Electronic Colloquium on Computational Complexity, Report No. 21 (2016)

Noisy Population Recovery from Unknown Noise

Shachar Lovett[*]
University of California, San Diego

Jiapeng Zhang[†]
University of California, San Diego

Abstract

The noisy population recovery problem is a statistical inference problem, which is a special case of the problem of learning mixtures of product distributions. Given an unknown distribution on n-bit strings with support of size k, and given access only to noisy samples from it, where each bit is flipped independently with some unknown noise probability, estimate from a few samples the underlying parameters of the model. Previous work designed quasi-polynomial time algorithms which work under the assumption that the noise parameters are exactly known. In this work, we remove this assumption, and show how to recover all the underlying parameters, even when the noise is unknown, in quasi-polynomial time.

[*] Email: slovett@cs.ucsd.edu. Research supported by NSF CAREER award 1350481.

[†] Email: jpeng.zhang@gmail.com. Research supported by NSF CAREER award 1350481.

ISSN 1433-8092

1 Introduction

Consider a database of patients in a hospital, where for each patient the database lists a large number of traits. Researchers are interested in obtaining this database to perform various statistical studies, but due to privacy concerns the database cannot be released. A possible solution (other than deleting identifying parameters of patients, such as their name) is to delete information at random from the database, or even better, add randomness to the information, with the goal that this will maintain the privacy of the original database, but would still provide researchers with useful information. The question is: does this process ensure privacy, or can the original database be recovered (up to its row order) from a lossy or noisy version of it?

The problem of recovery of data from lossy or noisy samples was studied extensively in statistics in the context of continuous distributions, and was introduced to computer science by Kearns et al. [KMR+94] who focused on discrete distributions. In this paper, we focus on the binary setting. Setting parameters, the goal is as follows: given a mixture of k product distributions supported on $\{0,1\}^n$, recover the parameters within accuracy ε as efficiently as possible, by an algorithm which has access to samples from the overall mixed distribution. As far as we know, this may be possible to achieve in time $\mathrm{poly}(n,k,1/\varepsilon)$ (although, there is some evidence that $n^{\log k}$ might be a lower bound, see [FOS08]). However, the best known algorithm to date is by [FOS08], whose algorithm requires running time of $\mathrm{poly}(n,k,1/\varepsilon)^{O(k^3)}$. Thus, it can only recover mixtures of essentially a fixed number of product distributions.

A special case of this problem, termed "population recovery", regained attention recently in a work by Dvir et al. [DRWY12], who related it to the problem of learning DNFs from partial information. The main reason is that in this restricted settings, much better dependence on the sparsity k is possible. It was further studied in [MS13; WY16; BIMP13; LZ15]. This problem is better described as that of recovering an unknown sparse distribution given noisy samples.

Formally, suppose there is an unknown distribution π over k unknown elements in $\{0,1\}^n$ (which we think of as "centers"), and a vector of noise parameters $\vec{\mu} = (\mu_1, \ldots \mu_n) \in [0,1]^n$. Samples from the noisy distribution are obtained as follows:

- Sample a string $v \in \{0,1\}^n$ according to π.
- Output $x = v \oplus e \in \{0,1\}^n$, obtained by sampling each $e_i \in \{0,1\}$ independently with $\Pr[e_i = 0] = (1+\mu_i)/2$.

Note that $\mu_i = 1$ corresponds to no error in coordinate i, while $\mu_i = 0$ corresponds to fully randomizing coordinate i. We denote this noisy distribution by $\sigma(\pi, \vec{\mu})$

The goal is to recover all of the underlying parameters of the model, as efficiently as possible, given samples from the noisy distribution. Most of the work so far [DRWY12; WY16; LZ15] focused on the case where the noise distribution is fully known, and moreover $\mu_1 = \ldots = \mu_n = \mu$ (we note that all these works easily extend to the case where $\mu_1, \ldots, \mu_n$ are exactly known, even if they are not equal). While an unrealistic assumption in practice, the main benefit is that the dependence on the sparsity of the mixture k can be greatly improved. The best results to date are by [LZ15] who show that in this case, all underlying parameters can be learned in time $\mathrm{poly}(n, 1/\varepsilon, k^{\log\log k})$, where we are suppressing the exact dependence on μ. One exception is the work of Batman et al. [BIMP13], who are able to recover in the same time complexity a superset of points in the support of π, even under much weaker assumptions on the noise. However, they cannot recover the remaining parameters (that is, the distribution π and the noise parameters $\vec{\mu}$).

1.1 Our results

The goal of the current work is to relax the conditions under which we can learn mixtures of product distributions. Our main focus is on removing the assumption that the noise parameters have to be known. This question was posed by [WY16] as an open problem; we resolve it in this work. To do that, we need to assume that the noise is somewhat bounded. For $\mu > 0$, we say that the noise parameters $\vec{\mu} = (\mu_1, \ldots, \mu_n)$ are μ-bounded if $\mu_i \geq \mu$ for all $i \in [n]$.

It is easy to see that, when the noise is a parameter, it is impossible in general to recover the "true" parameters $\pi, \vec{\mu}$ from the noisy distribution. For example, let $n = 1$, $\pi(0) = p, \pi(1) = 1 - p$ and $\mu_1 = \mu$. Then the noisy distribution is

$$\sigma(0) = p\frac{1+\mu}{2} + (1-p)\frac{1-\mu}{2} = \frac{1+(2p-1)\mu}{2}, \quad \sigma(1) = 1 - \sigma(0).$$

So, even given perfect knowledge of σ, we can only recover $(2p-1)\mu$. So for example, we cannot distinguish $p = 2/3, \mu = 1$ from $p = 3/4, \mu = 2/3$. Thus, the best that we can is to recover some $\hat{\pi}, \hat{\vec{\mu}}$ such that the noisy distribution $\sigma(\hat{\pi}, \hat{\vec{\mu}})$ is close in statistical distance to the observable noisy distribution. Thus, it gives a succinct representation to the observable data.

Theorem 1.1 (Main theorem). *Fix $\mu > 0$. Let π be an unknown distribution over $\{0,1\}^n$ of support size k. Let $\vec{\mu} \in [\mu, 1]^n$ be unknown noise parameters. Then, for any $\varepsilon > 0$ there exists an algorithm which, given samples from $\sigma(\pi, \vec{\mu})$, returns*

- *A distribution $\hat{\pi}$ over $\{0,1\}^n$ of support size at most k.*
- *A vector $\hat{\vec{\mu}} \in [\mu, 1]^n$ of noise parameters.*

Such that the noisy distributions $\sigma(\pi, \vec{\mu}), \sigma(\hat{\pi}, \hat{\vec{\mu}})$ are ε-close in statistical distance. The algorithm requires time $\mathrm{poly}(n^{\log k}, (1/\varepsilon)^{(\log k)^2}, k^{(\log k)^3})$, where we surpress the exact dependence on μ.

1.2 Proof overview

Our proof has three steps, the first step is to recover a list of candidates of noise parameters, such that one of them is very close to the real noise parameters. Our main contribution is the following theorem.

Theorem 1.2. *Let $V \subset \{0,1\}^n$ be an unknown set of size $|V| = k$, π an unknown probability distribution on V, $\vec{\mu} \in [\mu, 1]^n$ unknown noise parameters. Then, for any $\varepsilon > 0$ there exists an algorithm which, given samples from the noisy distribution $\sigma = \sigma(\pi, \vec{\mu})$, returns a list $L \subset [\mu, 1]^n$ of potential noise parameters such that*

- *There exist $\hat{\mu} \in L$ for which $\|\hat{\mu} - \mu\|_\infty \leq \varepsilon$.*
- $|L| \leq (k/\varepsilon\mu)^{O(\log^2 k)} \cdot n^{\log k}$.

Moreover, the algorithm runs in time $\mathrm{poly}(|L|)$.

Once we recovered almost perfect noise parameters, we will run Wigderson-Yehudayoff's algorithm, which can recover the underlying parameters assuming that the noise parameters are perfectly known. As part of the analysis, we show that there algorithm is robust, in the sense that if we know the noise parameters up to a small error, it still succeeds in recovering the remaining parameters V, π.

Theorem 1.3 (Recovering mixture from almost perfect noise parameters). *Fix $\mu > 0$. Let $\vec{\mu} \in [\mu, 1]^n$ be a vector of noise parameters, and let $\vec{\mu}' \in [\mu, 1]^n$ be assumed noise parameters, where $\|\vec{\mu} - \vec{\mu}'\|_\infty \le \delta$ for $\delta = \varepsilon \cdot k^{-O(\log k)}$. Let π be an unknown distribution over $\{0,1\}^n$ of support size k. Given access to samples from the noisy distribution corresponding to $\sigma(\pi, \vec{\mu})$, the algorithm given in [WY16], which assumes (erroneously) that $\vec{\mu}'$ are the noise parameters, still recovers π within accuracy ε in the same running time.*

We have so far built a short list of potential candidates for the distribution π and the noise parameters $\vec{\mu}$. In the last stage, we prune any distribution which does not match the actual observed noisy distribution. This part is generic and does not rely on the specific properties of our parametric model, except that we can both sample and compute probability of individual elements. See Lemma 3.6 for the details.

1.3 Open problems

An obvious open problem is to improve the parameters in Theorem 1.2. Concretely, if we can apply the method from [LZ15] to the framework of this paper, then it can improve our algorithm runtime to $\mathrm{poly}((n \cdot k/\varepsilon)^{(\log k)\cdot(\log\log k)^2})$.

Another open problem is to extend our current algorithm to the more general model of mixtures of product distributions. We note that in this case, the best algorithms have exponential dependency on k, in contrast with the current work where the dependence is quasi-polynomial (eg of the form $\exp(\log(k)^{O(1)})$.

2 Preliminaries

2.1 Noisy distributions

We recall and slightly adapt some definitions from the introduction. Let $V \subseteq \{0,1\}^n$ be a collection of $|V| = k$ binary vectors, and let π be a distribution on V, called the *mixing distribution*. Both V and π are unknown, and our goal is to recover V and π from noisy samples. Let $\vec{\mu} = (\mu_1, \dots \mu_n) \in [0,1]^n$ be a vector of (also unknown) noise parameters. We sample from the noisy distribution as follows:

- Sample a string $v \in V$ according to π.
- Output $x = v \oplus e \in \{0,1\}^n$, obtained by sampling each $e_i \in \{0,1\}$ independently with $\Pr[e_i = 0] = (1 + \mu_i)/2$.

We denote the resulting noisy distribution by $\sigma(\pi, \vec{\mu})$. For $0 < \mu < 1$, we say that the noise is μ-bounded if $\mu_i \ge \mu$ for all $i \in [n]$.

It will be useful to view this as applying a noise operator to the original distribution. Slightly abusing notations, identity a distribution π on $\{0,1\}^n$ with the function $\pi : \{0,1\}^n \to \mathbb{R}$ where $\pi(x)$ is the probability that π assigns to x. Let $T_{\vec{\mu}}$ denote the noise operator operating on functions $f : \{0,1\}^n \to \mathbb{R}$, defined as

$$(T_{\vec{\mu}} \circ f)(x) = \mathbb{E}_e[f(x + e)],$$

where $e \in \{0,1\}^n$ is sampled as $\Pr[e_i = 0] = (1 + \mu_i)/2$ independently for each $i \in [n]$. Then $\sigma(\pi, \vec{\mu}) = T_{\vec{\mu}} \circ \pi$.

For a distribution π on $\{0,1\}^n$ and $S \subseteq [n]$, define π_S to be the marginal distribution on $\{0,1\}^S$. For $\vec{\mu}$ let $\mu_S = (\mu_i : i \in S)$. Observe that the noisy distribution on S can be obtained by applying noise to the original distribution on S,

$$(T_{\vec{\mu}} \circ \pi)_S = T_{\mu_S} \circ \pi_S.$$

We denote by $T_{\vec{\mu}}^{-1}, T_{\mu_S}^{-1}$ the inverse operators, which exist whenever the noise is μ-bounded for any $\mu > 1$.

2.2 PID graphs

We follow definitions from [WY16]. Let $V \subset \{0,1\}^n$. A *PID graph* defined over V is a labeled directed graph $G = (V, E, \{S_v : v \in V\})$ defined as follows. The vertices of G are identified with V. For every $v \in V$, we associate a subset $S_v \subseteq [n]$, called the *partial ID* (abbreviated PID) of v. For any $S \subseteq [n]$, denote by $v[S] \in \{0,1\}^S$ the restriction of v to the coordinates in S. The *imposters* of v are the set of vertices which agree with v on S_v,

$$I(v) = \{u \in V \setminus \{v\} : u[S_v] = v[S_v]\}.$$

The edges in G are $E = \{(v,u) : v \in V, u \in I(v)\}$. A partial ID v_S is said to be a *unique ID* (abbreviated UID) for $v \in V$ if $I(v) = \emptyset$. That is, v is uniquely identified by its restriction to S_v.

Wigderson and Yehudayoff proved that for any choice of $V \subset \{0,1\}^n$ there exists a choice of PIDs such that the resulting PID graph is efficient in two ways: the size of the sets S_v, as well as the depth of of the tree, are both at most logarithmic in V. In this paper, all logarithms are in base two.

Theorem 2.1 ([WY16]). *For any subset $V \subset \{0,1\}^n$ of size $|V| = k$, there exists a choice of $S_v \subset [n]$ for each $v \in V$, such that*

(i) $|S_v| \leq \log k$ for all $v \in V$.

(ii) The PID graph $G = (V, E, \{S_v : v \in V\})$ has depth $\leq \log k$.

Moreover, the choices of $\{S_v : v \in V\}$ and the PID graph can be constructed in time $\mathrm{poly}(n,k)$.

2.3 Previous work: learning when the noise parameters are known

Several clever ideas introduced in previous works allow to learn the underlying parameters of the distribution, under the assumption that the noise parameters are known exactly. To recall, the underlying parameters are as follows: an unknown distribution π over an unknown set in $\{0,1\}^n$ of size k. We denote by $V \subset \{0,1\}^n$ the support of π.

First, Dvir et al. [DRWY12] showed that one can reduce to the easier problem, where the support V of π is known. That is, assuming the existence of an algorithm which can recover π given V from noisy samples, we wish to recover both V, π. The high level idea is to recover them one coordinate at a time.First, if $n = 1$ then we can clearly set $V = \{0,1\}$ and recover π. Note that as we allow $\pi(v) = 0$ for some $v \in V$, so its sufficient that V is a superset of the support of π. Given the restriction of V, π to the first $t < n$ coordinates, we extend them to $t+1$ coordiantes as follows. First, extend V to all possible values of the $t+1$ bit; this at most doubles the size of V. Then, run our assumed algorith, which learns π given V. Then, delete from V any element whose probability under π is negligible. Thus, we may assume from now on that V is known while π is unknown; this will only introduce an overhead of n in the overall running time. For the full details, see the original paper [DRWY12].

Next, we introduce the learning algorithm of [WY16], under the assumption that both V and the noise parameters are known. It starts by constructing a PID graph for V of depth $\log k$, where each PID has size $\leq \log k$. We assume below that $\mu_1, \dots, \mu_n \geq \mu$. They do as follows:

- Estimate the restriction of the distribution π to each S_v. As the noise is known exactly, this can be achieved as follows. First, estimate the noisy distribution σ_{S_v}. Then compute

$$\pi|_{S_v} = T^{-1}_{\mu_S} \circ \sigma_{S_v}.$$

It can be shown that in order to estimate $\pi|_{S_v}$ within accuracy ε, we need to estimate σ_{S_v} within accuracy $\varepsilon \mu^{|S_v|}$. Recalling that $|S_v| = \log k$ and surpassing the dependence on μ, this takes $\mathrm{poly}(k, 1/\varepsilon)$ time.

- Solve equations for $\pi(v)$ for each $v \in V$. The main idea is to traverse the PID graph from bottom to top. Given a node $v \in V$, assume that we already recovered $\pi(u)$ for all his children, namely for all $u \in I(v)$. We also know $\Pr[\pi|_{S_v} = v[S_v]]$ by the first step. Then we can solve

$$\pi(v) = \Pr[\pi|_{S_v} = v[S_v]] - \sum_{u \in I(v)} \pi(u).$$

Due to approximation errors, if we are aiming for an overall error of ε in approximating π, it is needed to approximate $\pi|_{S_v}$ to within error of $\varepsilon \cdot k^{-O(\log k)}$, where we surpress the exact dependence on μ. The overall running time of the algorithm is thus $\mathrm{poly}(n, 1/\varepsilon, k^{\log k})$. See the original paper [WY16] for details.

We summarize these in the following theorem.

Theorem 2.2 (Recovering mixture from exact noise parameters). *Fix $\mu > 0$. Let $\vec{\mu} \in [\mu, 1]^n$ be a known vector of noise parameters. Let π be an unknown distribution over $\{0,1\}^n$ of support size k. Given access to samples from the noisy distribution corresponding to $(\pi, \vec{\mu})$, it is possible to recover π within accuracy ε in time*

$$\mathrm{poly}(n, 1/\varepsilon, k^{\log k})$$

where we supress the exact dependence on μ.

We would need a variant of Theorem 2.2 where we assume that the noise parameters are known not exactly, but up to a very small error, so that the existing algorithms would work as is. That is, if we are given noisy samples according to $(\pi, \vec{\mu})$, but instead we believe that the noise parameters are $\vec{\mu}'$ where $|\mu_i - \mu'_i| \leq \delta$ for all $i \in [n]$ where δ is small enough, then the algorithm given in Theorem 2.2 would still succeed in producing an ε approximation of π.

To calculate how small δ needs to be, observe that the noisy distribution is used in the algorithm of [WY16] to estimate $\pi|_{S_v}$ for all PIDs $S_v, v \in V$. So, they estimate $\sigma|_{S_v}$ and calculate $\pi|_{S_v} = T^{-1}_{\mu'_S} \circ \sigma_{S_v}$ instead of $\pi|_{S_v} = T^{-1}_{\mu'_S} \circ \sigma_{S_v}$.

Claim 2.3. *Let $\vec{\mu}, \vec{\mu}' \in [\mu, 1]^n$ be two noise distribution such that $\|\vec{\mu} - \vec{\mu}'\|_\infty \leq \delta$. Then for any $S \subset [n]$ and any distribution σ_S on $\{0,1\}^S$,*

$$\|T^{-1}_{\mu_S} \circ \sigma_S - T^{-1}_{\mu'_S} \circ \sigma_S\|_1 \leq \delta |S| (2/\mu)^{2|S|}.$$

We defer the proof to the appendix (Claim A.1). Note that as $|S_v| \leq \log k$ for all $v \in V$, the statistical distance introduced is $\delta \cdot \mathrm{poly}(k)$ where we surpress the dependence on μ. To recall, the algorithm of [WY16] requires an approximation of each $\pi|_{S_v}$ to within $\varepsilon \cdot k^{-O(\log k)}$ accuracy; this requires us to also choose $\delta = \varepsilon \cdot k^{-O(\log k)}$. We summarize this in the following theorem.

Theorem 1.3 (restated). *Fix $\mu > 0$. Let $\vec{\mu} \in [\mu, 1]^n$ be a vector of noise parameters, and let $\vec{\mu}' \in [\mu, 1]^n$ be assumed noise parameters, where $\|\vec{\mu} - \vec{\mu}'\|_\infty \leq \delta$ for $\delta = \varepsilon \cdot k^{-O(\log k)}$. Let π be an unknown distribution over $\{0,1\}^n$ of support size k. Given access to samples from the noisy distribution corresponding to $\sigma(\pi, \vec{\mu})$, the algorithm given in Theorem 2.2, which assumes (erroneously) that $\vec{\mu}'$ are the noise parameters, still recovers π within accuracy ε in the same time.*

3 Recovering mixtures with unknown noise parameters

Let $V \subset \{0,1\}^n$ be an unknown set of $|V| = k$ vectors, π an unknown distribution on V, and $\vec{\mu} \in [0,1]^n$ an unknown set of noise parameters which are assumed to be μ-bounded for some $0 < \mu < 1$. Our goal is to recover the noise parameters, and then apply Theorem 1.3 to recover the remaining parameters V, π.

Let $\sigma = \sigma(\pi, \vec{\mu}) = T_{\vec{\mu}} \circ \pi$ denote the observed distribution. Recall that σ_S for $S \subseteq [n]$ is the marginal of σ on S.

3.1 Recovering the noise parameters

Our main contribution is an algorithm which recovers a small list of noise parameters, one of which is guaranteed to be close to the true noise parameters.

Theorem 1.2 (restated). *Let $V \subset \{0,1\}^n$ be an unknown set of size $|V| = k$, π an unknown probability distribution on V, $\vec{\mu} \in [\mu, 1]^n$ unknown noise parameters. Then, for any $\varepsilon > 0$ there exists an algorithm which, given samples from the noisy distribution $\sigma = \sigma(\pi, \vec{\mu})$, returns a list $L \subset [\mu, 1]^n$ of potential noise parameters such that*

- *There exist $\hat{\mu} \in L$ for which $\|\hat{\mu} - \mu\|_\infty \leq \varepsilon$.*
- $|L| \leq (k/\varepsilon\mu)^{O(\log^2 k)} \cdot n^{\log k}$.

Moreover, the algorithm runs in time $\mathrm{poly}(|L|, n)$.

Let V be the unknown population set, and let $\{S_v \subseteq [n] : v \in V\}$ be the (unknown) corresponding PIDs guaranteed by Theorem 2.1, so that $|S_v| \leq \log k$ and the resulting PID graph $G = (V, E, \{S_v : v \in V\})$ has depth $\leq \log k$.

For $v \in V$, let $d(v) \in \{1, \ldots, \log k\}$ denote the *depth* of v in G, where the root have depth 1 and leaves have depth $\leq \log k$. Intuitively, the probabilities of nodes with higher depth are easier to recover, as they have less imposters. To formalize this, we introduce the notion of dominant elements.

Definition 3.1 (Dominant elements). *Let $\gamma := \varepsilon/(16k)$. An element $v \in V$ is said to* dominant *if $\pi(v) \geq \gamma^{d(v)}$, and for each $u \in I(v)$, $\pi(u) < \gamma^{d(u)}$.*

Claim 3.2. *There exist at least one dominant element $v \in V$.*

Proof. Let $A = \{v \in V : \pi(v) \geq \gamma^{d(v)}\}$. If A is nonempty, then any node $v \in A$ of maximum depth is dominant. The set A cannot be empty since this would imply that

$$1 = \sum_{v \in V} \pi(v) \leq \sum_{v \in V} \gamma^{d(v)} \leq k\gamma = \varepsilon/16 < 1.$$

Assume that $v \in V$ is dominant, and that furthermore we know $(\mu_i : i \in S_v)$. We next show how to use this to learn all the other noise parameters.

Claim 3.3. *Let $v \in V$ be a dominant element and denote $S := S_v$. Let $\rho := T^{-1}_{\mu_S} \circ \sigma$. For each $i \notin S$ define*

$$\hat{\mu}_i := 2\frac{\rho_{S\cup\{i\}}[v_{S\cup\{i\}}]}{\rho_S(v[S])} - 1.$$

Then $\|\hat{\vec{\mu}} - \vec{\mu}\|_\infty \le \varepsilon/2$, where to recall $\vec{\mu}$ are the real noise parameters.

Proof. Let $d = d(v)$. Initially, note that $\rho_S = \pi_S$ and hence

$$\rho_S(v[S]) = \pi(v[S]) = \pi(v) + \sum_{u \in I(v)} \pi(u).$$

By our assumption that v is dominant, we have $\pi(v) \ge \gamma^d$ and

$$\sum_{u \in I(v)} \pi(u) \le k\gamma^{d+1} \le (k\gamma)\pi(v).$$

Thus,

$$1 \le \frac{\rho_S(v[S])}{\pi(v)} \le 1 + k\gamma.$$

Next, as $\rho_{S\cup\{i\}} = T_{\mu_i} \circ \pi$ then we have

$$\begin{aligned}\rho_{S\cup\{i\}}(v[S\cup\{i\}]) &= (T_{\mu_i} \circ \pi)(v[S\cup\{i\}]) \\ &= \sum_{u \in \{v\}\cup I_v, u_i = v_i} \pi(u)(1+\mu_i)/2 + \sum_{u \in \cup I_v, u_i \ne v_i} \pi(u)(1-\mu_i)/2.\end{aligned}$$

We thus have

$$|\rho_{S\cup\{i\}}(v[S\cup\{i\}]) - \pi(v)(1+\mu_i)/2| \le \sum_{u \in I(v)} \pi(u) \le (k\gamma)\pi(v).$$

Combining the estimates we obtain that

$$|\rho_{S\cup\{i\}}(v[S\cup\{i\}]) - \rho_S(v[S]) \cdot (1+\mu_i)/2| \le 2k\gamma\pi(v) \le 4k\gamma\rho_S(v[S]).$$

Hence

$$\left|\frac{\rho_{S\cup\{i\}}(v[S\cup\{i\}])}{\rho_S(v[S])} - \frac{1+\mu_i}{2}\right| \le 4k\gamma.$$

This implies that $|\hat{\mu}_i - \mu_i| \le 8k\gamma \le \varepsilon/2$. □

In order to apply Claim 3.3, we would need to first guess $v \in V$ to be dominant, and then enumerate over μ_i, $i \in S$. The next claim shows that it suffices to approximate μ_i within a fine enough accuracy.

Claim 3.4. *Let $v \in V$ be a dominant element and denote $S := S_v$. Let $\mu'_i \in [\mu, 1]$ for $i \in S$ be such that $|\mu'_i - \mu_i| \le \delta$ for $\delta = (k/\varepsilon\mu)^{-O(\log k)}$. Define $\rho' := T^{-1}_{\mu'_S} \circ \sigma$. For each $i \notin S$ define*

$$\hat{\mu}_i := 2\frac{\rho'_{S\cup\{i\}}[v_{S\cup\{i\}}]}{\rho'_S(v[S])} - 1.$$

Then $|\hat{\mu}_i - \mu_i| \le \varepsilon$ for all $i \in [n]$.

Proof. Let $\rho = T_{\mu_S}^{-1} \circ \sigma$ be the distribution obtain after removing the correct noises in the coordinates of S. We will show that as expected, $\rho'_{S\cup\{i\}}[v_{S\cup\{i\}}] \approx \rho_{S\cup\{i\}}[v_{S\cup\{i\}}]$ and $\rho'_S[v_S] \approx \rho_S[v_S]$, and then apply Claim 3.3. Formally, we will show for a small enough $\eta > 0$ that

$$\|\rho' - \rho\|_1 \leq \eta.$$

In the proof of Claim 3.3 we showed that

$$\rho_S(v[S]) \geq \pi(v) \geq \gamma^{|S|}.$$

Thus, if we choose $\eta = O(\varepsilon \cdot \gamma^{|S|})$ then we would obtain than both the enumerator and denominator in the definition of $\hat{\mu}_i$ approximate up to a multiplicative factor of $1 \pm O(\varepsilon)$ the corresponding quantities with ρ instead of ρ'; the claim then follows from Claim 3.3.

So, our goal is to show that for a small enough $\delta > 0$ we obtain $\|\rho' - \rho\|_1 \leq \eta$. We have

$$\rho = T_{\mu_S}^{-1} \circ \sigma = \left(\prod_{i\in S} T_{\mu_i}^{-1}\right) \circ \sigma.$$

Explicitly computing this, the inverse noise operator T_μ^{-1} corresponds to the following 2×2 matrix

$$\frac{1}{2}\begin{pmatrix} 1+\mu & 1-\mu \\ 1-\mu & 1+\mu \end{pmatrix}^{-1} = \frac{1}{2\mu}\begin{pmatrix} 1+\mu & -1+\mu \\ -1+\mu & 1+\mu \end{pmatrix}$$

Thus we have

$$\rho(x) = \sum_{y\in\{0,1\}^n : y_{S^c} = x_{S^c}} \prod_{i\in S} \left(\frac{(-1)^{x_i+y_i} + \mu_i}{2\mu_i}\right) \sigma(y)$$

and similarly

$$\rho'(x) = \sum_{y\in\{0,1\}^n : y_{S^c} = x_{S^c}} \prod_{i\in S} \left(\frac{(-1)^{x_i+y_i} + \mu'_i}{2\mu'_i}\right) \sigma(y).$$

We can bound

$$\begin{aligned} &\left|\prod_{i\in S}\left(\frac{(-1)^{x_i+y_i}+\mu_i}{2\mu_i}\right) - \prod_{i\in S}\left(\frac{(-1)^{x_i+y_i}+\mu_i}{2\mu_i}\right)\right| \\ &\leq \frac{1}{\mu^{|S|-1}} \sum_{i\in S} \frac{|\mu'_i - \mu_i|}{2\mu'_i\mu_i} \\ &\leq \frac{|S|\delta}{\mu^{|S|+1}}. \end{aligned}$$

where we assume $\mu'_i, \mu_i \geq \mu$ and $|\mu'_i - \mu_i| \leq \delta$. Thus

$$\|\rho' - \rho\|_1 \leq \frac{|S|\delta 2^{|S|}}{\mu^{|S|+1}} \leq \delta(k/\mu)k^{\log(1/\mu)}\log k.$$

we to obtain $\|\rho' - \rho\|_1 \leq O(\varepsilon\gamma^{|S|})$ we need to choose $\delta = (k/\varepsilon\mu)^{-O(\log k)}$. □

In final step, we need to argue that given some noise parameters μ' we can estimate ρ'.

Claim 3.5. *Fix $S \subseteq [n]$, noise parameters $\mu'_S \in [\mu, 1]^S$. Let $\rho'(x) = T^{-1}_{\mu'_S}\sigma$. Then we can estimate ρ'_S and $\rho'_{S\cup\{i\}}$ for any i within error η, with success probability $1-\delta$, using $(2/\mu)^{O(|S|)}\mathrm{poly}(1/\eta)\log(1/\delta)$ samples.*

Proof. Let $S' = S$ or $S' = S \cup \{i\}$. Estimate $\sigma_{S'}$ within error $\mu^{|S'|}\eta$ with probability $1-\delta$ using standard estimation techniques. As each value of $\rho'_{S'}$ is the linear combination of the elements of $\sigma_{S'}$ with coefficients bounded by $(1/\mu)^{|S'|}$, the claim follows. □

We can now describe the algorithm, that would generate a list of potential suggestions for the noise parameters. The main observation is that in order to compute $\hat{\vec{\mu}}$ given in Claim 3.4, we don't really need to know $v \in V$. We only need to know:

- a PID $S_v \subset [n]$ of size $|S_v| \le \log k$
- The value $v[S_v]$
- A good enough approximation for μ_{S_v}.

Note that Claim 3.4 requires us to know $v[S_v \cup \{i\}]$ for each $i \in [n]$. However, as we assume that v is dominant, this can be easily found from samples. Indeed, let $S = S_v$, fix $i \notin S$ and let $S' = S \cup \{i\}$ and $v' = v \oplus e_i$. Then

$$\pi_{S'}(v'[S']) \le (k\gamma) \cdot \pi_{S'}(v[S]) \le \varepsilon \cdot \pi_{S'}(v[S])$$

That is, if we sample $x \sim \pi$ and condition that $x_S = v_S$, then $\Pr[x_i = v_i] \ge 1 - \varepsilon$. Now, we do not have access to π, but we do have access to $\rho = T^{-1}_{\mu_S} \circ \sigma$. Note that $\rho_{S'} = T_{\mu_i} \circ \pi$, and hence

$$\begin{aligned}\Pr_{x\sim\rho}[x_i = v_i | x_S = v_S] &= \Pr_{x\sim\pi}[x_i = v_i | x_S = v_S] \cdot \left(\frac{1+\mu_i}{2}\right) + \Pr_{x\sim\pi}[x_i = v'_i | x_S = v_S] \cdot \left(\frac{1-\mu_i}{2}\right) \\ &\ge \frac{1+\mu_i}{2} - \varepsilon \\ &\ge \frac{1+\mu}{2} - \varepsilon.\end{aligned}$$

Thus, we can learn all the bits of v by samples. We thus obtain the following algorithm.

Algorithm: Recover-Noise-Parameters

```
• Input: Samples from $\sigma = \sigma(\pi, \vec{\mu})$.
• Output: A list of candidates of noisy parameters.
1. Initialize empty list $L = \emptyset$.
2. Enumerate $S \subseteq [n]$ with $|S| \le \log k$ and $v_S \in \{0,1\}^S$. For each choice:
    2.1 Enumerate all possible values of $\mu'_S \in [\mu, 1]^S$ within accuracy $\delta = (k/\varepsilon\mu)^{-O(\log k)}$ in each coordinate. For each choice:
        2.2.1 For each $i \in [n]$, estimate $T^{-1}_{\mu'_S} \circ \sigma_{S\cup\{i\}}$ and compute $\mu'_i$ for all $i \notin S$ as in Claim 3.4.
        2.2.2 Add $\mu' \in [\mu, 1]^n$ to the list $L$.
4. Output $L$.
```

This concludes the proof of Theorem 1.2: some choice of $v \in V$ is dominant by Claim 3.2. For this v, some choice of μ'_S is δ-close to the true noise parameters on S. By Claim 3.4, this suffices to learn the remaining noise parameters. The running time is dominated by the enumeration of $\vec{\mu}'$, which takes $(1/\delta)^{\log k} = (k/\varepsilon\mu)^{-O(\log^2 k)}$ time.

3.2 The full algorithm

Given the algorithms given in Theorem 1.3 and Theorem 1.2, the full algorithm follows by simply composing the two. To recall, the algorithm given in Theorem 1.3 guarantees to approximate π within an error of ε, assuming knowledge of all the noise parameters to within accuracy of $\delta = 1/\varepsilon \cdot k^{-O(\log k)}$. Its run time is $\text{poly}(n^{\log k}, 1/\varepsilon, k^{\log k})$. The algorithm given in Theorem 1.2 outputs a list L of potential noise parameters. If we want one of them to be a δ-approximation of the true noise parameters, we have $|L| = (k/\delta)^{O(\log^2 k)}$. Thus, we obtain a list of

$$T = \text{poly}(n^{\log k}, (1/\varepsilon)^{\log^2 k}, k^{\log^3 k})$$

potential choices of $\{(\hat{\pi}_i, \hat{\vec{\mu}}_i) : i \in [T]\}$, where it is guaranteed that there exists some $i \in [T]$ for which

$$\|\hat{\pi}_i - \pi\|_1 \leq \varepsilon, \quad \|\hat{\vec{\mu}}_i - \vec{\mu}\|_\infty \leq \delta$$

and hence $\|\sigma(\pi, \vec{\mu}) - \sigma(\hat{\pi}_i, \hat{\vec{\mu}}_i)\|_1 \leq O(\varepsilon)$.

So, we are given a list of potential noisy distributions described by $\hat{\pi}_i, \hat{\vec{\mu}}_i$. Let $\hat{\sigma}_i = \sigma(\hat{\pi}_i, \hat{\vec{\mu}}_i)$. Observe that we can both sample from $\hat{\sigma}_i$, as well as calculate the probability of each specific element. The next lemma shows that this is sufficient to prune any distribution $\hat{\sigma}_i$ which is far from the true distribution σ.

Lemma 3.6 (Pruning lemma). *Let $\mathcal{S} = \{\sigma_i : i \in [T]\}$ be a set of distributions over a universe X, given by some succinct representation, such that we can:*

- *Sample efficiently from each σ_i.*
- *Calculate $\sigma_i(x)$ for each $x \in X$.*

Let σ be an (unknown) observable distribution over X that we can get samples from. Then, for any $\varepsilon > 0$ we can find a subset $\mathcal{S}' \subset \mathcal{S}$ such that:

- *If the statistical distance between σ and σ_i is $\geq 4\varepsilon$, then $\sigma_i \notin \mathcal{S}'$.*
- *If the statistical distance between σ and σ_i is $\leq \varepsilon$, then $\sigma_i \in \mathcal{S}'$.*

Moreover, we can find $\mathcal{S}'$ with high probability in time $O((T/\varepsilon)^2 \cdot \log T)$.

Proof. The main idea is as follows. Assume that $\sigma_i, \sigma_j \in \mathcal{S}$ are two distributions whose statistical distance is $\geq 4\varepsilon$. Then it cannot be the case that both are ε close to σ. The following procedure will reject at least one of them which is not ε close to σ. Define

$$A = \{x : \sigma_i(x) > \sigma_j(x)\}.$$

By definition of statistical distance,

$$\Pr_{x \sim \sigma_i}[x \in A] - \Pr_{x \sim \sigma_j}[x \in A] \geq 4\varepsilon.$$

We will estimate $\Pr[\sigma \in A], \Pr[\sigma_i \in A], \Pr[\sigma_j \in A]$, and reject either σ_i or σ_j (or both) for which the probability is 2ε far from that of σ. To estimate $\Pr[\sigma \in A]$, sample $x \sim \sigma$, calculate $\sigma_i(x), \sigma_j(x)$ and decide whether $x \in A$ or not. Similarly estimate $\Pr[\sigma_j \in A]$ and $\Pr[\sigma_j \in A]$. Reject i if the estimated $\Pr[\sigma \in A] - \Pr[\sigma_i \in A]$ exceed 2ε. To make sure that we make the correct decision with probability $1 - \delta$ we need $O((1/\varepsilon)^2 \log(1/\delta))$ samples.

So, set $\delta = 1/|T|^2$, enumerate all pairs $i, j \in [T]$ and apply the above procedure. Let $\mathcal{S}'$ be the non-rejected distributions. By the above analysis, with high probability we will keep all distributions which are ε-close to σ, and reject all distributions which are 4ε-far from σ. □

The full algorithm is as follows.

Algorithm: Recover-Mixture

- **Input:** Samples from $\sigma = \sigma(\pi, \vec{\mu})$.
- **Output:** $\hat{\pi}, \hat{\vec{\mu}}$ such that $\sigma(\hat{\pi}, \hat{\vec{\mu}})$ is ε-close to σ in statistical distance.

1. Enumerate noise parameters by Recover-Noise-Parameters given in Theorem 1.2.
2. For each noise parameter $\hat{\vec{\mu}}_i$, recover $\hat{\pi}_i$ by applying the Wigderson-Yehudayoff algorithm given in Theorem 1.3.
3. Apply the pruning procedures given in Lemma 3.6 to prune any pair $\hat{\pi}_i, \hat{\vec{\mu}}_i$ whose noisy distribution $\sigma(\hat{\pi}_i, \hat{\vec{\mu}}_i)$ is ε-far from the observable noisy distribution.
4. Output an arbitrary $\hat{\pi}_i, \hat{\vec{\mu}}_i$ which was not pruned.

References

[BIMP13] Lucia Batman, Russell Impagliazzo, Cody Murray, and Ramamohan Paturi. “Finding Heavy Hitters from Lossy or Noisy Data”. In: *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques.* Springer, 2013, pp. 347–362 (cit. on p. 1).

[DRWY12] Zeev Dvir, Anup Rao, Avi Wigderson, and Amir Yehudayoff. “Restriction access”. In: *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference.* ACM. 2012, pp. 19–33 (cit. on pp. 1, 4, 5).

[FOS08] Jon Feldman, Ryan O’Donnell, and Rocco A Servedio. “Learning mixtures of product distributions over discrete domains”. In: *SIAM Journal on Computing* 37.5 (2008), pp. 1536–1564 (cit. on p. 1).

[KMR+94] Michael Kearns, Yishay Mansour, Dana Ron, Ronitt Rubinfeld, Robert E Schapire, and Linda Sellie. “On the learnability of discrete distributions”. In: *Proceedings of the twenty-sixth annual ACM symposium on Theory of computing.* ACM. 1994, pp. 273–282 (cit. on p. 1).

[LZ15] Shachar Lovett and Jiapeng Zhang. “Improved Noisy Population Recovery, and Reverse Bonami-Beckner Inequality for Sparse Functions”. In: *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing.* ACM. 2015, pp. 137–142 (cit. on pp. 1, 3).

[MS13] Ankur Moitra and Michael Saks. “A polynomial time algorithm for lossy population recovery”. In: *Foundations of Computer Science (FOCS), 2013 IEEE 54th Annual Symposium on.* IEEE. 2013, pp. 110–116 (cit. on p. 1).

[WY16] Avi Wigderson and Amir Yehudayoff. “Population recovery and partial identification”. In: *Machine Learning* 102.1 (2016), pp. 29–56 (cit. on pp. 1–6).

A Technical claims

Claim A.1. *Let $\vec{\mu}, \vec{\mu}' \in [\mu, 1]^n$ be two vectors with $\|\vec{\mu} - \vec{\mu}'\|_\infty \le \epsilon$, then for all $f : \{0,1\}^n \to \mathbb{R}$ and $S \subseteq [n]$,*

$$\|T_{\mu_S}^{-1} f - T_{\mu'_S}^{-1} f\|_1 \le (1/\mu)^{2\cdot|S|} \cdot \epsilon \cdot 2^{|S|+1} \cdot |S| \cdot \|f\|_1$$

Proof. To prove the claim, we first need the following claim.

Claim A.2. *Let $\vec{\mu}, \vec{\mu}' \in [0, \mu]^n$ be two vectors with $\|\vec{\mu} - \vec{\mu}'\|_\infty \le \epsilon$, then for all $f : \{0,1\}^n \to \mathbb{R}$ and $S \subseteq [n]$, we have that*

$$\|(T_{\mu_S} - T_{\mu'_S}) f\|_1 \le \epsilon \cdot 2^{|S|+1} \cdot |S| \cdot \|f\|_1.$$

Proof. By the definitions, we have that

$$\begin{aligned}
\|(T_{\mu_S} - T_{\mu'_S})f\|_1 &= \sum_x |(T_{\mu_S} f)(x) - (T_{\mu'_S} f)(x)| \\
&= \sum_x |\mathbb{E}_{e\sim\mu_S}[f(x+e)] - \mathbb{E}_{e\sim\mu'_S}[f(x+e)]| \\
&\le \sum_x \max_{e\in\{0,1\}^S}\{|\Pr_{e'\sim\mu_S}[e'=e] - \Pr_{e'\sim\mu'_S}[e'=e]|\} \cdot \Big(\sum_{e\in\{0,1\}^S} |f(x+e)|\Big) \\
&= 2^{|S|}\|f\|_1 \cdot \max_{e\in\{0,1\}^S}\{|\Pr_{e'\sim\mu_S}[e'=e] - \Pr_{e'\sim\mu'_S}[e'=e]|\} \\
&\le \epsilon \cdot 2^{|S|+1} \cdot |S| \cdot \|f\|_1
\end{aligned}$$

The claim then follows. □

Then by A.2, we have that

$$\begin{aligned}
\|T^{-1}_{\mu_S} f - T^{-1}_{\mu'_S} f\|_1 &= \|T^{-1}_{\mu_S}(f - T_{\vec{\mu}_S} T^{-1}_{\vec{\mu}'_S} f)\|_1 \\
&\le \|T^{-1}_{\mu_S}\|_{1\to 1} \cdot \|f - T_{\mu_S} T^{-1}_{\mu'_S} f\|_1 \\
&\le (1/\mu)^{|S|} \cdot \|f - (T_{\mu'_S} - T_{\mu'_S} + T_{\mu_S}) T^{-1}_{\mu'_S} f\|_1 \\
&= (1/\mu)^{|S|} \cdot \|(T_{\mu'_S} - T_{\mu_S})(T^{-1}_{\mu'_S} f)\|_1 \\
&\le (1/\mu)^{|S|} \cdot \|(T_{\mu'_S} - T_{\mu_S})\|_{1\to 1} \cdot \|T^{-1}_{\mu'_S}\|_{1\to 1} \cdot \|f\|_1 \\
&\le (1/\mu)^{2\cdot|S|} \cdot \epsilon \cdot 2^{|S|+1} \cdot |S| \cdot \|f\|_1
\end{aligned}$$

This completes the proof. □

ECCC ISSN 1433-8092

http://eccc.hpi-web.de