute the variance in the number of times the different domain
elements occur, and it is close to 100. Based on these samples, you can confidently deduce that the
true distribution is very close to a uniform distribution over 1,000 domain elements, and that the
true probability of a domain element seen 90 times is roughly the same as that of an element observed
110 times. The basic reasoning is as follows: if the true distribution were the uniform distribution,
then the noise from the random sampling would exhibit the observed variance in the number of
occurrences; if there was any significant variation in the true probabilities of the different domain
elements, then, combined with the noise added via the random sampling, the observed variance
would be significantly larger than 100. The $\ell_1$ error of the empirical distribution would be roughly
0.1, whereas the “de-noised” distribution would have error less than 0.01.

**Example 2.** Suppose you are given 1,000 independent draws from an unknown distribution, and all
1000 samples are unique domain elements. You can safely conclude that the combined probability of
all the observed domain elements is likely to be much less than 1/100—if this were not the case, one
would expect at least one of the observed elements to occur twice in the sample. Hence the empirical
distribution of the samples is likely to have $\ell_1$ distance nearly 2 from the true distribution, whereas
this reasoning would suggest that one should return the zero vector, which would have $\ell_1$ distance
at most 1.

In both of the above examples, the key to the “de-noising” was the realization that the true
distributions possessed some structure—structure that was both easily deduced from the samples,
and structure that, once known, could then be leveraged to de-noise the empirical distribution.
Our main result is an algorithm which de-noises the empirical distribution as much as is possible,
whenever such denoising is possible. Specifically, our algorithm achieves, up to a subconstant term,
the best error that any algorithm could achieve—even an algorithm that is given the unlabeled
vector of true probabilities and simply needs to correctly label the probabilities.

**Theorem 1.** There is a function $\text{err}(n)$ that goes to zero as $n$ gets large, and an algorithm, which
given $n$ independent draws from any distribution $p$ of discrete support, outputs a labelled vector $q$, such that

$$E[||p - q||_1] \leq \text{opt}(p, n) + \text{err}(n),$$

where $\text{opt}(p, n)$ is the minimum expected error that any algorithm could achieve on the following
learning task: given $p$, and given $n$ samples drawn independently from a distribution that is identical
to $p$ up to an arbitrary relabeling of the domain elements, learn the distribution.

The performance guarantees of the above algorithm can be equivalently stated as follows: let
$S \leftarrow p$ denote that $S$ is a set of $n$ independent draws from distribution $p$, and let $\pi(p)$ denote
a distribution that is identical to $p$, up to relabeling the domain elements according a labeling
scheme $\pi$ chosen from a sufficiently large support. Our algorithm, which maps a set of samples $S$
to a labelled vector $q = f(S)$, satisfies the following: For any distribution $p$,

$$E_{S \leftarrow p}[||p - q||_1] \leq \min_{\text{alg}s} \max_{\pi} \left( E_{S \leftarrow \pi(p)}[\pi(p) - A(S)] \right) + o_n(1),$$

where $o_n(1) \to 0$ as $n \to \infty$ is independent of $p$ and depends only on $n$.

One surprising implication of the above result is that, for large samples, prior knowledge of
the “shape” of the distribution, or knowledge of the rate of decay of the tails of the distribution,
cannot improve the accuracy of the learning task. For example, typical Bayesian assumptions that the frequency of words in natural language satisfy Zipf distributions, or the frequencies of different species of bacteria in the human gut satisfy Gamma distributions or various power-law distributions, etc, can improve the expected error of the learned distribution by at most subconstant factors.

The key intuition behind this optimal de-noising, and the core of our algorithm, is the ability to very accurately approximate the unlabeled vector of probabilities of the true distribution, given access to independent samples. In some sense, our result can be interpreted as the following statement: up to an additive subconstant factor, one can always recover an approximation of the unlabeled vector of probabilities more accurately than one can disambiguate and label such a vector. That is, if one has enough samples to accurately label the unlabeled vector of probabilities, then one also has more than enough samples to accurately learn that unlabeled vector. Of course, this statement can only hold up to some additive error term, as the following example illustrates.

**Example 3.** Given samples drawn from a distribution supported on two unknown domain elements, if one is told that both probabilities are exactly 1/2, then as soon as one observes both domain elements, one knows the distribution exactly, and thus the expected error given n samples will be $O(1/2^n)$ as this bounds the probability that one of the two domain elements is not observed in a set of n samples. Without the prior knowledge that the two probabilities are 1/2, the best algorithm will have expected error $\approx 1/\sqrt{n}$.

The above example illustrates that prior knowledge of the vector of probabilities can be helpful. Our result, however, shows that this phenomena only occurs to a significant extent for very small sample sizes; for larger samples, no distribution exists for which prior knowledge of the vector of probabilities improves the expected error of a learning algorithm beyond a universal subconstant additive term that goes to zero as a function of the sample size.

Our algorithm proceeds via two steps. In the first step, the samples are used to output an approximation of the vector of true probabilities. We show that, with high probability over the randomness of the n independent draws from the distribution, we accurately recover the portion of the vector of true probabilities consisting of values asymptotically larger than $1/n \log n$. The following proposition formally quantifies this initial step:

**Proposition 1.** There exists an algorithm such that, for any function $f(n) = \omega(n)$ that goes to infinity as n gets large (e.g. $f(n) = \log \log n$), there is a function $\omega(n)$ of n that goes to zero as n gets large, such that given n samples drawn independently from any distribution p, the algorithm outputs an unlabeled vector, q, such that, with probability $1 - e^{-n^{O(1)}}$, there exists a labeling $\pi(q)$ of the vector q such that

$$\sum_i \left| \max \left( \frac{p(x)}{n \log n}, \frac{f(n)}{n \log n} \right) - \max \left( \pi(q)(x), \frac{f(n)}{n \log n} \right) \right| < o(n),$$

where $p(x)$ denotes the true probability of domain element x in distribution p.

The power of the above proposition lies in the following trivial observation: for any function $g(n) = o(1/n)$, the domain elements x that both occur in the n samples and have true probability $p(x) < g(n)$, can account for at most $o(1)$ probability mass, in aggregate. Hence the fact that Proposition 1 only guarantees that we are learning the probabilities above $1/n \log n = o(1/n)$ gives rise to, at most, an $\ell_1$ error of $o(1)$ in our final returned vector.
The second step of our algorithm leverages the accurate approximation of the unlabeled vector of probabilities to optimally assign probability values to each of the observed domain elements. This step of the algorithm can be interpreted as solving the following optimization problem: given \( n \) independent draws from a distribution, and an unlabeled vector \( v \) representing the true vector of probabilities of distribution \( p \), for each observed domain element \( x \), assign the probability \( q(x) \) that minimizes the expected \( \ell_1 \) distance \( |q(x) - p(x)| \). This optimization task is well-defined, though computationally intractable. Nevertheless, we show that a very natural and computationally tractable scheme, which assigns a probability \( q(x) \) that is a function of only \( v \) and the number of occurrences of \( x \), incurs an expected error within \( o(1) \) of the expected error of the optimal scheme (which assigns \( q(x) \) as a function of \( v \) and the entire set of samples).

Beyond yielding a near optimal learning algorithm, there are several additional benefits to our approach of first accurately reconstructing the unlabeled vector of probabilities. For instance, such an unlabeled vector allows us to estimate properties of the underlying distribution including estimating the error of our returned vector, and estimating the error in our estimate of each observed domain element’s probability.

1.1 Related Work

Perhaps the first work on correcting the empirical distribution—which serves as the jumping-off point for nearly all of the subsequent work on this problem that we are aware of—is the work of Turing, and I.J. Good [18] (see also [19]). In the context of their work at Bletchley Park as part of the British WWII effort to crack the German enigma machine ciphers, Turing and Good developed a simple estimator that corrected the empirical distribution, in some sense to capture the “missing” probability mass of the distribution. This estimator and its variants have been employed widely, particularly in the context of natural language processing and other settings in which significant portions of the distribution are comprised of domain elements with small probabilities (e.g. [12]). In its most simple form, the Good-Turing frequency estimation scheme estimates the total probability of all domain elements that appear exactly \( i \) times in a set of \( n \) samples as \( \left(\frac{i+1}{i+2}\right)\frac{F_{i+1}}{n} \), where \( F_j \) is the total number of species that occur exactly \( j \) times in the samples. The total probability mass consisting of domain elements that are not seen in the samples—the “missing” mass, or, equivalently, the probability that the next sample drawn will be a new domain element that has not been seen previously—can be estimated via this formula as \( F_1/n \), namely the fraction of the samples consisting of domain elements seen exactly once.

The Good–Turing estimate is especially suited to estimating the total mass of elements that appear few times; for more frequently occurring domain elements, this estimate has high variance—for example, if \( F_{i+1} = 0 \), as will be the case for most large \( i \), then the estimate is 0. However, for frequently occurring domain elements, the empirical distribution will give an accurate estimate of their probability mass. There is an extremely long and successful line of work, spanning the past 60 years, from the computer science, statistics, and information theory communities, proposing approaches to “smoothing” the Good–Turing estimates, and combining such smoothed estimates with the empirical distribution (e.g. [19, 16, 21, 22, 23, 15, 4]).

Our approach—to first recover an estimate of the unlabeled vector of probabilities of the true distribution—deviates fundamentally from this previous work, which all attempts to accurately estimate the total probability mass of the domain elements observed \( i \) times. As the following example illustrates, even if one knows the exact total probability comprised of the elements observed \( i \) times, for all \( i \), such knowledge can not be used to yield an optimal learning algorithm, and could
result in an \( \ell_1 \) error that is a factor of two larger than that of our approach.

**Example 4.** Consider \( n \) independent draws from a distribution in which 90\% of the domain elements occur with probability \( 10/n \), and the remaining 10\% occur with probability \( 11/n \). All variants of the Good-Turing frequency estimation scheme would end up, at best, assigning probability \( 10.1/n \) to most of the domain elements, incurring an \( \ell_1 \) error of roughly 0.2. This is because, for elements seen roughly 10 times, the scheme would first calculate that the average mass of such elements is \( 10.1/n \), and then assign this probability to all such elements. Our scheme, on the other hand, would realize that approximately 90\% of such elements have probability \( 10/n \), and 10\% have probability \( 11/n \), but then would assign the probability minimizing the expected error—namely, in this case, our algorithm would assign the median probability, \( 10/n \), to all such elements, incurring an \( \ell_1 \) error of approximately 0.1.

**Worst-case vs Instance Optimal Testing and Learning** Sparked by the seminal work of Goldreich, Goldwasser and Ron [17] and that of Batu et al. [7, 6], there has been a long line of work considering distributional property testing, estimation, and learning questions from a worst case standpoint—typically parameterized via an upper bound on the support size of the distribution from which the samples are drawn (e.g. [8, 25, 5, 20, 21, 30, 27, 26, 11, 28]).

The desire to go beyond this type of worst-case analysis and develop algorithms which provably perform better on “easy” distributions has led to two different veins of further work. One vein considers different common types of structure that a distribution might possess—such as monotonicity, unimodality, skinny tails, etc., and how such structure can be leveraged to yield improved algorithms [13, 9, 14]. While this direction is still within the framework of worst–case analysis, the emphasis is on developing a more nuanced understanding of why “easy” instances are easy.

Another vein of very recent work beyond worst-case analysis (of which this paper is an example) seeks to develop “instance-optimal” algorithms that are capable of exploiting whatever structure is present in the instance. For the problem of identity testing—given the explicit description of description \( p \), deciding whether a set of samples was drawn from \( p \) versus a distribution with \( \ell_1 \) distance at least \( \epsilon \) from \( p \)—recent work gave an algorithm and an explicit function of \( p \) and \( \epsilon \) that represents the sample complexity of this task, for each \( p \) [29]. In a similar spirit, with the dual goals of developing optimal algorithms as well as understanding the fundamental limits of when such instance–optimality is not possible, Acharya et al. have a line of work from the perspective of competitive analysis [1, 2, 3, 4]. Broadly, this work explores the following question: to what extent can an algorithm perform as well as if it knew, a priori, the structure of the problem instance on which it was run? For example, the work [2] considers the two-distribution identity testing question: given samples drawn from two unknown distributions, \( p \) and \( q \), how many samples are required to distinguish the case that \( p = q \) from \( \|p - q\|_1 \geq \epsilon \)? They show that if \( n_{p,q} \) is the number of samples required by an algorithm that knows, ahead of time, the unlabeled vector of probabilities of \( p \) and \( q \), then the sample complexity is bounded by \( n_{p,q}^{3/2} \), and that, in general, a polynomial blowup is necessary—there exists \( p, q \) for which no algorithm can perform this task using fewer than \( n_{p,q}^{7/6} \) samples.

**Relation to [26, 28]** This present paper has two technical parts: the first component is recovering an approximation to the unlabeled vector of probabilities, and the second part is a leveraging of the recovered unlabeled vector of probabilities to output a labeled vector. The majority of the approach and technical machinery that we employ for the first part is based the ideas and techniques in [26, 28]—particularly a Chebyshev polynomial earthmover scheme, which was also repurposed.
for a rather different purpose in [27]. The papers [26, 28] were concerned with developing estimators for entropy, support size, etc.—properties that depend only on the unlabeled vector of probabilities. The emphasis in those papers, in contrast to this, was on giving tight worst-case bounds on these estimation tasks for the class of distributions with a given support size. In addition to the very different goals of this paper,

1.2 Definitions

We refer to the unlabeled vector of probabilities of a distribution as the histogram of the distribution. This is simply the histogram, in the usual sense of the word, of the vector of probabilities of the domain elements. We give a formal definition:

**Definition 1.** The histogram of a distribution $p$, with a finite or countably infinite support, is a mapping $h_p : [0, 1] \to \mathbb{N} \cup \{0\}$, where $h_p(x)$ is equal to the number of domain elements that each occur in distribution $p$ with probability $x$. Formally, $h_p(x) = |\{\alpha : p(\alpha) = x\}|$, where $p(\alpha)$ is the probability mass that distribution $p$ assigns to domain element $\alpha$. We will also allow for “generalized histograms” in which $h_p$ does not necessarily take integral values.

In analogy with the histogram of a distribution, it will be convenient to have an unlabeled representation of the set of samples. We define the fingerprint of a set of samples, which essentially removes all the label-information:

**Definition 2.** Given samples $X = (x_1, \ldots, x_n)$, the associated fingerprint, $F = (F_1, F_2, \ldots)$, is the “histogram of the histogram” of the sample. Formally, $F$ is the vector whose $i$th component, $F_i$, is the number of elements in the domain that occur exactly $i$ times in $X$.

We note that in some of the literature, the fingerprint is alternately termed the pattern, histogram, histogram of the histogram or collision statistics of the samples.

The following metric will be useful for comparing histograms:

**Definition 3.** For two distributions $p_1, p_2$ with respective histograms $h_1, h_2$, and a real number $\tau \in [0, 1]$, we define the $\tau$-truncated relative earthmover distance between them, $R_\tau(p_1, p_2) := R_\tau(h_1, h_2)$, as the minimum over all schemes of moving the probability mass in the first histogram to yield the second histogram, where the cost per unit mass of moving from probability $x$ to probability $y$ is $|\log \max(x, \tau) - \log \max(y, \tau)|$.

The following fact, whose proof is contained in Appendix A, relates the $\tau$-truncated relative earthmover distance between two distributions, $p_1, p_2$, to an analogous but weaker statement about the $\ell_1$ distance between $p_1$ and a distribution obtained from $p_2$ by choosing an optimal relabeling of the support:

**Fact 1.** Given two distributions $p_1, p_2$ satisfying $R_\tau(p_1, p_2) \leq \epsilon$, there exists a relabeling $\pi$ of the support of $p_2$ such that

$$\sum_i |\max(p_1(i), \tau) - \max(p_2(\pi(i)), \tau)| \leq 2\epsilon.$$
2 Recovering the histogram

For clarity of exposition, we state the algorithm and its analysis in terms of two positive constants, $B, C$, which can be defined arbitrarily provided the following inequalities hold:

$$0.1 > B > C > \frac{B}{2} > 0.$$ 

**Algorithm 1.**

**Input:** Fingerprint $F$ obtained from $n$-samples.

**Output:** Histogram $h_{LP}$.

- Define the set $X := \{ \frac{1}{n}, 2\cdot \frac{1}{n}, 3\cdot \frac{1}{n}, \ldots, \frac{n^B + n^C}{n} \}$.
- For each $x \in X$, define the associated variable $v_x$, and consider the solution to the following linear program:

$$\text{Minimize } \sum_{i=1}^{n^B} |F_i - \sum_{x \in X} \text{poi}(nx, i) \cdot v_x|$$

Subject to:

- $\sum_{x \in X} x \cdot v_x + \sum_{i > n^B + 2n^C} \frac{i}{n} F_i = 1$ (total prob. mass = 1)
- $\forall x \in X, v_x \geq 0$ (histogram entries are non-negative)

- Let $h_{LP}$ be the histogram formed by setting $h_{LP}(x_i) = v_{x_i}$ for all $i$, where $(v_x)$ is the solution to the linear program, and then for each integer $i > n^B + 2n^C$, incrementing $h_{LP}(\frac{i}{n})$ by $F_i$.

The following theorem quantifies the performance of the above algorithm:

**Theorem 2.** There exists an absolute constant $c$ such that for sufficiently large $n$ and any $w \in [1, \log n]$, given $n$ independent draws from a distribution $p$ with histogram $h$, with probability $1 - e^{-n^{O(1)}}$ the generalized histogram $h_{LP}$ returned by Algorithm 1 satisfies

$$R_w^{n \log n}(h, h_{LP}) \leq \frac{c}{\sqrt{w}}.$$ 

By Fact 1 this theorem is stronger than Proposition 1 modulo the fact that the entries of the histogram returned by the above algorithm are non-integral. In Appendix C we provide a simple algorithm that rounds a generalized histogram to an (integral) histogram, while changing it very little in relative earthmover distance $R_0(\cdot, \cdot)$. Together with the above theorem, this yields the specific statement of Proposition 1.

The proof of the above theorem relies on an explicit earthmover scheme that leverages a Chebyshev polynomial construction similar to that employed in [26]. The two key properties of the scheme are 1) the truncated relative earthmover cost of the scheme is small, and 2) given two histograms that have similar expected fingerprints, the results of applying the scheme to the pair of histograms will result in histograms that are very close to each other in truncated relative earthmover distance. The technical details differ slightly from those in [26], and are given in a self-contained fashion in Appendix B.
Disambiguating the histogram

Our algorithm for disambiguating the histogram—for assigning to each domain element an estimate of its true probability—proceeds in two stages: first we use the $n$ samples received to run Algorithm 1 and return an estimate of the histogram; then we use this estimate of the histogram to pick a sequence of probabilities $(m_j)$ with which we label the samples—where each domain element seen 1 time in the samples is attributed a probability $m_1$, each domain element seen 2 times in the samples is attributed a probability $m_2$, etc. Because the construction of $m_j$ will be used more generally in the proofs of this section, we state it as a separate definition.

**Definition 4.** Given a histogram $h$, let $S_h$ be the multiset of probabilities of domain elements—that is, for each probability $x$ for which $h(x)$ is some positive integer $i$, add $i$ copies of $x$ to $S$. Given a number of samples $n$, and an index $j$, consider weighting each element $x \in S_h$ by $\text{poi}(nx, j)$. Define $m_{h,j,n}$ to be the median of this weighted multiset.

Explicitly, the median of a weighted set of real numbers is a number $m$ such that at most half the weight lies on numbers greater than $m$, and at most half lies on numbers less than $m$. Taking advantage of the medians defined above, our reconstruction algorithm follows:

**Algorithm 2.**
Input: $n$ samples from a distribution $h$.
Output: An assignment of a probability to each nonzero entry of $h$.
- Run Algorithm 1 to return a histogram $u$.
- Modify $u$ to create $\bar{u}$ by, for each $j \leq \log^2 n$ adding $\frac{n}{j \log^2 n}$ elements of probability $\frac{j}{n}$ and removing corresponding mass arbitrarily from the rest of the distribution.
- Then to each fingerprint entry $j < \log^2 n$, assign those domain elements probability $m_{\bar{u},j,n}$, (as defined in Definition 3) and to each higher fingerprint entry $j \geq \log^2 n$ assign those domain elements their empirical probability $\frac{j}{n}$.

Our main theorem, restated here for convenience, characterizes the performance of the above algorithm:

**Theorem 1** There is a function $\text{err}(n)$ that goes to zero as $n$ gets large, such that Algorithm 2, when given as input $n$ independent draws from any distribution $p$ of discrete support, outputs a labelled vector $q$, such that

$$E[||p - q||_1] \leq \text{opt}(p, n) + \text{err}(n),$$

where $\text{opt}(p, n)$ is the minimum expected error that any algorithm could achieve on the following learning task: given $p$, and given $n$ samples drawn independently from a distribution that is identical to $p$ up to an arbitrary relabeling of the domain elements, learn the distribution.

The proof of Theorem 1 relies on constructing an estimate, $\text{dev}_{j,n}(A, m_{B,j,n})$, that captures the expected contribution to the $\ell_1$ error due to elements that occur exactly $j$ times, given that the true distribution we are trying to reconstruct has histogram $A$, and our reconstruction is based on
the medians \( m_{B,j,n} \) derived from a (possibly different) histogram \( B \). The proof then has two main components. First we show that \( \text{dev}_{j,n}(h, m_{h,j,n}) \) approximately captures the performance of the optimal algorithm with very high probability, namely that using the true histogram \( h \) to choose medians \( m_{h,j,n} \) lets us estimate the performance of the best possible algorithm. Next, we show that the clean functional form of this estimate implies that \( \text{dev}(\cdot, \cdot) \) varies slowly with respect to changes in the second histogram, and thus that with only negligible performance loss we may reconstruct distributions using medians derived from an estimate \( u \) of the true histogram, thus allowing us to analyze the real performance of Algorithm \( 2 \).

3.1 Separating out the probabilistic portion of the analysis

Our analysis is somewhat delicate because we reuse the same samples both to estimate the histogram \( h \), and then to label the domain elements given an approximate histogram. For this reason, we will very carefully separate out the probabilistic portion of the sampling process, identifying a list of convenient properties which happen with very high probability in the sampling process, and then deterministically analyze the case when these properties hold, which we will refer to as a “faithful” set \( S \) of samples from the distribution.

We first describe a simple discretization of histograms \( h \), dividing the domain into buckets which will simplify further analysis, and is a crucial component of the definition of “faithful”.

**Definition 5.** Given a histogram \( h \) and a number of samples \( n \), define the \( k \)th bucket of \( h \) to consist of those histogram entries with probabilities in the half-open interval \( \left( \frac{k}{n \log^2 n}, \frac{k+1}{n \log^2 n} \right) \). Letting \( h_k \) be \( h \) restricted to its \( k \)th bucket, define \( B_{\text{poi}}(j,k) = \sum_{x : h_k(x) \neq 0} h(x) \text{poi}(nx,j) \) to be the expected number of elements from bucket \( k \) that are seen exactly \( j \) times, if \( \text{Poi}(n) \) samples are taken. Given a set of samples \( S \), let \( B_S(j,k) \) be the number of elements in bucket \( k \) of \( h \) that are seen exactly \( j \) times in the samples \( S \), where in both cases \( h \) and \( n \) are implicit in the notation.

Given this notion of “buckets”, we define faithful to mean 1) each domain element is seen roughly the number of times we would expect to see it, and 2) for each pair \((j,k)\), the number of domain elements from bucket \( k \) that are seen exactly \( j \) times is very close to its expectation (where we compute expectations under a Poisson distribution of samples, because “Poissonization” will simplify subsequent analysis).

**Definition 6.** Given a histogram \( h \) and a number of samples \( n \), a set of \( n \) samples, \( S \), is called faithful if:

1. Each item of probability \( x \) appears in the samples a number of times \( j \) satisfying \( |nx - j| < \max\{\log^{1.5} n, \sqrt{n^{1.5} \log n} \} \), and

2. For each \( j < \log^2 n \) and \( k \), we have \( |B_{\text{poi}}(j,k) - B_S(j,k)| < n^{0.6} \).

This notion of “faithful” holds with near certainty, as shown in the following lemma, allowing us to assume in (most of) the proofs in the rest of this section that our learning algorithm receives a faithful set of samples.

**Lemma 1.** For any histogram \( h \) and number of samples \( n \), with probability \( 1 - n^{-\omega(1)} \), a set of \( n \) samples drawn from \( h \) will be faithful.
Proof. Since the number of times an item of probability \( x \) shows up in \( n \) samples is the binomial distribution \( \text{Bin}(n, x) \), the first condition of “faithful”—essentially that this random variable will be within \( \log^{3/4} n \) standard deviations of its mean—follows with probability \( 1 - n^{-\omega(1)} \) from standard Chernoff/Hoeffding bounds.

For the second condition, since \( \text{Poi}(n) \) has probability \( \Theta(1/\sqrt{n}) \) of equaling \( n \), we consider the related process where \( \text{Poi}(n) \) samples are drawn. The number of times each domain element \( x \) is seen is now distributed as \( \text{Poi}(nx) \), independent of each other domain element. Thus the number of elements from bucket \( k \) seen exactly \( j \) times is the sum of independent Bernoulli random variables, one for each domain element in bucket \( k \). The expected number of such elements is \( B_{\text{poi}}(j, k) \) by definition. Since \( B_{\text{poi}}(j, k) \leq n \) by definition, we have that the variance of this random variable is also at most \( n \), and thus Chernoff/Hoeffding bounds imply that the probability that it deviates from its expectation by more than \( n^{0.6} \) is at most \( \exp(-n^{0.1}) \). Thus the probability of such a deviation is at most a \( \Theta(\sqrt{n}) \) factor higher when taking exactly \( n \) samples than when taking \( \text{Poi}(n) \) samples; taking a union bound over all \( j \) and \( k \) yields the desired result.

\[ \square \]

### 3.2 An estimate of the optimal error

We now introduce the key definition of \( \text{dev}(\cdot, \cdot) \), which underpins our analysis of the error of estimation algorithms. The definition of \( \text{dev}(\cdot, \cdot) \) captures the following process: Suppose we have a probability value \( m_j \), and will assign this probability value to every domain element that occurs exactly \( j \) times in the samples. We estimate the expected error of this reconstruction, in terms of the probability that each domain element shows up exactly \( j \) times. While the below definition, stated in terms of a Poisson process, is neither clearly related to the optimal error \( \text{opt}(h, n) \), nor the actual error of any specific algorithm, it has particularly clean properties which will help us show that it can be related to both \( \text{opt}(h, n) \) (in this subsection) as well as the expected error achieved by Algorithm 2 (shown in Section 3.3).

**Definition 7.** Given a histogram \( h \), a real number \( m \), a number of samples \( n \), and a nonnegative integer \( j \), define \( \text{dev}_{j, n}(h, m) = \sum_{x: h(x) \neq 0} |x - m|h(x)\text{Poi}(nx, j) \).

This definition provides crucial motivation for Definition 4 which defined the medians \( m_{h, j, n} \) used in Algorithm 2 since \( m_{h, j, n} \) is the value of \( m \) that minimizes the previous definition, \( \text{dev}_{j, n}(h, m) \). (The median of a—possibly weighted—set of numbers is the location \( m \) that minimizes the total—possibly weighted—distance from the set to \( m \).)

We now show the key result of this section, that \( \text{dev}_{j, n}(h, m_{h, j, n}) \) essentially captures the best possible expected error obtainable on the portion of the distribution seen \( j \) times in the samples.

**Lemma 2.** Given a histogram \( h \), let \( S \) be the multiset of probabilities of a faithful set of samples of size \( n \). For each index \( j < \log^2 n \), consider those domain elements that occur exactly \( j \) times in the samples and let \( S_j \) be the multiset of probabilities of those domain elements. Let \( \sigma_j \) be the sum over \( S_j \) of each element’s distance from the median (counting multiplicity) of \( S_j \). Then
\[
\sum_{j < \log^2 n} |\sigma_j - \text{dev}_{j, n}(h, m_{h, j, n})| = O(\log^{-2} n).
\]

**Proof.** Let \( B_S(j, k) \) denote the number of domain elements from bucket \( k \) that appeared \( j \) times in the sample. From the first condition of the definition of “faithful”, we have that all buckets with probability above \( \frac{2}{n} \log^2 n \) are empty, and for each of the other buckets we have \( |B_S(j, k) - B_{\text{poi}}(k, j)| \leq n^{0.6} \).
Recall that both $\sigma_j$ and $\text{dev}_{j,n}(h,m_{h,j,n})$ compute the total distance of a weighted multiset from its median, where for $\text{dev}_{j,n}(h,m_{h,j,n})$ the multiset is the histogram $h$ with each entry $x$ having multiplicity $h(x)$ and weight $\text{poi}(nx,j)$. Thus the total weight for this multiset within bucket $k$ is exactly $B_{\text{poi}}(j,k)$ by definition. Thus for the two multisets, the amounts of weight in each bucket match to within $n^{0.6}$ for all buckets. This fact is enough to prove our bound, as we explain below.

Consider transforming one weighted multiset into the other, maintaining a bound on how much the total distance from the median changes. We make crucial use of the fact that the “total distance to the median” is robust to small changes in the weighted multiset, since the median is the location that minimizes this total distance. Moving $\alpha$ weight by a distance of $\beta$ can increase the total (weighted) distance to the median by at most $\alpha \cdot \beta$ since this is how much the total weighted distance to the old median changes, and the new median must be at least as good; conversely, such a move cannot decrease the total distance by more than $\alpha \cdot \beta$ as the inverse move would violate the previous bound. Adding $\alpha$ weight to the distribution at distance $\beta$ from the current median similarly cannot decrease the total distance, but also cannot increase the total distance by more than $\alpha \cdot \beta$, with the corresponding statements holding for removing $\alpha$ weight.

Thus, transforming all the $S_j$ into their Poissonized analogs requires two types of transformations: 1) moving up to $n$ samples within their buckets; 2) adding or removing up to $n^{0.6}$ weight from buckets for various combinations of $j$ and $k$. Since buckets have width $1/(n \log^2 n)$, transformations of the first type change the total distance to the median by at most $\log^{-2} n$; since $j < \log^2 n$ and all buckets above probability $\frac{2}{3} \log^2 n$ are empty, transformations of the second type change the total distance by at most the product of the weight adjustment $n^{0.6}$, the number of $j,k$ pairs $2 \log^{3-2} n$, and size of the probability range under consideration which is $\frac{2}{n} \log^2 n$, yielding a bound of $\frac{1}{\sqrt{n}} \log^{3-2} n$. Thus in total the change is $O(\log^{-2} n)$ as desired.

The above lemma essentially shows that $\text{dev}_{j,n}(h,m_{h,j,n})$ captures how well we could hypothetically estimate the probabilities of all the domain elements seen $j$ times, under the unrealistically optimistic assumption that we know the (unlabeled) multiset of probabilities of elements seen $j$ times. Before showing how our algorithm can perform almost this well based on only the samples, we first formalize this reasoning.

**Definition 8.** We call a distribution learner “simple” if all the domain elements seen exactly $j$ times in the samples get assigned the same probability.

Given $n$ samples from a distribution $p$, with $p_{(j)}$ being those domain elements that occurred exactly $j$ times in the sample, we note that the probability of obtaining these samples is invariant to any permutation of $p_{(j)}$. Thus if a hypothetical learner $L$ assigns different probabilities to elements seen $j$ times in the sample, then its average performance over a random permutation of the domain elements can only improve if we simplify $L$ by having it instead assign to all the elements seen $j$ times, the median of the multiset that it was originally assigning.

For this reason, when we are discussing an optimal distribution learner, we will henceforth assume it is simple.

**Lemma 3.** Given a histogram $h$, let $S$ be the multiset of probabilities of a faithful set of samples of size $n$. Given an index $j < \log^2 n$, consider those domain elements that occur exactly $j$ times in the sample; let $S_j$ be the multiset of probabilities of those domain elements. Let $\sigma_j$ be the sum over $S_j$ of each element’s distance from the median of $S_j$ (counting multiplicity). Then any simple learner, when given the sample, must have error at least $\sigma_j$ on the domain elements that appear $j$ times in the sample.
Proof. The median of $S_j$ is the best possible estimate any simple learner can yield—even given the true distribution—so the error of this estimate bounds the performance of a simple learner.  

Combining this with Lemma 2 immediately yields:

**Corollary 1.** For any distribution $h$, the total error of any simple learning algorithm, given $n$ faithful samples from $h$, is at least $\left( \sum_{j < \log^2 n} \text{dev}_{j,n}(h, m_{h,j,n}) \right) - O(\log^{-2} n)$. Further, for any algorithm—simple or not—if we average its performance over all relabelings of the domain of $h$ and the corresponding samples, it will have expected error bounded by the same expression.

### 3.3 Our error estimate is Lipschitz with respect to mis-estimating the distribution

We now relate the error bound of Corollary 1 to the performance of our algorithm, via two steps. The bound in the corollary is in terms of $m_{h,j,n}$, the median computed in terms of the true histogram $h$ which is unknown to the algorithm; instead the algorithm works with an estimate $\bar{u}$ of the true histogram. The next lemma shows that estimating in terms of $\bar{u}$ is almost as good as using $h$.

**Fact 2.** For any distribution $h$, index $j \ge 1$, and real parameter $t \ge 1$, weighting each domain element $x$ by $\text{poi}(nx,j)$, the total weight on domain elements that are at least $t$ standard deviations away from $\frac{j}{n}$—namely, for which $|nx - j| \ge \sqrt{j}t$ is at most $n \cdot \exp(-\Omega(t))$.

**Lemma 4.** Given a number of samples $n$, a histogram $h$ and a second histogram $\bar{u}$ that is 1) close to $h$ in the sense of Proposition 1 in that there exists distributions $p,q$ corresponding to $h, \bar{u}$ respectively for which $\sum_i \max(p(i), \frac{1}{n} \log^{-0.25} n) - \max(q(i), \frac{1}{n} \log^{-0.25} n) \le \log^{-0.25} n$, and 2) the histogram $\bar{u}$ is “fattened” in the sense that for each $j \le \log^2 n$ there are at least $\frac{n}{j \log^{-0.25} n}$ elements of probability $\frac{1}{n}$. Then $\sum_{j < \log^2 n} \text{dev}_{j,n}(h, m_{\bar{u},j,n}) \le o(1) + \sum_{j < \log^2 n} \text{dev}_{j,n}(h, m_{h,j,n})$.

Since for each $j$, as noted earlier, $m_{h,j,n}$ is the quantity which minimizes $\text{dev}_{j,n}(h, m)$, each term $\text{dev}_{j,n}(h, n_{\bar{u},j,n})$ on the left hand side is greater than or equal to the corresponding $\text{dev}_{j,n}(h, n_{h,j,n})$ on the right hand side, so the lemma implies that the left and right hand sides of the expression in the lemma, beyond having related sums, are in fact term-by-term close to each other.

The proof relies on first comparing $m_{h,j}$ and $m_{\bar{u},j}$ to $\frac{j}{n}$, and then showing that $\text{dev}_{j}(h,m)$ is Lipshitz with respect to changes in $h$ of the type described by the guarantees of Proposition 1.

**Proof.** We drop the “,$n$” subscripts here for notational convenience.

Recall that the quantities $m_{h,j}$ and $m_{\bar{u},j}$ are medians computed after weighting by a Poisson function centered at $j$, and thus we would expect these medians to be close to $\frac{j}{n}$. We first show that the “fattening” condition makes $m_{\bar{u},j}$ well-behaved, and then show, given this, that the lemma works both in the case that $m_{h,j}$ is far from $\frac{j}{n}$, and then for the case where they are close.

By condition 2 of the lemma, the “fattening” assumption, for any index $j < \log^2 n$, we have $\sum_{x : \bar{u}(x) \neq 0} h(x) \text{poi}(nx,j) = 1/\log^{O(1)} n$. Thus, by Fact 2 the median $m_{\bar{u},j}$ must satisfy $|n \cdot m_{\bar{u},j} - j| < \sqrt{j} \log^{O(1)} n$, since the fraction of the Poisson-weighted distribution that is at locations more than $\frac{1}{n} \sqrt{j} \log^{O(1)} n$ distance from $\frac{j}{n}$ is (much) less than $1/2$.

Given the above bound on $m_{\bar{u},j}$, we now turn to $m_{h,j}$. Consider the case $|n \cdot m_{h,j} - j| > \sqrt{j} \log^{O(1)} n$. By Fact 2 weighting each domain element $x$ by $\text{poi}(nx,j)$, the total weight on the far side of the median $m_{h,j}$ from $\frac{j}{n}$, is at most $n \cdot \exp(-\Omega(\log^{0.1} n))$. Since (by definition of
“median”) half the weight is on each side of the median, the total weight \( \sum_{x:h(x)\neq 0} h(x) \text{poi}(nx, j) \) must also be bounded by \( n \cdot \exp(-\Omega(\log^{0.1} n)) \). Recall the definition of the left hand side of the inequality of the lemma, \( \text{dev}(h, m_{\tilde{a}, j}) = \sum_{x:h(x)\neq 0} |x - m_{\tilde{a}, j}| h(x) \text{poi}(nx, j) \). Thus for the portion of this sum where \( x < \frac{2}{n} \log^2 n \), since from the previous paragraph \( m_{\tilde{a}, j} \) is also bounded by \( \frac{2}{n} \log^2 n \) for large enough \( n \), we can bound \( \sum_{x < \frac{2}{n} \log^2 n : h(x) \neq 0} |x - m_{\tilde{a}, j}| h(x) \text{poi}(nx, j) \) by the product \( n \cdot \exp(-\Omega(\log^{0.1} n)) \cdot \frac{2}{n} \log^2 n = \exp(-\Omega(\log^{0.1} n)) \). For those \( x \geq \frac{2}{n} \log^2 n \), since \( j < \frac{1}{n} \log^2 n \), we have the tail bounds \( \text{poi}(nx, j) = n^{-\omega(1)} \), implying the total for such \( x \) is also bounded by \( \exp(-\Omega(\log^{0.1} n)) \), which is our final bound for this case—summing these bounds over all \( j < \log^2 n \) yields the desired bound \( \sum_{j < \log^2 n} \text{dev}(h, m_{\tilde{a}, j}) \leq o(1) \), where the sum is over those \( j \) for which this case applies, \( |n \cdot m_{h,j} - j| \geq \sqrt{j} \log^{0.1} n \).

Thus it remains to prove the claim when both \( m_{h,j} \) and \( m_{\tilde{a}, j} \) are close to \( \frac{j}{n} \). To analyze this case, we show that \( \text{dev}(h, m) \) is Lipschitz with respect to the closeness in \( h \) and \( \tilde{u} \) guaranteed by condition 1 of the lemma, provided \( |n \cdot m - j| \leq \sqrt{j} \log^{0.1} n \). The guarantee on \( h \) and \( \tilde{u} \) means that one can transform one distribution into the other by two kinds of transformations: 1) changing the distributions by \( \log^{-0.25} n \) in the \( L_1 \) sense, and 2) arbitrary mass-preserving transformation of elements of probability less than \( \frac{1}{n} \log^{-0.25} n \). We thus bound the change in \( \text{dev}(h, m) \) under both types of transformations.

To analyze \( L_1 \) modifications, consider an arbitrary probability \( x \), and consider the derivative of \( \text{dev}(h, m) \) as we take an element of probability \( x \) and change \( x \). Recalling the definition \( \text{dev}(h, m) = \sum_{x:h(x)\neq 0} |x - m| h(x) \text{poi}(nx, j) \), we see that this derivative equals \( \frac{d}{dx} |x - m| h(x) \text{poi}(nx, j) \), which is bounded (by the product rule and triangle inequality) as \( |x - m| \frac{d}{dx} \text{poi}(nx, j) + |x - m| \text{poi}(nx, j) \), where \( \frac{d}{dx} \text{poi}(nx, j) = n \cdot \text{poi}(nx, j - 1) \cdot (1 - \frac{1}{nx}) \). Rewriting \( m \) as \( m_j \) to indicate its dependence on \( j \), we want to bound the sum of this derivative over \( j < \log^2 n \), since the exact dependence for each individual \( j \) is much harder to talk about than the overall dependence. We have \( \sum_j \text{poi}(nx, j) + |x - m_j| n \cdot \text{poi}(nx, j - 1) \cdot (1 - \frac{1}{nx}) \), where \( \sum_j \text{poi}(nx, j) \leq 1 \). To bound the remaining part of the sum, we first consider the case \( x < \frac{1}{n} \), in which case we bound \( |x - m_j| \leq \frac{1}{n} (1 + j + \sqrt{j} \log^{0.1} n) \) and \( (1 - \frac{nx}{j}) \leq 1 \), thus yielding the bound \( \sum_{j \geq 1} |x - m_j| n \cdot \text{poi}(nx, j - 1) \cdot (1 - \frac{1}{nx}) \leq \sum_{j \geq 0} (2 + j + \sqrt{j} + 1 \log^{0.1} n) \text{poi}(nx, j) \leq \sum_{j \geq 0} (2 + j + \sqrt{j} + 1 \log^{0.1} n) / j = O(\log^{0.1} n) \). For \( x \geq \frac{1}{n} \), since \( \text{poi}(nx, j - 1) \) decays exponentially fast for \( j \) more than \( \sqrt{nx} \) away from \( nx \), we can bound this sum as being on the order of \( \sqrt{nx} \) times its maximum value when \( j \) is in this range. In this range we have \( |x - m_j| \leq |x - \frac{j}{n}| + |\frac{j}{n} - m_j| = \frac{1}{n} O(\sqrt{nx} \log^{0.1} n) \), and \( \text{poi}(nx, j - 1) = O(1/\sqrt{nx}) \), yielding a total bound of \( O(\sqrt{nx} \sqrt{nx} \sqrt{nx} \sqrt{nx} \log^{0.1} n) = O(\log^{0.1} n) \), as in the previous case. Thus we conclude that the sum over all \( j \) of the amount \( \text{dev}(h, m) \) changes with respect to \( L_1 \) changes in \( h \) is \( O(\log^{0.1} n) \).

We next bound the total change to \( \text{dev}(h, m) \) induced by the second type of modification, arbitrary mass-preserving transformations of elements of probability \( x < \frac{1}{n} \log^{-0.25} n \). For \( j = 1 \), we bound the components of \( \text{dev}(h, m) = \sum_{x:h(x)\neq 0} |x - m| h(x) \text{poi}(nx, j) \) by bounding the two terms in the product: \( |x - m| \in [m - \frac{1}{n} \log^{-0.25} n, m + \frac{1}{n} \log^{-0.25} n] \), and \( \text{poi}(nx, 1) = nx \cdot e^{-nx} \in [nx(1 - \log^{-0.25} n)^2, nx] \). Thus for \( m \) either \( m_h \) or \( m_{\tilde{a}} \), since by the assumption of this case \( m \leq \frac{1}{n} (1 + \log^{0.1} n) \), from the bounds above, the contribution to \( \text{dev}(h, m) \) from those \( x < \frac{1}{n} \log^{-0.25} n \) is within \( o(1) \) of \( mn \) times the total mass in the distribution below \( \frac{1}{n} \log^{-0.25} n \), showing that arbitrary modifications of the second type modify \( \text{dev}(h, m) \) by \( o(1) \).

Analyzing the remaining \( j \geq 2 \) terms, omitting the \( |x - m| \) multiplier for the moment, we have
Proof. Recalling the “buckets” from Definition 5, consider for arbitrary integer \( h \) and \( x \), the sum \( \sum_{j \geq 2} (\log^{-0.25} n)^{j-1} (\log^{-0.25} n + j + \sqrt{j} \log^{0.1} n) = o(1) \). Because of the bound that \( m_h, m_\bar{a} \) are each within \( \frac{1}{n} \sqrt{\log^{0.1} n} \) of \( \frac{i}{n} \), we have that \( |x - m| \leq \frac{1}{n} (\log^{-0.25} n + j + \sqrt{j} \log^{0.1} n) \). Thus the change to \( \text{dev}_j(h, m) \) from changes of the second type, summed over all \( j \geq 2 \), is bounded by the sum \( \sum_{j \geq 2} (\log^{-0.25} n)^{j-1} (\log^{-0.25} n + j + \sqrt{j} \log^{0.1} n) = o(1) \), as desired.

Putting the pieces together, the closeness of \( h \) and \( \bar{u} \) implies by the above Lipschitz argument that changing the distribution between \( h \) and \( \bar{u} \), under the fixed median \( m_{\bar{a}, j} \) does not increase \( \text{dev}(\cdot, \cdot) \) too much: \( \sum_{j < \log^2 n} \text{dev}_j(h, m_{\bar{a}, j}) \leq o(1) + \sum_{j < \log^2 n} \text{dev}_j(\bar{u}, m_{\bar{a}, j}) \). Further, since \( m_{\bar{a}, j} \) minimizes this last expression, the right hand side can only increase if we replace \( \text{dev}_j(\bar{u}, m_{\bar{a}, j}) \) by \( \text{dev}_j(\bar{u}, m_{h, j}) \) in this last inequality. Finally, a second application of the same Lipschitz property implies \( \sum_{j < \log^2 n} \text{dev}_j(\bar{u}, m_{h, j}) \leq o(1) + \sum_{j < \log^2 n} \text{dev}_j(h, m_{h, j}) \). Combining these three inequalities yields the bound of the lemma, \( \sum_{j < \log^2 n} \text{dev}_j(h, m_{\bar{a}, j}) \leq o(1) + \sum_{j < \log^2 n} \text{dev}_j(h, m_{h, j}) \), as desired.

Lemma 5. For sufficiently large \( n \), given a fattened distribution \( \bar{\mu} \), for any \( j < \log^2 n \), the median \( m_{\bar{\mu}, j, n} \) is at most \( \frac{2}{n} \log^2 n \).

Proof. Recall that \( m_{\bar{\mu}, j, n} \) is defined as the median of the multiset of probabilities of \( \bar{\mu} \) after each probability \( x \) has been weighted by \( \text{poi}(nx, j) \). For \( x \geq \frac{2}{n} \log^2 n \) and \( j < \log^2 n \), these weights will each be \( n^{-\Omega(1)} \) small by Poisson tail bounds; and because of the fattening, the elements added at probability \( \frac{i}{n} \) will contribute inverse polylogarithmic weight. Since the median must have at most half the weight to its left, the median cannot be as large as our bound \( \frac{2}{n} \log^2 n \), as desired.

Lemma 6. Given a histogram \( h \), a number of samples \( n \), and for each fingerprint entry \( j < \log^2 n \) a probability \( m_j < \frac{2}{n} \log^2 n \) to which we attribute each domain element that shows up \( j \) times in the sample, then for any faithful set of samples from \( h \), the total error made for all \( j < \log^2 n \) is within \( o(1) \) of \( \sum_{j < \log^2 n} \text{dev}_{j,n}(h, m_j) \).

Proof. Recalling the “buckets” from Definition 5, consider for arbitrary integer \( k \), those elements of \( h \) in bucket \( k \), which we denote \( h_k \)—namely, those probabilities of \( h \) lying in the interval \( \left( \frac{k}{n \log^2 n}, \frac{k+1}{n \log^2 n} \right) \), where by the first condition of “faithful”, none of these probabilities are above \( \frac{2}{n} \log^2 n \) for large enough \( n \). Further, let \( S_{j,k} \) be the multiset of probabilities of those domain elements from bucket \( k \) of \( h \) that each get seen exactly \( j \) times in the sample. The total error of our estimate \( m_j \) on bucket \( k \) is thus \( \sum_{x \in S_{j,k}} |m_j - x| \), which since buckets have width \( 1/(n \log^2 n) \), is within \( |S_{j,k}|/(n \log^2 n) \) of \( |S_{j,k}| \cdot |m_j - k/(n \log^2 n)| \), where we have approximated each \( x \) by the left endpoint of the bucket containing \( x \). By the second condition of “faithful”, \( S_{j,k} \) is within \( n^{0.6} \) of its expectation, \( B_{\text{poi}}(j, k) \), and since by assumption \( m_j < \frac{2}{n} \log^2 n \), we have that our previous error bound \( |S_{j,k}| \cdot |m_j - k/(n \log^2 n)| \) is within \( \frac{2}{n \log^2 n} \) of \( B_{\text{poi}}(j, k) \cdot |m_j - k/(n \log^2 n)| \). We rewrite this final expression via the definition of \( B_{\text{poi}} \) as \( \sum_{x : h_k(x) \neq 0} |m - k/(n \log^2 n)| h(x) \text{poi}(nx, j) \). We compare this final expression to the portion of the deviation \( \text{dev}_{j,n}(h, m_j) \) that comes from bucket \( k \), namely \( \sum_{x : h_k(x) \neq 0} |m_j - x| h(x) \text{poi}(nx, j) \), where since \( \sum_{x : h_k(x) \neq 0} |m_j - x| h(x) \text{poi}(nx, j) = B_{\text{poi}}(j, k) \) and \( x \) is within \( 1/(n \log^2 n) \) of \( k/(n \log^2 n) \), the difference between these is clearly bounded by \( B_{\text{poi}}(j, k)/((n \log^2 n) \). Using the triangle inequality to add up the three error terms we have accrued yields that our estimate for the \( L_1 \) error we make for elements seen \( j \) times from bucket \( k \) is accurate to within \( \frac{2}{n \log^2 n} \) and \( \frac{2}{n \log^2 n} \) clearly sums up to \( o(1) \) over all \( j, k \) pairs. Further, since \( S_{j,k} \) is within \( n^{0.6} \) of
$B_{poi}(j, k)$ by the definition of faithful, the sum of the first term is within $o(1)$ of the sum of the third term and it remains only to analyze the third term involving $B_{poi}(j, k)$. From its definition, $\sum_{j,k} B_{poi}(j, k)$ is the expected number of distinct items seen, when making $Poi(n)$ draws from the distribution, throwing out those elements which violate the $j$ and $k$ constraints; hence this sum over all $j, k$ pairs is at most $n$, bounding the total error of our “dev” estimates by $O(1/\log^2 n)$, as desired.

3.4 Proof of Theorem 1

We now assemble the pieces and prove Theorem 1.

Proof of Theorem 1. Consider the output of Algorithm 1 as run in the first step of Algorithm 2. Proposition 1 outlines two cases: with $o(1)$ probability the closeness property outlined in the proposition fails to hold, and in this case, Algorithm 2 may output a distribution up to $L_1$ distance 2 from the true distribution; because this is a low-probability event, this contributes $2 \cdot o(1) = o(1)$ to the expected error. Otherwise, $u$ is close to $h$, and the fattened version $\bar{u}$ is similarly close, which lets us apply Lemma 4 to conclude that $\sum_{j< \log^2 n} dev_{j,n}(h, m_{\bar{u},j,n}) \leq o(1) + \sum_{j< \log^2 n} dev_{j,n}(h, m_{h,j,n})$. Corollary 1 says that $\sum_{j< \log^2 n} dev_{j,n}(h, m_{h,j,n})$ essentially lowerbounds the optimal error $opt(h, n)$, which we combine with the previous bound to yield

$$\sum_{j< \log^2 n} dev_{j,n}(h, m_{\bar{u},j,n}) \leq opt(h, n) + o(1).$$

Lemma 1 guarantees that the samples will be faithful except with $o(1)$ probability, which, as above, means that even if these unfaithful cases contribute the maximum possible distance 2 to the $L_1$ error, the expected contribution from these cases is still $o(1)$, and thus we will assume a faithful set of samples below. Lemmas 5 and 6 imply that for any faithful sample, the error made by Algorithm 2 on attributing those elements seen fewer than $\log^2 n$ times is within $o(1)$ of $\sum_{j< \log^2 n} dev_{j,n}(h, m_{\bar{u},j,n})$, and hence at most $o(1)$ worse than $opt(h, n)$.

Condition 1 of the definition of faithful (Definition 6) implies that all of the elements seen at least $\log^2 n$ times originally had probability at least $\frac{1}{n}(\log^2 n - \log^{1.75} n)$ and that the relative error between the number of times each of these elements is seen and its expectation is thus at most $\log^{-1/4} n$. Thus using the empirical estimate on those elements appearing at least $\log^2 n$ times—as Algorithm 2 does—contributes $O(\log^{-1/4} n)$ total error on these elements. Thus all the sources of error add up to at most $o(1)$ worse than $opt(h, n)$ in expectation, yielding the theorem.

References

[1] J. Acharya, H. Das, A. Jafarpour, A. Orlitsky, and S. Pan. Competitive closeness testing. In Conference on Learning Theory (COLT), 2011.
[2] J. Acharya, H. Das, A. Jafarpour, A. Orlitsky, S. Pan, and A. Suresh. Competitive classification and closeness testing. In Conference on Learning Theory (COLT), 2012.
[3] J. Acharya, H. Das, A. Jafarpour, A. Orlitsky, S. Pan, and A. Suresh. A competitive test for uniformity of monotone distributions. In AISTATS, 2013.
[4] J. Acharya, A. Jafarpour, A. Orlitsky, and A.T. Suresh. Optimal probability estimation with applications to prediction and classification. In COLT, 2013.
[5] T. Batu, S. Dasgupta, R. Kumar, and R. Rubinfeld. The complexity of approximating the entropy. In *Symposium on Theory of Computing (STOC)*, 2002.

[6] T. Batu, E. Fischer, L. Fortnow, R. Kumar, R. Rubinfeld, and P. White. Testing random variables for independence and identity. In *IEEE Symposium on Foundations of Computer Science (FOCS)*, 2001.

[7] T. Batu, L. Fortnow, R. Rubinfeld, W.D. Smith, and P. White. Testing that distributions are close. In *IEEE Symposium on Foundations of Computer Science (FOCS)*, 2000.

[8] T. Batu, R. Kumar, and R. Rubinfeld. Sublinear algorithms for testing monotone and unimodal distributions. In *Symposium on Theory of Computing (STOC)*, 2004.

[9] M. Brautbar and A. Samorodnitsky. Approximating entropy from sublinear samples. In *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2007.

[10] A. Chakrabarti, G. Cormode, and A. McGregor. A near-optimal algorithm for computing the entropy of a stream. In *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2007.

[11] S. Chan, I. Diakonikolas, G. Valiant, and P. Valiant. Optimal algorithms for testing closeness of discrete distributions. In *SODA*, 2014.

[12] S. Chen and J. Goodman. An empirical study of smoothing techniques for language modeling. *Computer Speech and Language*, 13:359–394, 1999.

[13] C. Daskalakis, I. Diakonikolas, R. A. Servedio, G. Valiant, and P. Valiant. Testing k-modal distributions: Optimal algorithms via reductions. *Manuscript*, 2011.

[14] I. Diakonikolas, D. Kane, and V. Nikishkin. Testing identity of structured distributions. In *SODA*, 2015.

[15] E. Drukh and Y. Mansour. Concentration bounds for unigrams language model. In *COLT*, 2004.

[16] W.A. Gale and G. Sampson. Good-Turing frequency estimation without tears. *Journal of Quantitative Linguistics*, 2(3):217–237, 1995.

[17] O. Goldreich, S. Goldwasser, and D. Ron. Property testing and its connection to learning and approximation. In *IEEE Symposium on Foundations of Computer Science (FOCS)*, 1996.

[18] I. J. Good. The population frequencies of species and the estimation of population parameters. *Biometrika*, 40(16):237–264, 1953.

[19] I.J. Good and G.H. Toulmin. The number of new species, and the increase in population coverage, when a sample is increased. *Biometrika*, 43:45–63, 1956.

[20] S. Guha, A. McGregor, and S. Venkatasubramanian. Streaming and sublinear approximation of entropy and information distances. In *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2006.

[21] D.A. McAllester and R.E. Schapire. On the convergence rate of Good-Turing frequency estimators. In *COLT*, 2000.

[22] A. Orlitsky, N.P. Santhanam, and J. Zhang. Always Good Turing: asymptotically optimal probability estimation. *Science*, 302(5644):427–431, October 2003.

[23] A. Orlitsky, N.P. Santhanam, and J. Zhang. Always Good Turing: asymptotically optimal probability estimation. In *IEEE Symposium on Foundations of Computer Science (FOCS)*, 2003.

[24] L. Paninski. Estimating entropy on m bins given fewer than m samples. *IEEE Trans. on Information Theory*, 50(9):2200–2203, 2004.
A Proof of Fact 1

For convenience, we restate Fact 1:

Fact 1 Given two distributions $p_1, p_2$ satisfying $R_\tau(p_1, p_2) \leq \epsilon$, there exists a relabeling $\pi$ of the support of $p_2$ such that

$$\sum_i |\max(p_1(i), \tau) - \max(p_2(\pi(i)), \tau)| \leq 2\epsilon.$$

Proof of Fact 1. We relate relative earthmover distance to the minimum $L_1$ distance between relabeled histograms, with a proof that extends to the case where both distances are defined above a cutoff threshold $\tau$. The main idea is to point out that “minimum rearranged” $L_1$ distance can be expressed in a very similar form to earthmover distance. Given two histograms $h_1, h_2$, the minimum $L_1$ distance between any labelings of $h_1$ and $h_2$ is clearly the $L_1$ distance between the labelings where we match up elements of the two histograms in sorted order. Further, this is seen to equal the (regular, not relative) earthmover distance between the histograms $h_1$ and $h_2$, where we consider there to be $h_1(x)$ “histogram mass” at each location $x$ (instead of $h_1(x) \cdot x$ “probability mass” as we did for relative earthmover distance), and place extra histogram entries at 0 as needed so the two histograms have the same total mass.

Given this correspondence, consider an optimal relative earthmover scheme between $h_1$ and $h_2$, and in particular, consider an arbitrary component of this scheme, where some probability mass $\alpha$ gets moved from some location $x$ in one of the distributions to some location $y$ in the other, at cost $\alpha \log \max(x, \tau) \max(y, \tau)$, and suppose without loss of generality that $x \geq y$.

We now reinterpret this move in the $L_1$ sense, translating from moving probability mass to moving histogram mass. In the non-relative earthmover problem, $\alpha$ probability mass at location $x$ corresponds to $\alpha \frac{x}{y}$ “histogram mass” at $x$, which we then move to $y$ at cost $\left(\max(x, \tau) - \max(y, \tau)\right) \frac{\alpha}{y}$; however, to simulate the relative earthmover scheme, we need the full $\frac{\alpha}{y}$ mass to appear at $y$, so we move the remaining $\frac{\alpha}{y} - \frac{\alpha}{x}$ mass up from 0, at cost $\left(\frac{\alpha}{y} - \frac{\alpha}{x}\right)(\max(y, \tau) - \tau)$. 

17
To relate these 3 costs (the original relative earthmover cost, and the two components of the non-relative histogram earthmover cost), we note that if both $x$ and $y$ are less than or equal to $\tau$ then all 3 costs are 0. Otherwise, if $x, y > \tau$ then the first component of the histogram cost equals $(1 - \frac{y}{x})\alpha$ and the second is bounded by this, as $(\frac{y}{x} - \frac{\tau}{x})(\max(y, \tau) - \tau) < (\frac{y}{x} - \frac{\tau}{x})y = (1 - \frac{y}{x})\alpha$. Further, for the case under consideration where $\tau < y \leq x$, we have $(1 - \frac{y}{x})\alpha \leq \alpha \log \frac{x}{y}$, which equals the relative earthmover cost. Thus the histogram cost in this case is at most twice the relative earthmover cost.

In the remaining case, $y \leq \tau < x$, and the second component of the histogram cost equals 0 because $\max(y, \tau) - \tau = 0$. The first component simplifies as $(\max(x, \tau) - \max(y, \tau))\frac{\alpha}{x} = (x - \tau)\frac{\alpha}{x} = (1 - \frac{\tau}{x})\alpha \leq \alpha \log \frac{x}{\tau}$, where this last expression is the relative earthmover cost. Thus in all cases, the histogram cost is at most twice the relative earthmoving cost.

Since the histogram cost was one particular “histogram moving scheme”, and as we argued above, the “minimum permuted $L_1$ distance” is the minimum over all such schemes, we conclude that this $L_1$ distance is at most twice the relative earthmover distance, as desired.

\[ \square \]

## B Proof of Theorem 2

In this section, we prove Theorem 2 characterizing the performance of the Algorithm 1 which recovers an accurate approximation of the histogram of the true distribution. For convenience, we restate Theorem 2:

**Fact 1** There exists an absolute constant $c$ such that for sufficiently large $n$ and any $w \in [1, \log n]$, given $n$ independent draws from a distribution $p$ with histogram $h$, with probability $1 - e^{-\Omega(1)}$ the generalized histogram $h_{\text{LP}}$ returned by Algorithm 1 satisfies

\[
R_{w n \log n} (h, h_{\text{LP}}) \leq c \sqrt{w}.
\]

The proof decomposes into three parts. In Appendix B.1 we compartmentalize the probabilistic portion of the proof by defining a set of conditions that are satisfied with high probability, such that if the samples in question satisfy the properties, then the algorithm will succeed. This section is analogous to the definition of a “faithful” set of samples of Definition 6 and we re-use the terminology of “faithful”. In Appendix B.2 we show that, provided the samples in question are “faithful”, there exists a feasible solution to the linear program defined in Algorithm 1 which 1) has small objective function value, and 2) is very close to the true histogram from which the samples were drawn, in terms of $\tau$-truncated relative earthmover distance—for an appropriate choice of $\tau$. In Appendix B.3 we show that if two feasible solutions to the linear program defined in Algorithm 1 both have small objective function value, then they are close in tau-truncated relative earthmover distance. The key tool here is a Chebyshev polynomial earthmover scheme. Finally, in Appendix B.4 we put together the above pieces to prove Theorem 2 given the existence of a feasible point that has low-objective function value that is close to the true histogram, and the fact that any two solutions that both have low objective function value must be close to each other, it follows that the solution to the linear program that is found in Algorithm 1 must be close to the true histogram.
B.1 Compartmentalizing the Probabilistic Portion

The following condition defines what it means for a set of samples drawn from a distribution to be “faithful” with respect to positive constants $B, D \in (0, 1)$:

**Definition 9.** A set of $n$ samples with fingerprint $F$, drawn from a distribution $p$ with histogram $h$, is said to be faithful with respect to positive constants $B, D \in (0, 1)$ if the following conditions hold:

- For all $i$,
  \[ |F_i - \sum_{x : h(x) \neq 0} h(x) \cdot \text{poi}(nx, i)| \leq \max \left( F_i^{\frac{1}{2}+D}, n^{B(\frac{1}{2}+D)} \right). \]

- For all domain elements $i$, letting $p(i)$ denote the true probability of $i$, the number of times $i$ occurs in the samples from $p$ differs from $n \cdot p(i)$ by at most
  \[ \max \left( (n \cdot p(i))^{\frac{1}{2}+D}, n^{B(\frac{1}{2}+D)} \right). \]

- The “large” portion of the fingerprint $F$ does not contain too many more samples than expected: Specifically,
  \[ \sum_{i > n^{B+2n^{\delta}}} F_i \leq n^{1/2+D} + n \sum_{x \leq \frac{nB+2n^{\delta}}{n} : h(x) > 0} x \cdot h(x). \]

The following proposition is proven via the standard “Poissonization” technique and Chernoff bounds.

**Proposition 2.** For any constants $B, D \in (0, 1)$, there is a constant $\alpha > 0$ and integer $n_0$ such that for any $n \geq n_0$, a set of $n$ samples consisting of independent draws from a distribution is “faithful” with respect to $B, D$ with probability at least $1 - e^{-n^\alpha}$.

**Proof.** We first analyze the case of a $\text{Poi}(n)$-sized sample drawn from a distribution with histogram $h$. Thus
\[ \mathbb{E}[F_i] = \sum_{x : h(x) \neq 0} h(x) \cdot \text{poi}(nx, i). \]

Additionally, the number of times each domain element occurs is independent of the number of times the other domain elements occur, and thus each fingerprint entry $F_i$ is the sum of independent random 0/1 variables, representing whether each domain element occurred exactly $i$ times in the samples (i.e. contributing 1 towards $F_i$). By independence, Chernoff bounds apply.

We split the analysis into two cases, according to whether $\mathbb{E}[F_i] \geq n^B$. In the case that $\mathbb{E}[F_i] < n^B$, we leverage the basic Chernoff bound that if $X$ is the sum of independent 0/1 random variables with $\mathbb{E}[X] \leq S$, then for any $\delta \in (0, 1)$,
\[ \Pr[|X - \mathbb{E}[X]| \geq \delta S] \leq 2e^{-\delta^2 S/3}. \]
Applied to our present setting where $F_i$ is a sum of independent 0/1 random variables, provided $E[F_i] < n^B$, we have:

$$\Pr \left[ |F_i - E[F_i]| > (n^B)^{\frac{1}{2}+D} \right] \leq 2e^{-\left( \frac{1}{(n^B)^{1/2-D}} \right)^2 \frac{n^B}{3}} = 2e^{-n^{2BD}/3}.$$ 

In the case that $E[F_i] \geq n^B$, the same Chernoff bound yields

$$\Pr \left[ |F_i - E[F_i]| \geq E[F_i]^{\frac{1}{2}+D} \right] \leq 2e^{-\left( \frac{E[F_i]^{1/2-D}}{E[F_i]} \right)^2 \frac{E[F_i]}{3}} = 2e^{-E[F_i]^{2D}/3} \leq 2e^{-n^{2BD}/3}.$$ 

A union bound over the first $n$ fingerprints shows that the probability that given a set of samples (consisting of $Poi(n)$ draws), the probability that any of the fingerprint entries violate the first condition of faithful is at most $n \cdot 2e^{-\frac{n^{2BD}}{3}} \leq e^{-\Omega(1)}$ as desired.

For the second condition of “faithful”, in analogy with the above argument, for any $\lambda \leq S$, and $\delta \in (0, 1)$,

$$\Pr[|Poi(\lambda) - \lambda| > \delta S] \leq 2e^{-\delta^2 S/3}.$$ 

Hence for $x = n \cdot p(i) \geq n^B$, the probability that the number of occurrences of domain element $i$ differs from its expectation of $n \cdot p(i)$ by at least $(n \cdot p(i))^{\frac{1}{2}+D}$ is bounded by $2e^{-\delta^2 (n \cdot p(i))^{2D}/3} \leq e^{-n^{\Omega(1)}}$.

Similarly, in the case that $x = n \cdot p(i) < n^B$,

$$\Pr[|Poi(x) - x| > n^{\frac{1}{2}+D}] \leq e^{-n^{\Omega(1)}}.$$ 

For the third condition, by the Poisson tail bounds of the previous paragraph, the total aggregate number of occurrences of all elements with probability greater than $\frac{n^B + n^C}{n}$ will differ from its expectation by at most $n^{1/2+D}$, with probability $1 - e^{-n^{\Omega(1)}}$. Additionally, by the first condition of “faithful”, with probability $1 - e^{-n^{\Omega(1)}}$ no domain element $i$ with $p(i) < \frac{n^B + n^C}{n}$ will appear more than $n^B + 2n^C$. Hence with probability $1 - e^{-n^{\Omega(1)}}$ all elements that contribute to the sum $\sum_{i>n^B+2n^C} F_i$ will have probability greater than $\frac{n^B + n^C}{n}$. The third condition then follows by a union bound over these two $e^{-n^{\Omega(1)}}$ failure probabilities.

Thus we have shown that provided we are considering a sample size of $Poi(n)$, the probability that the conditions hold is at least $1 - e^{-n^{\Omega(1)}}$. To conclude, note that $\Pr[Poi(n) = n] > \frac{1}{3\sqrt{n}}$, and hence the probability that the conditions do not hold for a set of exactly $n$ samples (namely, the probability that they do not hold for a set of $Poi(n)$ samples, conditioned on the sample size being exactly $n$), is at most a factor of $3\sqrt{n}$ larger, and hence this probability of failure is still $e^{-n^{\Omega(1)}}$, as desired.

**B.2 Existence of a Good Feasible Point**

**Proposition 3.** Provided $F$ is a “faithful” fingerprint derived from a distribution with histogram $h$, there exists a feasible point, $(v_x)$, for the linear program of Algorithm ?? with objective function value at most $O(n^{\frac{1}{2}+B+D})$ such that for any $\tau > 1/n^{3/2}$, the $\tau$-truncated relative earthmover distance between the generalized histogram corresponding to $(v_x)$ with the empirical fingerprint $F_{i>n^B+2n^C}$ appended, and the true histogram, $h$, is bounded by $O \left( \max(n^{-B\frac{1}{2}+D}, n^{-(B-C)}) \right)$, where the big $O$ hides an absolute constant.
Proof. Let \((v_x)\) be defined as follows: initialize \((v_x)\) to be identically zero. For each \(y \leq \frac{n^B + n^C}{n}\) s.t. \(h(y) > 0\), increment \(v_x\) by \(h(y)\frac{y}{x}\), where \(x = \min\{x \in X : x \geq y\}\). Finally, define

\[
m := 1 - \left( \sum_{i > n^B + 2n^C} \frac{i}{n} F_i + \sum_{x \in X} x \cdot v_x \right).
\]

If \(m > 0\), increment \(v_x\) by \(m/x\) for \(x = \frac{n^B + n^C}{n}\). If \(m < 0\), then arbitrarily reduce \(v_x\) until a total of \(m\) units of mass have been removed.

We first argue that the \(\tau\)-truncated relative earthmover distance is small, and then will argue about the objective function value. Let \(h'\) denote the histogram obtained by appending the empirical fingerprint \(F_{i > n^B + 2n^C}\) to \((v_x)\). We construct an earthmoving scheme between \(h\) and \(h'\) as follows: 1) for all \(y \leq \frac{n^B + n^C}{n}\) s.t. \(h(y) > 0\), we move \(h(y)\cdot y\) mass to location \(x = \min\{x \in X : x \geq y\}\); 2) for each domain element \(i\) that occurs more than \(n^B + 2n^C\) times, we move \(p(i)\) mass from location \(p(i)\) to \(\frac{X_i}{n}\) where \(X_i\) denotes the number of occurrences of the \(i\)th domain element; 3) finally, whatever discrepancy remains between \(h\) and \(h'\) after the first two earthmoving phases, we move to probability \(\frac{m^B}{n}\). Clearly this is an earthmoving scheme. For \(\tau \geq 1/n^{3/2}\), the \(\tau\)-truncated relative earthmover cost of the first phase is trivially at most \(\log(\frac{n^{B-\tau(B+1/D)} + 1/n^2}{1/n^{3/2}}) = O(1/\sqrt{n})\). By the second condition of “faithful”, the relative earthmover cost of the second phase of the scheme is bounded by \(\log(\frac{n^{B-n(B(1/2+D))}}{n^B}) = O(n^{-B(1/2-D)})\). To bound the cost of the third phase, note that the first phase equates the two histograms below probability \(n^Bn\). By the second condition of “faithful”, after the second phase, there is at most \(O(n^{-B(1/2-D)})\) unmatched probability caused by the discrepancy between \(\frac{X_i}{n}\) and \(p(i)\) for elements observed at least \(n^B + 2n^C\) times. Hence after this \(O(n^{-B(1/2-D)})\) discrepancy is moved to probability \(\frac{m^B}{n}\), the entirety of the remaining discrepancy lies in the probability range \([\frac{n^B}{n}, c]\), where \(c\) is an upper bound on the true probability of an element that does not appear at least \(n^B + 2n^C\) times; from the second condition of “faithful”, \(c \leq \frac{m^B + 4n^C}{n}\), and hence the total \(\tau\)-truncated relative earthmover distance is at most \(O\left(\max(n^{-B(1/2-D)}, n^{-B-C})\right)\), as desired.

To complete the proof of the proposition, note that by construction, \((v_x)\) is a feasible point for the linear program. To see that the objective function is as claimed, note that \(\frac{n}{x} \cdot \text{poi}(nx, i) \leq n\), and since we are rounding the true histogram to probabilities that are multiples of \(1/n^2\), each “fingerprint expectation”, \(\sum_{x \in X} \text{poi}(nx, i) \cdot v_x\), differs from \(\sum_{x : h(x) \neq 0} \text{poi}(nx, i) \cdot h(x)\) by at most \(1/\sqrt{n}\). Together with the first condition of “faithful” which implies that each of the observed fingerprints \(F_i\) satisfies \(|F_i - \sum_{x : h(x) \neq 0} \text{poi}(nx, i) \cdot h(x)| \leq n^{B+D/2}\), we conclude that the total objective function value is at most \(n^B(n^{B+D/2} + 1/\sqrt{n}) = O(n^{B+D+B+C})\).

B.3 The Chebyshev Bump Earthmoving Scheme

**Proposition 4.** Given a “faithful” fingerprint \(F_i\), then any pair of solutions \(v_x, v'_x\) to the linear program of Algorithm 1 that both have objective function values at most \(O(n^{B+D+B+C})\) satisfy the following: for any \(w \in [1, \log n]\), their \(\frac{w}{n \log n}\)-truncated relative earthmover distance \(R_{w/n \log n}(v_x, v'_x) \leq O(1/\sqrt{w})\).

The proof of the above proposition relies on an explicit earthmover scheme that leverages a Chebyshev polynomial construction. The two key properties of the scheme are 1) the truncated
Define \( g \) and each integer \( A \) bump earthmoving scheme. Definition 11. \((\text{histogram resulting from this scheme by definition 12.})\)

The \( \text{Poisson functions, } f \sum_{i \in \{1, \ldots, s\}} \text{real numbers and associated bump center define their associated bump centers} \)

\[ f_i(x) = \sum_{j=0}^{\infty} a_{ij} \text{poi}(nx, j), \text{ such that } \sum_{j=0}^{\infty} |a_{ij}| \leq \beta. \]

Given a generalized histogram \( h \), the scheme works as follows: for each \( x \) such that \( h(x) \neq 0 \), and each integer \( i \geq 0 \), move \( xh(x) \cdot f_i(x) \) units of probability mass from \( x \) to \( c_i \). We denote the histogram resulting from this scheme by \((c, f)(h)\).

Definition 11. A bump earthmoving scheme \((c, f)\) is \([\epsilon, \tau]\)-good if for any generalized histogram \( h \) the \( \tau \)-truncated relative earthmover distance between \( h \) and \((c, f)(h)\) is at most \( \epsilon \).

Below we define the Chebyshev bumps to be a “third order” trigonometric construction:

Definition 12. The Chebyshev bumps are defined in terms of \( n \) as follows. Let \( s = 0.2 \log n \). Define \( g_1(y) = \sum_{j=-s}^{s-1} \cos(jy) \).

\[
g_2(y) = \frac{1}{16s} \left( g_1(y - \frac{3\pi}{2s}) + 3g_1(y - \frac{\pi}{2s}) + 3g_1(y + \frac{\pi}{2s}) + g_1(y + \frac{3\pi}{2s}) \right),
\]

and, for \( i \in \{1, \ldots, s-1\} \) define \( g^i_3(y) := g_2(y - \frac{i\pi}{s}) + g_2(y + \frac{i\pi}{s}) \), and \( g_0^3 = g_2(y) \), and \( g_3^3 = g_2(y + \pi) \). Let \( t_i(x) \) be the linear combination of Chebyshev polynomials so that \( t_i(\cos(y)) = g^i_3(y) \).

We thus define \( s+1 \) functions, the “skinny bumps”, to be \( B_i(x) = t_i(1 - \frac{\pi y}{2s}) \sum_{j=0}^{s-1} \text{poi}(xn, j) \), for \( i \in \{0, \ldots, s\} \). That is, \( B_i(x) \) is related to \( g_3^i(y) \) by the coordinate transformation \( x = \frac{2s}{n}(1 - \cos(y)) \), and scaling by \( \sum_{j=0}^{s-1} \text{poi}(xn, j) \).

Definition 13. The Chebyshev earthmoving scheme is defined in terms of \( n \) as follows: as in Definition 12, let \( s = 0.2 \log n \). For \( i \geq s+1 \), define the \( i \)-th bump function \( f_i(x) = \text{poi}(nx, i-1) \) and associated bump center \( c_i = \frac{i-1}{n} \). For \( i \in \{0, \ldots, s\} \) let \( f_i(x) = B_i(x) \), and for \( i \in \{1, \ldots, s\} \), define their associated bump centers \( c_i = \frac{2s}{n}(1 - \cos(\pi i/s)) \), with \( c_0 = c_1 \).

The following proposition characterizes the key properties of the Chebyshev earthmoving scheme. Namely, that the scheme is, in fact, an earthmoving scheme, that each bump can be expressed as a low-weight linear combination of Poisson functions, and that the scheme inculs a small truncated relative earthmover cost.

Proposition 5. The Chebyshev earthmoving scheme of Definition 13, defined in terms of \( n \), has the following properties:

- For any \( x \geq 0 \),

\[
\sum_{i \geq 0} f_i(x) = 1,
\]

hence the Chebyshev earthmoving scheme is a valid earthmoving scheme.
• Each $B_i(x)$ may be expressed as $\sum_{j=0}^{\infty} a_{ij} \text{poi}(nx, j)$ for $a_{ij}$ satisfying
  $$\sum_{j=0}^{\infty} |a_{ij}| \leq 2n^{0.3}.$$  

• The Chebyshev earthmoving scheme is $O(1/\sqrt{w}, w/n \log n)$-good, for any $w \in [1, \log n]$, where the $O$ notation hides an absolute constant factor.

The proof of the first two bullets of the proposition closely follow the arguments in [26]. For the final bullet point, the intuition of the proof is the following: the $i$th bump $B_i$, with center $c_i = 2ns^{-1}(1 - \cos(i\pi/s)) \approx i^2 \frac{2}{ns}$ has a width of $O(i/\sqrt{w})$, and $B_i(x)$ decays rapidly (as the fourth power) away from its center, $c_i$. Specifically, $B_i(c_i \pm \frac{\alpha}{ns}) \leq O(1/\alpha^4)$. Hence, at worst, the cost of the earthmoving scheme will be dominated by the cost of moving the mass around the smallest $c_i$ that exceeds the truncation parameter $w/n \log n$. Such a bump will have width $O(\sqrt{w}/ns) = O(\sqrt{w}/n \log n)$, which will incur a per-unit mass relative earthmover cost of $O(\sqrt{1/w})$.

For completeness, we give a complete proof of Proposition 5, with the three parts split into distinct lemmas:

**Lemma 7.** For any $x$,
  $$\sum_{i=-s+1}^{s} g_2(x + \frac{\pi i}{s}) = 1,$$
  and $\sum_{i=0}^{\infty} f_i(x) = 1$.

**Proof.** $g_2(y)$ is a linear combination of cosines at integer frequencies $j$, for $j = 0, \ldots, s$, shifted by $\pm \pi/2s$ and $\pm 3\pi/s2$. Since $\sum_{i=-s+1}^{s} g_2(x + \frac{\pi i}{s})$ sums these cosines over all possible multiples of $\pi/s$, we note that all but the frequency 0 terms will cancel. The cos(0) = 1 term will show up once in each $g_1$ term, and thus $1 + 3 + 3 + 1 = 8$ times in each $g_2$ term, and thus $8 \cdot 2s$ times in the sum in question. Together with the normalizing factor of $16s$, the total sum is thus 1, as claimed.

For the second part of the claim,
  $$\sum_{i=0}^{\infty} f_i(x) = \left( \sum_{j=-s+1}^{s} g_2(\cos^{-1}\left(\frac{xn}{2s} - 1 + \frac{\pi j}{s}\right)) \right) \sum_{j=0}^{s-1} \text{poi}(xn, j) + \sum_{j \geq s} \text{poi}(xn, j)$$
  $$= 1 \cdot \sum_{j=0}^{s-1} \text{poi}(xn, j) + \sum_{j \geq s} \text{poi}(xn, j) = 1.$$ 

We now show that each Chebyshev bump may be expressed as a low-weight linear combination of Poisson functions.

**Lemma 8.** Each $B_i(x)$ may be expressed as $\sum_{j=0}^{\infty} a_{ij} \text{poi}(nx, j)$ for $a_{ij}$ satisfying
  $$\sum_{j=0}^{\infty} |a_{ij}| \leq 2n^{0.3}.$$
Proof. Consider decomposing \( g^i_3(y) \) into a linear combination of \( \cos(\ell y) \), for \( \ell \in \{0, \ldots, s\} \). Since \( \cos(-\ell y) = \cos(\ell y) \), \( g^i_1(y) \) consists of one copy of \( \cos(sy) \), two copies of \( \cos(ry) \) for each \( \ell \) between 0 and \( s \), and one copy of \( \cos(0y); \) \( g^i_2(y) \) consists of \( \left( \frac{1}{16s} \right) \) times \( 8 \) copies of different \( g^i_1(y) \)'s, with some shifted so as to introduce sine components, but these sine components are canceled out in the formation of \( g^i_3(y) \), which is a symmetric function for each \( i \). Thus since each \( g^i_3 \) contains at most two \( g^i_2 \)'s, each \( g^i_3(y) \) may be regarded as a linear combination \( \sum_{\ell=0}^s \cos(\ell y) b_{i\ell} \) with the coefficients bounded as \( |b_{i\ell}| \leq \frac{2}{s^2} \).

Since \( t_i \) was defined so that \( t_i(\cos(y)) = g^i_3(y) = \sum_{\ell=0}^s \cos(\ell y) b_{i\ell} \), by the definition of Chebyshev polynomials we have \( t_i(z) = \sum_{\ell=0}^s T_\ell(z) b_{i\ell} \). Thus the bumps are expressed as

\[
B_i(x) = \left( \sum_{\ell=0}^s T_\ell(1 - \frac{x^n}{2s}) b_{i\ell} \right) \left( \sum_{j=0}^{s-1} \text{poi}(x^n, j) \right).
\]

We further express each Chebyshev polynomial via its coefficients as \( T_\ell(1 - \frac{x^n}{2s}) = \sum_{m=0}^\ell \beta_{\ell m} (1 - \frac{x^n}{2s})^m \) and then expand each term via binomial expansion as \( (1 - \frac{x^n}{2s})^m = \sum_{q=0}^m (-\frac{x^n}{2s})^q \binom{m}{q} \) to yield

\[
B_i(x) = \sum_{\ell=0}^s \sum_{m=0}^\ell \sum_{q=0}^m \sum_{j=0}^{s-1} \beta_{\ell m} \left( \frac{-x^n}{2s} \right)^q \binom{m}{q} b_{i\ell} \text{poi}(x^n, j). \]

We note that in general we can reexpress \( x^n \text{poi}(x^n, j) = x^n x^{j+n/q-x^n} j! = \text{poi}(x^n, j+q) \frac{(j+q)!}{j!} \), which finally lets us express \( B_i \) as a linear combination of Poisson functions, for all \( i \in \{0, \ldots, s\} \):

\[
B_i(x) = \sum_{\ell=0}^s \sum_{m=0}^\ell \sum_{q=0}^m \sum_{j=0}^{s-1} \beta_{\ell m} \left( \frac{-1}{2s} \right)^q \binom{m}{q} \frac{(j+q)!}{j!} b_{i\ell} \text{poi}(x^n, j+q). \]

It remains to bound the sum of the absolute values of the coefficients of the Poisson functions. That is, by the triangle inequality, it is sufficient to show that

\[
\sum_{\ell=0}^s \sum_{m=0}^\ell \sum_{q=0}^m \sum_{j=0}^{s-1} \left| \beta_{\ell m} \left( \frac{-1}{2s} \right)^q \binom{m}{q} \frac{(j+q)!}{j!} b_{i\ell} \right| \leq 2n^{0.3}.
\]

We take the sum over \( j \) first: the general fact that \( \sum_{i=0}^{\ell} \binom{m+i}{i} = \binom{\ell}{\ell+1} \) implies that \( \sum_{j=0}^{s-1} \frac{(j+q)!}{j!} = \sum_{j=0}^{s-1} \frac{(j+q)!}{q!} = q! \frac{(s+q)!}{(s-1)!} = \frac{1}{q+1(s-1)!} \), and further, since \( q \leq m \leq \ell \leq s \) we have \( s + q \leq 2s \) which implies that this final expression is bounded as \( \frac{1}{q+1(s-1)!} = \frac{s}{q+1(s-1)!} \leq s \cdot (2s)^q \). Thus we have

\[
\sum_{\ell=0}^s \sum_{m=0}^\ell \sum_{q=0}^m \sum_{j=0}^{s-1} \left| \beta_{\ell m} \left( \frac{-1}{2s} \right)^q \binom{m}{q} \frac{(j+q)!}{j!} b_{i\ell} \right| \leq \sum_{\ell=0}^s \sum_{m=0}^\ell \sum_{q=0}^m \left| \beta_{\ell m} s \frac{(m)!}{q!} b_{i\ell} \right| = \sum_{\ell=0}^s \left| b_{i\ell} \right| \sum_{m=0}^\ell \left| \beta_{\ell m} \right|^2.
\]

Chebyshev polynomials have coefficients whose signs repeat in the pattern \( (+, 0, -), \) and thus we can evaluate the innermost sum exactly as \( |T_\ell(2r)| \), for \( r = \sqrt{-1} \). Since we bounded
| \dot{b}_i | \leq \frac{4}{\pi} above, the quantity to be bounded is now \( s \sum_{\ell=0}^{s} T_{\ell}(2r) \). Since the explicit expression for Chebyshev polynomials yields \( |T_{\ell}(2r)| = \frac{1}{2} \left[ (2 - \sqrt{5})^\ell + (2 + \sqrt{5})^\ell \right] \), and since \( |2 - \sqrt{5}| = (2 + \sqrt{5})^{-\ell} \), we finally bound \( s \sum_{\ell=0}^{s} \frac{1}{2} |T_{\ell}(2r)| \) for \( \ell = 0 \). The bound \( |2 - \sqrt{5}| < (2 + \sqrt{5})^s < 2 \cdot k^{0.3} \), as desired, since \( s = 0.2 \log n \) and \( \log(2 + \sqrt{5}) < 1.5 \) and \( 0.2 \cdot 1.5 = 0.3 \).

The following lemma quantifies the “skinnyness” of the Chebyshev bumps, which is the main component in the proof of the quality of the scheme (the third bullet in Proposition 5).

**Lemma 9.** \(|g_2(y)| \leq \frac{\pi^7}{y^{4s}} \) for \( y \in [\pi, \pi] \setminus (-3\pi/s, 3\pi/s) \), and \(|g_2(y)| \leq 1/2 \) everywhere.

**Proof.** Since \( g_1(y) = \sum_{j=-s}^{s-1} \cos jy \sin(sy) \), and \( \sin((\alpha + \pi)) = -\sin(\alpha) \), we have the following:

\[
g_2(y) = \frac{1}{16s} \left( g_1(y - \frac{3\pi}{2s}) + 3g_1(y - \frac{\pi}{2s}) + 3g_1(y + \frac{\pi}{2s}) + g_1(y + \frac{3\pi}{2s}) \right)
\]

\[
= \frac{1}{16s} \left( \sin(ys + \pi/2) \left( \cot(y/2 - \frac{3\pi}{4s}) - 3 \cot(y/2 - \frac{\pi}{4s}) \right) \right.
\]

\[
\left. + 3 \cot(y/2 + \frac{\pi}{4s}) - \cot(y/2 + \frac{3\pi}{4s}) \right) \right)
\]

Note that \( \left( \cot(y/2 - \frac{3\pi}{4s}) - 3 \cot(y/2 - \frac{\pi}{4s}) + 3 \cot(y/2 + \frac{\pi}{4s}) - \cot(y/2 + \frac{3\pi}{4s}) \right) \) is a discrete approximation to \( (\pi/2s)^3 \) times the third derivative of the cotangent function evaluated at \( y/2 \). Thus it is bounded in magnitude by \( (\pi/2s)^3 \) times the maximum magnitude of \( \frac{d^3}{dx^3} \cot(x) \) in the range \( x \in [\frac{y}{2} - \frac{3\pi}{4s}, \frac{y}{2} + \frac{3\pi}{4s}] \). Since the magnitude of this third derivative is decreasing for \( x \in (0, \pi) \), we can simply evaluate the magnitude of this derivative at \( \frac{y}{2} - \frac{3\pi}{4s} \). We thus have \( \frac{d^3}{dx^3} \cot(x) = \frac{-2(1 + \cos(2x))}{\sin^3(x)} \), whose magnitude is at most \( \frac{6}{(2x/\pi)^3} \) for \( x \in (0, \pi) \). For \( y \in [3\pi/s, \pi] \), we trivially have that \( \frac{y}{2} - \frac{3\pi}{4s} \geq \frac{y}{4} \), and thus we have the following bound:

\[
|\cot(y/2 - \frac{3\pi}{4s}) - 3 \cot(y/2 - \frac{\pi}{4s}) + 3 \cot(y/2 + \frac{\pi}{4s}) - \cot(y/2 + \frac{3\pi}{4s})| \leq \left( \frac{\pi}{2s} \right)^3 \frac{6}{(y/2)^3} \leq \frac{12\pi^7}{y^{4s}}.
\]

Since \( g_2(y) \) is a symmetric function, the same bound holds for \( y \in [-\pi, -3\pi/s] \). Thus \(|g_2(y)| \leq \frac{12\pi^7}{16s^2 \cdot y^{4s}} \) for \( y \in [-\pi, \pi] \setminus (-3\pi/s, 3\pi/s) \). To conclude, note that \( g_2(y) \) attains a global maximum at \( y = 0 \), with \( g_2(0) = \frac{1}{16s} \left( 6 \cot(\pi/4s) - 2 \cot(3\pi/4s) \right) \leq \frac{24\pi^7}{16s \cdot 24\pi} = 1/2 \).

We now prove the final bullet point of Proposition 5.

**Lemma 10.** The Chebyshev earthmoving scheme is \( O(1/\sqrt{w}) \cdot \frac{w}{n \log n} \)-good, for any \( w \in [1, \log n] \), where the \( O \) notation hides an absolute constant factor.

**Proof.** We split this proof into two parts: first we will consider the cost of the portion of the scheme associated with all but the first \( s + 1 \) bumps, and then we consider the cost of the skinny bumps \( f_i \) with \( i \in \{0, \ldots, s\} \).

For the first part, we consider the cost of bumps \( f_i \) for \( i \geq s + 1 \); that is the relative earthmover cost of moving \( \text{poi}(x_n, i) \) mass from \( x \) to \( \frac{x_n}{n} \) summed over \( i \geq s \). By definition of relative earthmover distance, the cost of moving mass from \( x \) to \( \frac{x_n}{n} \) is \( |\log \frac{x_n}{x}| \), which, since \( \log y \leq y - 1 \), we bound by \( x_n - 1 \) when \( i < x_n \) and \( \frac{i}{n} - 1 \) otherwise. We thus split the sum into two parts.
For \( i \geq \lceil xn \rceil \) we have \( \text{poi}(xn, i)(\frac{i}{xn} - 1) = \text{poi}(xn, i - 1) - \text{poi}(xn, i) \). This expression telescopes when summed over \( i \geq \max\{s, \lceil xn \rceil \} \) to yield \( \text{poi}(xn, \max\{s, \lceil xn \rceil \} - 1) = O(\frac{1}{s}) \).

For \( i \leq \lceil xn \rceil - 1 \) we have, since \( i \geq s \), that \( \text{poi}(xn, i)(\frac{xn}{i+1} - 1) \leq \text{poi}(xn, i)((1 + \frac{1}{i})\frac{xn}{i} - 1) = (1 + \frac{1}{i})\text{poi}(xn, i + 1) - \text{poi}(xn, i) \). The \( \frac{1}{i} \) term sums to at most \( \frac{1}{s} \), and the rest telescopes to \( \text{poi}(xn, \lceil xn \rceil) - \text{poi}(xn, s) = O(\frac{1}{s}) \). Thus in total, \( f_i \) for \( i \geq s + 1 \) contributes \( O(\frac{1}{s}) \) to the relative earthmover cost, per unit of weight moved.

We now turn to the skinny bumps \( f_i(x) \) for \( i \leq s \). The simplest case is when \( x \) is outside the region that corresponds to the cosine of a real number — that is, when \( xn \geq 4s \). It is straightforward to show that \( f_i(x) \) is very small in this region. We note the general expression for Chebyshev polynomials: \( T_j(x) = \frac{1}{2}\left[(x - \sqrt{x^2 - 1})^j + (x + \sqrt{x^2 - 1})^j\right] \), whose magnitude we bound by \( |2x|^j \). Further, since \( 2x \leq \frac{2}{e}e^x \), we bound this by \( (\frac{2}{e})^j e^{|x|j} \), which we apply when \( |x| > 1 \). Recall the definition \( f_i(x) = t_i(1 - \frac{xn}{i})\sum_{j=0}^{s-1}\text{poi}(xn, j) \), where \( t_i \) is the polynomial defined so that \( t_i(\cos(y)) = g^s_i(y) \), that is, \( t_i \) is a linear combination of Chebyshev polynomials of degree at most \( s \) and with coefficients summing in magnitude to at most \( 2 \), as was shown in the proof of Lemma 8. Since \( xn > s \), we may bound \( \sum_{j=0}^{s-1}\text{poi}(xn, j) \leq s\text{poi}(xn, s) \). Further, since \( z \leq e^{s-1} \) for all \( z \), letting \( z = \frac{2}{e}^s \) yields \( x \leq 4s \cdot e^{\frac{4s}{e}} - 1 \), from which we may bound \( \text{poi}(xn, s) = \frac{(xn)^s e^{-\frac{xn}{s}}}{s!} \leq e^{-\frac{xn}{s}}(4s \cdot e^{\frac{4s}{e}} - 1)^s = \frac{e^4 e^{\frac{4s}{e}x}}{e^{4s} e^{\frac{4s}{e}x} / s} \leq 4^s e^{-3xn/4} \). We combine this with the above bound on the magnitude of Chebyshev polynomials, \( T_j(z) \leq (\frac{2}{e})^j e^{|z|j} \leq (\frac{2}{e})^j e^{|z|^j} \), where \( z = (1 - \frac{2\pi}{e}) \) yields \( T_j(z) \leq (\frac{2}{e})^s e^{|z|^s} \). Thus \( f_i(x) \leq \text{poly}(s)4^se^{-3xn/4}(\frac{2}{e})^s e^{|z|^s} = \text{poly}(s)(\frac{2}{e})^s e^{-\frac{3xn}{4}} \). Since \( \frac{2\pi}{e} \geq s \) in this case, \( f_i \) is exponentially small in both \( x \) and \( s \); the total cost of this earthmoving scheme, per unit of mass above \( \frac{4s}{e} \) is obtained by multiplying this by the logarithmic relative distance the mass has to move, and summing over the \( s + 1 \) values of \( i \leq s \), and thus remains exponentially small, and is thus trivially bounded by \( O(\frac{1}{\sqrt{s}}) \).

To bound the cost in the remaining case, when \( xn \leq 4s \) and \( i \leq s \), we work with the trigonometric functions \( g^s_i(y) \), instead of \( t_i \) directly. Since mass may be moved freely below probability \( \frac{w}{n \log n} \), we may assume that all the mass below this value is located at probability exactly \( \frac{w}{n \log n} \).

For \( y \in (0, \pi] \), we seek to bound the per-unit-mass relative earthmover cost of, for each \( i \geq 0 \), moving \( g^s_i(y) \) mass from \( \frac{2\pi}{e}(1 - \cos(y)) \) to \( c_i \). By the above comments, it suffices to consider \( y \in [O(\frac{2\pi}{e}), \pi] \). This contribution is at most

\[
\sum_{i=0}^{s} \left| g^s_i(y) \left( \log(1 - \cos(y)) - \log(1 - \cos(\frac{i\pi}{s})) \right) \right|.
\]

We analyze this expression by first showing that for any \( x, x' \in (0, \pi] \),

\[
\left| \log(1 - \cos(x)) - \log(1 - \cos(x')) \right| \leq 2|\log x - \log x'|.
\]

Indeed, this holds because the derivative of \( \log(1 - \cos(x)) \) is positive, and strictly less than the derivative of \( 2 \log x \); this can be seen by noting that the respective derivatives are \( \frac{\sin(y)}{1 - \cos(y)} \) and \( \frac{2}{y} \), and we claim that the second expression is always greater. To compare the two expressions, cross-multiply and take the difference, to yield \( y \sin y - 2 + 2 \cos y \), which we show is always at most 0 by noting that it is 0 when \( y = 0 \) and has derivative \( y \cos y - \sin y \), which is negative since \( y < \tan y \). Thus we have that \( |\log(1 - \cos(y)) - \log(1 - \cos(\frac{i\pi}{s}))| \leq 2|\log y - \log \frac{y}{s}| \); we use this
bound in all but the last step of the analysis. Additionally, we ignore the \( \sum_{j=0}^{s-1} \text{pois}(xn, j) \) term as it is always at most 1.

We will now show that
\[
|g_3^0(y) (\log y - \log \frac{\pi}{s})| + \sum_{i=1}^{s} |g_3^i(y) (\log y - \log \frac{i\pi}{s})| = O\left(\frac{1}{sy}\right),
\]
where the first term is the contribution from \( f_0, c_0 \). For \( i \) such that \( y \in (\frac{(i-3)\pi}{s}, \frac{(i+3)\pi}{s}) \), by the second bounds on \( |g_2| \) in the statement of Lemma 9, \( g_3^i(y) < 1 \), and for each of the at most 6 such \( i \), \(|(\log y - \log \frac{\max(1,i)\pi}{s})| < \frac{1}{sy} \), to yield a contribution of \( O\left(\frac{1}{sy}\right) \). For the contribution from \( i \) such that \( y \leq \frac{(i-3)\pi}{s} \) or \( y \geq \frac{(i+3)\pi}{s} \), the first bound of Lemma 9 yields \( g_3^i(y) = O\left(\frac{1}{y\pi}\right) \). Roughly, the bound will follow from noting that this sum of inverse fourth powers is dominated by the first few terms. Formally, we split up our sum over \( i \in [s] \setminus [\frac{ys}{\pi} - 3, \frac{ys}{\pi} + 3] \) into two parts according to whether \( i > \frac{ys}{\pi} \):

\[
\sum_{i=1}^{s} \frac{1}{(ys - i\pi)^4} |(\log y - \log \frac{i\pi}{s})| \leq \sum_{i \geq \frac{ys}{\pi} + 3} \left(\frac{ys}{\pi} - i\right)^4 (\log i - \log \frac{ys}{\pi}) \leq \pi^4 \int_{w=\frac{ys}{\pi} + 2}^{\infty} \left(\frac{ys}{\pi} - w\right)^4 (\log w - \log \frac{ys}{\pi}).
\]

Since the antiderivative of \( \frac{1}{(\alpha - w)^4} (\log w - \log \alpha) \) with respect to \( w \) is
\[
-2w(w^2 - 3w\alpha + 3\alpha^2) \log w + 2(w - \alpha)^3 \log(w - \alpha) + \alpha(2w^2 - 5w\alpha + 3\alpha^2 + 2\alpha^2 \log \alpha),
\]
the quantity in Equation (1) is equal to the above expression evaluated with \( \alpha = \frac{ys}{\pi} \), and \( w = \alpha + 2 \), to yield
\[
O\left(\frac{1}{ys}\right) - \log \frac{ys}{\pi} + \log(2 + \frac{ys}{\pi}) = O\left(\frac{1}{ys}\right).
\]

A nearly identical argument applies to the portion of the sum for \( i \leq \frac{ys}{\pi} + 3 \), yielding the same asymptotic bound of \( O\left(\frac{1}{ys}\right) \). As it suffices to consider \( y \geq O\left(\frac{\sqrt{m}}{s}\right) \), this bounds the total per-unit mass \( \frac{w}{n \log n} \)-truncated relative earthmover cost as \( O\left(\frac{1}{\sqrt{w}}\right) \), as desired.

\[\Box\]

**B.4 Proof of Theorem 2**

We now assemble the key propositions from the above sections to complete our proof of Theorem 2.

Proposition 2 guarantees that with high probability, the samples will be “faithful”. For the remainder of the proof, we will assume that we are working with a faithful set of \( n \) independent draws from a distribution with true histogram \( h \). Proposition 3 guarantees that there exists a feasible point \( (v_x) \) for the linear program of Algorithm 1 with objective function at most \( O\left(n^{B+C}\right) \), such that if the empirical fingerprint above probability \( \frac{n^B + 2nC}{n} \) is appended, the resulting histogram \( h_1 \) satisfies \( R_x(h, h_1) \leq O\left(\max(n^{-B(D+3)/2}, n^{-(B-C)}\right) \), for any \( \tau \geq 1/n^{3/2} \).

Let \( h_2 \) denote the histogram resulting from Algorithm 1. Hence the portion of \( h_2 \) below probability \( \frac{n^B + 2nC}{n} \) corresponds to a feasible point of the linear program with objective function bounded

27
by $O(n^{1/2+B+C})$. Additionally, $h_1(x)$ and $h_2(x)$ are identical for all $x > \frac{n^{B+nC}}{n}$, as, by construction, they are both zero for all $x \in \left(\frac{n^{B+nC}}{n}, \frac{n^{B+2nC}}{n}\right]$, and are both equal to the empirical distribution of the samples above this region. We will now leverage the Chebyshev earthmoving scheme, via Proposition 5 to argue that for any triangle inequality we have that for any $\lambda_i$, such that when applied to any histogram $g$, the amount of probability mass that ends up at each bump center, $c_i$ is given as $\sum_{j \geq 0} \alpha_{i,j} \sum_{x: h(x) \neq 0} \text{poi}(nx, j)xg(x)$, for some set of coefficients $\alpha_{i,j}$ satisfying for all $i$, $\sum_{j \geq 0} |\alpha_{i,j}| \leq 2n^{0.3}$.

Consider the results of applying the Chebyshev earthmoving scheme to histograms $h_1$ and $h_2$. We first argue that the discrepancy in the amount of probability mass that results at the bump center will be negligible for any $i \geq n^B + 2n^C$. Indeed, since $h_1$ and $h_2$ are identical above probability $\frac{n^B+n^C}{n}$ and $\sum_{i \geq n^B+2n} \text{poi}(\lambda, i) = e^{-\Omega(n)}$ for $\lambda \leq n^B + n^C$, the discrepancy in the mass at all bump centers $c_i$ for $i \geq n^B + 2n^C$ is trivially bounded by $o(1/n)$.

We now address the discrepancy in the mass at the bump centers $c_i$ for $i < n^B + 2n^C$. For any such $i$ the discrepancy is bounded by the following quantity:

$$\left| \sum_{j \geq 0} \alpha_{i,j} \sum_{x: h(x) \neq 0} \text{poi}(nx, j)x(h_1(x) - h_2(x)) \right| = \sum_{j \geq 0} \sum_{x: h(x) \neq 0} \alpha_{i,j} \left| \frac{j}{n} \text{poi}(nx, j + 1) (h_1(x) - h_2(x)) \right|
\leq \sum_{j \geq 1} \alpha_{i,j-1} \left| \frac{j}{n} \sum_{x: h(x) \neq 0} \text{poi}(nx, j) (h_1(x) - h_2(x)) \right|
\leq o(1/n) + \sum_{j = 1} n^{B+4n^C} \frac{j}{n} \sum_{x: h(x) \neq 0} \text{poi}(nx, j) (h_1(x) - h_2(x))
\leq n^{0.3}(n^B + 4n^C)^2 \cdot O(n^{B+C})
= O(n^{0.3+\frac{1}{2}+3B+C-1}).$$

Where, in the third line, we leveraged the bound $\sum_j |\alpha_{i,j}| \leq n^{0.3}$ and the bound of $O(n^{B+C})$ on the linear program objective function corresponding to $h_1$ and $h_2$, which measures the discrepancies between $\sum_x \text{poi}(nx, j)h(x)$ and the corresponding fingerprint entries. Note that the entirety of this discrepancy can be trivially equalized at a relative earthmover cost of $O(n^{0.3+\frac{1}{2}+3B+C-1} \log(n))$.

by, for example, moving this discrepancy to probability value 1. To complete the proof, by the triangle inequality we have that for any $w \in [1, \log n]$, letting $g_1$ and $g_2$ denote the respective results of applying the Chebyshev earthmoving scheme to histograms $h_1$ and $h_2$, we have the
\[ R_{\frac{\log n}{n \log n}}(h, h_2) \leq R_{\frac{\log n}{n \log n}}(h, h_1) + R_{\frac{\log n}{n \log n}}(h_1, g_1) + R_{\frac{\log n}{n \log n}}(g_1, g_2) + R_{\frac{\log n}{n \log n}}(g_2, h_2) \]
\[ \leq O \left( \max(n^{-B - D}, n^{-(B - C)}) \right) + O(1/\sqrt{w}) + O(n^{0.3 + (3B + C - 1) \log(n)}) + O(1/\sqrt{w}) \]
\[ \leq O(1/\sqrt{w}). \]

C  Rounding a Generalized Histogram

Algorithm \ref{alg:round} returns a generalized histogram. Recall that generalized histograms are histograms but without the condition that their values are integers, and thus may not correspond to actual distributions—whose histogram entries are always integral. While a generalized distribution suffices to establish Theorem \ref{thm:main}, we observe that it is possible to round a generalized histogram without significantly altering it, in truncated relative earthmover distance. The following algorithm and lemma characterizing its performance show one way to round the generalized histogram to obtain a histogram that is close in truncated relative earthmover distance. This, together with Theorem \ref{thm:alg2}, establishes Proposition \ref{prop:round}.

**Algorithm 3. Round to Histogram**

**Input:** Generalized histogram \( g \).

**Output:** Histogram \( h \).

- Initialize \( h \) to consist of the integral elements of \( g \).
- For each integer \( j \geq 0 \):
  - Let \( x_{j1}, x_{j2}, \ldots, x_{j\ell} \) be the elements of the support of \( g \) that lie in the range \( [2^{-(j+1)}, 2^{-j}] \) and that have non-integral histogram entries; let \( m := \sum_{i=1}^{\ell} x_{ji}g(x_{ji}) \) be the total mass represented; initialize histogram \( h' \) to be identically 0 and set variable \( \text{diff} := 0 \).
  - For \( i = 1, \ldots, \ell \):
    * If \( \text{diff} \leq 0 \) set \( h'(x_{ji}) = \lceil g(x_{ji}) \rceil \), otherwise, if \( \text{diff} > 0 \) set \( h'(x_{ji}) = \lfloor g(x_{ji}) \rfloor \).
    * Increment \( \text{diff} \) by \( x_{ji}(h'(x_{ji}) - g(x_{ji})) \).
  - For each \( i \in 1, \ldots, \ell \) increment \( h(\frac{m}{m + \text{diff}} x_{ji}) \) by \( h'(x_{ji}) \).

**Lemma 11.** Let \( h \) be the output of running Algorithm \ref{alg:round} on generalized histogram \( g \). The following conditions hold:

- For all \( x, h(x) \in \mathbb{N} \cup \{0\} \), and \( \sum_{x: h(x) \neq 0} x h(x) = 1 \), hence \( h \) is a histogram of a distribution.
- \( R_0(h, g) \leq 20\alpha \) where \( \alpha := \max(x : g(x) \not\in \mathbb{N} \cup \{0\}) \).

**Proof.** For each stage \( j \) of Algorithm \ref{alg:round} the algorithm goes through each of the histogram entries \( g(x_{ji}) \) rounding them up or down to corresponding values \( h'(x_{ji}) \) and storing the cumulative difference in probability mass in the variable \( \text{diff} \). Thus if this region of \( g \) initially had probability
mass \( m \), then \( h' \) will have probability mass \( m + \text{diff} \). We bound this by noting that since the first element of each stage is always rounded up, and \( 2^{-(j+1)} \) is the smallest possible coordinate in this stage, the mass of \( h' \), namely \( m + \text{diff} \), is always at least \( 2^{-(j+1)} \). Since each element of \( h' \) is scaled by \( \frac{m}{m + \text{diff}} \) before being added to \( h \), the total mass contributed by stage \( j \) to \( h \) is exactly \( m \), meaning that each stage of rounding is “mass-preserving”.

Denoting by \( g_j \) the portion of \( g \) considered in stage \( j \), and denoting by \( h_j \) this stage’s contribution to \( h \), we now seek to bound \( R(h_j, g_j) \).

Recall the cumulative distribution, which for any distribution over the reals, and any number \( y \), is the total amount of probability mass in the distribution between 0 and \( y \). Given a generalized histogram \( g \), we can define its (generalized) cumulative distribution by \( c(g)(x) := \sum_{x \leq y, g(x) \neq 0} x g(x) \). We note that at each stage \( j \) of the Algorithm and in each iteration \( i \) of the inner loop, the variable \( \text{diff} \) equals the difference between the cumulative distributions of \( h' \) and \( g_j \) at \( x_{ji} \), and hence also on the region immediately to the right of \( x_{ji} \). Further, we note that at iteration \( i \), \(|\text{diff}| \) is bounded by \( x_{ji} \) since at each iteration, if \( \text{diff} \) is positive it will decrease and if it is negative it will increase, and since \( h'(x_{ji}) \) is a rounded version of \( g(x_{ji}) \), \( \text{diff} \) will be changed by \( x_{ji}(h'(x_{ji}) - g(x_{ji})) \) which has magnitude at most \( x_{ji} \). Combining these two observations yields that for all \( x \), \(|c(h')(x) - c(g_j)(x)| \leq x \).

To bound the relative earthmover distance we note that for distributions over the reals, the earthmover distance between two distributions can be expressed as the integral of the absolute value of the difference between their cumulative distributions; since relative earthmover distance can be related to the standard earthmover distance by changing each \( x \) value to \( \log x \), the change of variables theorem gives us that \( R(a, b) = \int \frac{1}{x} |c(b)(x) - c(a)(x)| \, dx \). We can thus use the bound from the previous paragraph in this equation after one modification: since \( h' \) has total probability mass \( m + \text{diff} \), its relative earthmover distance to \( g_j \) with probability mass \( m \) is undefined, and we thus define \( h'' \) to be \( h' \) with the modification that we subtract \( \text{diff} \) probability mass from location \( 2^{-j} \) (it does not matter to this formalism if \( \text{diff} \) is negative, or if this makes \( h''(2^{-j}) \) negative). We thus have that \( R(h'', g_j) = \int_{2^{-j}}^{2^{-j+1}} \frac{1}{x} |c(h')(x) - c(g_j)(x)| \, dx \leq \int_{2^{-j+1}}^{2^{-j}} \frac{1}{x} \, dx = 2^{-j+1} \).

We now bound the relative earthmover distance from \( h'' \) to \( h_j \) via the following two-part earth-moving scheme: all of the mass in \( h'' \) that comes from \( h' \) (specifically, all the mass except the \(-\text{diff} \) mass added at \( 2^{-j} \)) is moved to a \( \frac{m + \text{diff}}{m + \text{diff}} \) fraction of its original location, at a relative earthmover cost \( (m + \text{diff}) \cdot \log \frac{m}{m + \text{diff}} \); the remaining \(-\text{diff} \) mass is moved wherever needed, involving changing its location by a factor as much as \( 2 \cdot \max\{\frac{m}{m + \text{diff}}, \frac{m + \text{diff}}{m}\} \) at a relative earthmover cost of at most \( \max\{\frac{m}{m + \text{diff}}, \frac{m + \text{diff}}{m}\} \) at a relative earthmover cost of at most \( \frac{m}{m + \text{diff}} \). Thus our total bound on \( R(g_j, h_j) \), by the triangle inequality, is \( 2^{-(j+1)} + (m + \text{diff}) \cdot \log \frac{m}{m + \text{diff}} + |\text{diff}| \cdot (\log 2 + \log \frac{m}{m + \text{diff}}) \), which we use when \( m \geq 2^{-j} \), in conjunction with the two bounds derived above, that \(|\text{diff}| \leq 2^{-j} \) and that \( m + \text{diff} \geq 2^{-(j+1)} \), yielding a total bound on the earthmover distance of \( 5 \cdot 2^{-j} \) for the \( j \)th stage when \( m \geq 2^{-j} \). When \( m \leq 2^{-j} \) we note directly that \( m \) mass is being moved a relative distance of at most \( 2 \cdot \max\{\frac{m}{m + \text{diff}}, \frac{m + \text{diff}}{m}\} \) at a cost of \( m \cdot (\log 2 + \log \frac{m}{m + \text{diff}}) \) which we again bound by \( 5 \cdot 2^{-j} \). Thus, summing over all \( j \geq \lceil \log_2 \alpha \rceil \), yields a bound of \( 20 \alpha \).