2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)

Beyond trace reconstruction: Population recovery from the deletion channel

Frank Ban
UC Berkeley
fban@berkeley.edu

Xi Chen, Adam Freilich*, Rocco A. Servedio and Sandip Sinha
Columbia University
xichen, rocco, sandip@cs.columbia.edu, *freilich.am@gmail.com

***Abstract*—*Population recovery* is the problem of learning an unknown distribution over an unknown set of n-bit strings, given access to independent draws from the distribution that have been independently corrupted according to some noise channel. Recent work has intensively studied such problems both for the bit-flip noise channel and for the erasure noise channel.**

In this paper we initiate the study of population recovery under the *deletion channel*, in which each bit b is independently *deleted* with some fixed probability and the surviving bits are concatenated and transmitted. This is a far more challenging noise model than bit-flip noise or erasure noise; indeed, even the simplest case in which the population is of size 1 (corresponding to a trivial probability distribution supported on a single string) corresponds to the *trace reconstruction* problem, which is a challenging problem that has received much recent attention.

In this work we give algorithms and lower bounds for population recovery under the deletion channel when the population size is some value $\ell > 1$. As our main sample complexity upper bound, we show that for any population size $\ell = o(\log n/\log\log n)$, a population of ℓ strings from $\{0,1\}^n$ can be learned under deletion channel noise using $2^{n^{1/2+o(1)}}$ samples. On the lower bound side, we show that at least $n^{\Omega(\ell)}$ samples are required to perform population recovery under the deletion channel when the population size is ℓ, for all $\ell \leq n^{1/2-\varepsilon}$.

Our upper bounds are obtained via a robust multivariate generalization of a polynomial-based analysis, due to Krasikov and Roddity [KR97], of how the k-deck of a bit-string uniquely identifies the string; this is a very different approach from recent algorithms for trace reconstruction (the $\ell = 1$ case). Our lower bounds build on moment-matching results of Roos [Roo00] and Daskalakis and Papadimitriou [DP15].

I. Introduction

In recent years the unsupervised learning problem of *population recovery* has emerged as a significant focus of research attention in theoretical computer science [DRWY12], [MS13], [BIMP13], [LZ15], [DST16], [WY16], [PSW17], [DOS17b]. In the population recovery problem there is an unknown distribution $\mathbf{X}$ over an unknown set of n-bit strings from $\{0,1\}^n$, and the learner's job is to reconstruct a high-accuracy approximation of $\mathbf{X}$ given access to noisy independent draws from $\mathbf{X}$ (so each data point which the learning algorithm receives is independently generated as follows: an n-bit string is drawn from $\mathbf{X}$ and corrupted by some noise process, and the result is provided to the learning algorithm). The two noise models which have chiefly been studied to date are the *bit-flip* noise model, in which each coordinate is independently flipped with some fixed probability, and the *erasure* noise model, in which each coordinate is independently replaced by '?' with some fixed probability.

Since the population recovery problem was first introduced in [DRWY12], [WY16], a number of positive results and lower bounds have been obtained for different variants of the problem. In one popular version of the problem [PSW17], [DOS17b], [MS13], for a particular noise model (bit-flip or erasure) the distribution $\mathbf{X}$ may be an arbitrary distribution over $\{0,1\}^n$, and the goal is to learn the distribution $\mathbf{X}$ with respect to ℓ_∞ distance (i.e. to output a list of strings $x^1, \ldots, x^r \in \{0,1\}^n$ and associated weights $\tilde{\mathbf{X}}(x^i)$ such that $|\mathbf{X}(x^i) - \tilde{\mathbf{X}}(x^i)| \leq \varepsilon$ for all $i \in [r]$ and $\mathbf{X}(x) \leq \varepsilon$ for all $x \in \{0,1\}^n \setminus \{x^1, \ldots, x^r\}$). In another well-studied version of the problem [WY16], [LZ15], [DST16], which is closely related to the problems we shall consider, the distribution $\mathbf{X}$ is promised to be supported on at most ℓ strings in $\{0,1\}^n$ (i.e. the "population size" is promised to be at most ℓ), and the goal is to output a hypothesis distribution $\tilde{\mathbf{X}}$ over $\{0,1\}^n$ which has total variation

2575-8454/19/$31.00 ©2019 IEEE
DOI 10.1109/FOCS.2019.00050

distance at most ε from $\mathbf{X}$. Significant progress has been made on determining the sample complexity of population recovery for both of these variants under the bit-flip and erasure noise models; we refer the interested reader to [DST16], [PSW17], [DOS17b] for the current state of the art.

This work: Population recovery from the deletion channel and its relation to trace reconstruction. In both the bit-flip noise model and the erasure noise model, all of the challenge in the population recovery problem stems from the fact that given a noisy draw from $\mathbf{X}$ it is *a priori* not clear which element of $\mathbf{X}$'s support was corrupted by noise to produce the noisy draw. Putting it another way, if the population size is promised to be $\ell = 1$, then under either of these two noise models it is trivially easy to learn a single unknown string from noisy examples.

In this work we study population recovery under the *deletion* noise model, which is far more challenging to handle than either bit-flip noise or erasure noise. The deletion channel is defined as follows: when a string x is passed through the deletion channel with deletion parameter δ, each coordinate x_i is independently deleted with probability δ, the surviving coordinates are concatenated, and the resulting string (of length $\boldsymbol{n}' \leq n$, where $\boldsymbol{n}'$ is distributed as $\mathrm{Bin}(n, 1-\delta)$) is the output of the noise process. Intuitively, the deletion channel is challenging because given a received word obtained by passing x through the δ-deletion channel (often referred to as a *trace* of x, and denoted by $\boldsymbol{z} \leftarrow \mathrm{Del}_\delta(x)$), it is not clear which coordinate of x gave rise to which coordinate of $\boldsymbol{z}$. Indeed, in contrast with the bit-flip and erasure noise models, even if the population size is guaranteed to be $\ell = 1$, the problem of recovering a single unknown string from independent traces is a well-known and challenging open problem, known as the *trace reconstruction problem* [Lev01b], [Lev01a], [BKKM04], [KM05], [HMPW08], [VS08], [MPV14], [DOS17a], [NP17], [PZ17], [HPP18], [HHP18].

There are several motivations for the study of population recovery under the deletion noise model. One motivation is the considerable recent research interest both in the trace reconstruction problem (the $\ell = 1$ case of population recovery under the deletion channel) and in population recovery problems under bit-flip and erasure models. Further motivation comes from potential relevance of the deletion channel population recovery problem both to recovery problems in computational biology and to other topics such as DNA data storage. Regarding biological recovery problems, considering population recovery (the $\ell > 1$ case) rather than trace reconstruction (the $\ell = 1$ case) relaxes the potentially unrealistic assumption that all of the received samples (of a protein sequence, DNA sequence, etc.) are derived from a single unknown target sequence rather than from multiple unknown sequences. Heuristic algorithms for population recovery-type problems have also been applied to DNA storage (see e.g., [YGM17] and [OAC$^+$18]). In these settings, each string in the population comes from a DNA sequence and the noisy channel can inflict a variety of errors including bit-flips and deletions.

Thus, the authors feel that the time is ripe for a theoretical study of population recovery under the challenging deletion model. In this paper we initiate such a study, obtaining sample complexity upper and lower bounds when the population is of size $\ell > 1$. Before describing our results for populations of size ℓ (equivalently, target distributions supported on at most ℓ strings), we first recall known upper and lower bounds for the trace reconstruction problem ($\ell = 1$) below.

Known bounds on trace reconstruction. The trace reconstruction problem was raised more than fifteen years ago [Lev01b], [Lev01a], [BKKM04], though in fact some variants of the problem go back at least to the 1970s [Kal73]. The first algorithm that provably succeeds with high probability in reconstructing an arbitrary $x \in \{0,1\}^n$ using subexponentially many traces is due to Mitzenmacher et al. [HMPW08], who showed that $2^{\tilde{O}(\sqrt{n})}$ many traces suffice for any constant deletion rate δ bounded away from 1. This result was improved in recent simultaneous and independent works of De et al. [DOS17a] and Nazarov and Peres [NP17]; these papers each showed that for any constant δ bounded away from 1, at most $2^{O(n^{1/3})}$ traces suffice to reconstruct any $x \in \{0,1\}^n$.[1]

Due to the seeming difficulty of the worst-case trace reconstruction problem (reconstructing an arbitrary $x \in \{0,1\}^n$), an average-case version of the problem (reconstructing a randomly chosen string $x \in \{0,1\}^n$), which turns out to be significantly easier in terms of sample complexity, has also received considerable attention. A number of early works [BKKM04], [KM05], [VS08] gave efficient algorithms that succeed for trace reconstruction of almost all $x \in \{0,1\}^n$ when the deletion rate δ is sufficiently low ($o_n(1)$ as a function of n). In [HMPW08] Mitzenmacher et al. gave an algorithm

[1] Hartung, Holden and Peres [HHP18] have recently extended this result to certain more general regimes where there can be different deletion probabilities for different coordinates and symbols.

which uses $\mathrm{poly}(n)$ traces to perform average-case trace reconstruction when the deletion rate δ is at most some sufficiently small constant. Recently the best results on average-case trace reconstruction have been significantly strengthened in works of Peres and Zhai [PZ17] and Holden, Pemantle and Peres [HPP18] which build on the worst-case trace reconstruction results of [DOS17a], [NP17]. The latter of these papers [HPP18] gives an algorithm which uses $\exp((\log n)^{1/3})$ traces to reconstruct a random $x \in \{0,1\}^n$ when the deletion rate is any constant bounded away from 1.

In terms of lower bounds, it is easy to see that if the deletion rate δ is at least some positive constant, then until $\Omega(\log n)$ draws have been received there will be some bits of the target string x about which no information has been received. Improving on this simple $\Omega(\log n)$ lower bound, McGregor et al. [MPV14] established a sample complexity lower bound of $\Omega(n)$ traces for any constant deletion rate. This was recently improved by Holden and Lyons [HL18] to $\tilde{\Omega}(n^{5/4})$.

Summarizing, for any constant deletion probability $0 < \delta < 1$ there is currently an exponential gap between the best lower bound of $\tilde{\Omega}(n^{5/4})$ samples and the best upper bound of $2^{O(n^{1/3})}$ samples for trace reconstruction of an arbitrary string $x \in \{0,1\}^n$.

A. Our results

Positive result. As our main positive result, we obtain an algorithm which learns any unknown distribution $\mathbf{X}$ supported on at most ℓ strings under the deletion channel. For any constant ℓ (and in fact even for ℓ as large as $o(\log n/\log\log n)$, its sample complexity is exponential in $n^{1/2+o(1)}$. In more detail, our main positive result is the following:

Theorem 1 (Learning an arbitrary mixture of ℓ strings under the deletion channel)**.** *There is an algorithm with the following performance guarantee: Let $\mathbf{X}$ be an arbitrary distribution over at most ℓ strings in $\{0,1\}^n$. For any deletion rate $0 < \delta < 1$ and any accuracy parameter ε, if the algorithm is given access to independent draws from $\mathbf{X}$ that are independently corrupted with deletion noise at rate δ, then the algorithm uses*

$$\frac{1}{\varepsilon^2} \cdot \left(\frac{2}{1-\delta}\right)^{\sqrt{n}\cdot(\log n)^{O(\ell)}}$$

many samples and with probability at least 0.99 outputs a hypothesis $\tilde{\mathbf{X}}$ which is supported over at most ℓ strings and has total variation distance at most ε from the unknown target distribution $\mathbf{X}$.

It is easy to see that if the target distribution is promised to be uniform over (a multi-set of) at most ℓ strings, then the algorithm of Theorem 1 can be used to exactly reconstruct the unknown multi-set. As we explain in Section II, while Theorem 1 extends prior results on trace reconstruction (the $\ell = 1$ case), it is proved using very different techniques from recent works [HMPW08], [DOS17a], [NP17], [PZ17], [HPP18], [HHP18] on trace reconstruction.

We note that for deletion rates δ that are bounded away from 1 by a constant, the $2^{O(n^{1/3})}$ sample complexity bounds of [DOS17a], [NP17] for trace reconstruction are better than the $\ell = 1$ case of our result. However, our bounds apply even if the deletion rate δ is very close to 1; in particular, [DOS17a], [NP17] give no results for very high deletion rates $\delta = 1 - o(1/\sqrt{n})$, while Theorem 1 gives a $2^{\tilde{O}(\sqrt{n})}$ bound for $\delta = 1 - 1/2^{\mathrm{polylog}(n)}$ and a $2^{o(n)}$ bound even for δ as large as $1 - 1/2^{\sqrt{n}/\mathrm{polylog}(n)}$. Of course, the main feature of Theorem 1 is that it applies when $\ell > 1$ (unlike [DOS17a], [NP17]).

Negative result. Complementing the sample complexity upper bound, we obtain a lower bound on the sample complexity of population recovery. Our lower bound shows that for a wide range of values of ℓ, at least $n^{\Omega(\ell)}$ samples are required when the population is of size at most ℓ. An informal version of our lower bound is as follows (see Theorem 6 in Section V for a detailed statement):

Theorem 2 (Sample complexity lower bound, informal statement)**.** *Let $0 < \delta < 1$ be any constant deletion probability and suppose that A is an algorithm which, when run on samples drawn from the δ-deletion channel over an arbitrary distribution $\mathbf{X}$ supported over at most $\ell \leq n^{0.499}$ many strings, with probability at least 0.51 outputs a hypothesis distribution $\tilde{\mathbf{X}}$ that has total variation distance at most 0.49 from the unknown target distribution $\mathbf{X}$. Then A must use $n^{\Omega(\ell)}$ many samples.*

II. Our techniques

As noted earlier, our positive result (Theorem 1) gives a sample complexity upper bound for the original ($\ell = 1$) trace reconstruction problem as a special case. We remark that both of the recent $2^{O(n^{1/3})}$ sample complexity upper bounds for the trace reconstruction problem [DOS17a], [NP17], as well as the earlier work of [HMPW08], employed essentially the same algorithmic approach, which is referred to in [DOS17a] as a "mean-based algorithm." At a high level, mean-based algorithms use their samples (traces) only to compute empirical estimates of the n expectations[2]

$$\mathbf{E}_{\boldsymbol{z}\leftarrow\mathrm{Del}_\delta(x)}[\boldsymbol{z}_0],\ \cdots,\mathbf{E}_{\boldsymbol{z}\leftarrow\mathrm{Del}_\delta(x)}[\boldsymbol{z}_{n-1}] \qquad (1)$$

corresponding to the coordinate means of the received traces; they then only use those n estimates to reconstruct the unknown target string x. Both of the algorithms in [DOS17a], [NP17], as well as the algorithm from [HMPW08] for trace reconstruction from an arbitrary string x, are mean-based algorithms. (Both [DOS17a] and [NP17] show that their sample complexity upper bounds are essentially best possible for any mean-based trace reconstruction algorithm.)

While mean-based algorithms have led to state-of-the-art results for trace reconstruction of a single string, this approach breaks down even for the simplest non-trivial cases of population recovery under the deletion channel. Indeed, even when $\ell = 2$ and the unknown distribution $\mathbf{X}$ is promised to be uniform over two strings, it is easy to see that the coordinate means do not provide enough information to recover $\mathbf{X}$. For example, if (x^1, x^2) and (y^1, y^2) are two pairs of strings whose sums (as vectors in $\mathbb{R}^n$) $x^1 + x^2$ and $y^1 + y^2$ are equal (such as $x^1 = 0^n$, $x^2 = 1^n$, $y^1 = 0^{n/2}1^{n/2}$, $y^2 = 1^{n/2}0^{n/2}$), it is easy to see that the coordinate means of received traces will match perfectly:

$$\mathop{\mathbf{E}}_{j\in\{1,2\}}\ \mathop{\mathbf{E}}_{\boldsymbol{z}\leftarrow\mathrm{Del}_\delta(x^j)}[\boldsymbol{z}_i] = \mathop{\mathbf{E}}_{j\in\{1,2\}}\ \mathop{\mathbf{E}}_{\boldsymbol{z}\leftarrow\mathrm{Del}_\delta(y^j)}[\boldsymbol{z}_i],$$

for every $i \in \{0, \ldots, n-1\}$. Thus the mean-based approach of [HMPW08], [DOS17a], [NP17] does not suffice for even the simplest version of the population recovery problem when $\ell = 2$. Indeed, our sample complexity upper bounds are obtained using a completely different approach, which we explain below.

A. Warm-up: A different approach to trace reconstruction (the $\ell = 1$ case)

As a warm-up to our main results, we first give a high-level explanation of how our approach can be used to obtain a simple $2^{\tilde{O}(\sqrt{n})}$-sample algorithm for the trace reconstruction problem. While this is a higher sample complexity than the state-of-the-art mean-based approach of [DOS17a], [NP17] (though our approach does better for very high deletion rates, as noted earlier), our approach has the crucial advantage that it can be adapted to go beyond the $\ell = 1$ case, whereas the mean-based approach cannot handle $\ell > 1$ as described above.

In a nutshell, the essence of our approach is to work with *subsequence frequencies* in the *original string* x (in contrast, note that the mean-based approach uses *single-coordinate frequencies* in the *received traces*). To explain further we introduce some useful terminology: the *k-deck* of a string $x \in \{0,1\}^n$, denoted $\mathsf{D}_k(x)$, is the multiset of all $\binom{n}{k}$ subsequences of x with length exactly k. Thus, the k-deck encapsulates all frequency information about length-k subsequences of x.

A question that arises naturally in the combinatorics of words is the following: what is the smallest value of k (as a function of n) so that for every string $x \in \{0,1\}^n$, the k-deck of x uniquely identifies x? Despite significant investigation dating back to the 1970s [Kal73], this basic quantity is still poorly understood. Improving on earlier $k \leq n/2$ bounds of Kalashnik [Kal73] and Manvel et al. [MMS+91] and a simultaneous $k = O(\sqrt{n \log n})$ bound of Scott [Sco97], Krasikov and Roddity [KR97] showed that $k = O(\sqrt{n})$ suffices. On the lower bounds side, the best lower bound known is $k = 2^{\Omega(\sqrt{\log n})}$, due to Dudík and Schulman [DS03] (improving on earlier $k = \Omega(\log n)$ lower bounds of [MMS+91] and [CK97]).

The relevance of upper bounds on k to the trace reconstruction problem is intuitively clear, and indeed, McGregor et al. [MPV14] observed that if the deletion rate δ is at most $1 - c\sqrt{(\log n)/n}$, then it is trivially easy to extract a random length-$O(\sqrt{n \log n})$ subsequence of x from a typical trace of x. Combining this with the $k = O(\sqrt{n \log n})$ upper bound of Scott [Sco97] and a straightforward sampling-based procedure (which estimates the frequency of each string in $\{0,1\}^k$ to high enough accuracy to determine its exact multiplicity in the k-deck), they obtained an information-theoretic sample complexity upper bound on trace reconstruction: for $\delta \leq 1 - c\sqrt{(\log n)/n}$, at most $n^{O(\sqrt{n \log n})}$ traces suffice to reconstruct any $x \in \{0,1\}^n$ with high probability.

As an initial observation, we slightly strengthen the [MPV14] result by showing that for *any* value of $\delta < 1$, an algorithm which combines sampling and dynamic programming can exactly infer the k-deck of an unknown string $x \in \{0,1\}^n$ with high probability using $(n/(1-\delta))^{O(k)}$ traces from $\mathrm{Del}_\delta(x)$. (See Theorem 4 for a detailed statement and proof of a more general version of this result.) Combining this with the [KR97] upper bound $k = O(\sqrt{n})$, we get that any string x can be reconstructed from δ-deletion noise using $(n/(1-\delta))^{O(\sqrt{n})}$ samples.

The above-outlined approach to trace reconstruction

[2] In this context, the original unknown target string x is viewed as belonging to $\{-1,1\}^n$, and a trace z obtained from $\mathrm{Del}_\delta(x)$ is viewed as a string in $\{-1,1\}^{n'}$ for some $n' \leq n$ with $n - n'$ zeros appended to the end. Throughout the paper, we use $[0 : n-1] = \{0, \ldots, n-1\}$ to index entries of a string of length n.

(the $\ell = 1$ case of population recovery) is the starting point for our main positive result, Theorem 1. In the next subsection we give a high-level description of some of the challenges that arise in dealing with multiple strings and how this work overcomes them.

B. *Ingredients in the proof of Theorem 1*

Recall that in the setting of Theorem 1 the unknown $\mathbf{X}$ is an arbitrary distribution supported on at most ℓ strings $x^1, \dots, x^\ell$ in $\{0,1\}^n$. Viewing $\mathbf{X}$ as a mixture of individual strings, there is a natural notion of the k-deck of $\mathbf{X}$, which we denote by $\mathsf{D}_k(\mathbf{X})$ and which is the weighted multi-set corresponding to the $\mathbf{X}$-mixture of the decks $\mathsf{D}_k(x^1), \dots, \mathsf{D}_k(x^\ell)$.[3] As a result, Theorem 1 will follow if we can show the following: if two distributions $\mathbf{X}, \mathbf{Y}$ over $\{0,1\}^n$ (each supported on at most ℓ strings) have $d_{\mathrm{TV}}(\mathbf{X}, \mathbf{Y}) > \varepsilon$, then for a not-too-large value of k, the k-decks $\mathsf{D}_k(\mathbf{X})$ and $\mathsf{D}_k(\mathbf{Y})$ (note that these are two weighted multi-sets of strings in $\{0,1\}^k$) must be "noticeably different." This is established in Lemma IV.6, which is the technical heart of our upper bound.

To explain our proof of Lemma IV.6 it is useful to revisit the $\ell = 1$ setting; the analogous (and much easier to prove) statement in this context is that given any two strings $x \neq y \in \{0,1\}^n$, the k-decks $\mathsf{D}_k(x)$ and $\mathsf{D}_k(y)$ are not identical when $k \geq C\sqrt{n}$ for some large enough constant C. This is the main result of [KR97] (and a similar statement, with a slightly weaker quantitative bound on k, is also proved in [Sco97]). Since the k-deck in and of itself is somewhat difficult to work with (being a multi-set over $\{0,1\}^k$), both [KR97] and [Sco97] work instead with the *summed k-deck*, which we denote by $\mathsf{SD}_k(x)$ and which is simply the vector in $\mathbb{N}^k$ obtained by summing all $\binom{n}{k}$ elements of the k-deck $\mathsf{D}_k(x)$ (recall that each element of $\mathsf{D}_k(x)$ is a vector in $\{0,1\}^k$). Both [KR97] and [Sco97] actually show that for a suitable value of k, the *summed* k-deck $\mathsf{SD}_k(x)$ uniquely identifies x among all strings in $\{0,1\}^n$. (Both papers also observe that by a simple counting argument, the smallest such k is at least $\tilde{\Omega}(\sqrt{n})$.) The [KR97] proof reduces the analysis of the summed k-deck to an extremal problem about univariate polynomials. The key ingredient of their proof is the following result about univariate polynomials, which was established in [BEK99] in their work on the Prouhet-Tarry-Escott problem:

> Given any nonzero vector $\delta \in \{-1,0,1\}^n$, there is a univariate polynomial p of degree $O(\sqrt{n})$ such that
>
> $$\sum_{0 \leq i < n} \delta_i \cdot p(i) \neq 0. \qquad (\dagger)$$

Setting $\delta = x - y \neq 0$, to finish the proof of $\mathsf{SD}_k(x) \neq \mathsf{SD}_k(y)$ when $x \neq y$ and $k \geq C\sqrt{n}$, [KR97] shows that choosing k to be $\deg(p) + 1$, the inequality ($\dagger$) implies that $\mathsf{SD}_k(x) \neq \mathsf{SD}_k(y)$.

Returning to our ℓ-string setting, we remark that several challenges arise which are not present in the one-string setting. To highlight one of these, due to the difficulty of analyzing the entire k-deck of $\mathbf{X}$ it is natural to try to work with the summed k-deck $\mathsf{SD}_k(\mathbf{X})$ (a non-negative vector in $\mathbb{R}^k$), which is obtained by summing all elements of the weighted multi-set $\mathsf{D}_k(\mathbf{X})$. Indeed it can be shown via a rather straightforward extension of the [KR97] analysis that, when $\mathbf{X}$ is uniform over $x^1, \dots, x^\ell$, the summed k-deck with $k = O(\sqrt{n}\log \ell)$ suffices to exactly reconstruct the *sum* $x^1 + \cdots + x^\ell$ (a vector in $\mathbb{N}^n$). But even for uniform distributions, a difficulty which arises is that the summed k-deck (even with $k = n$) cannot distinguish between two uniform distributions over $x^1, \dots, x^\ell$ versus $y^1, \dots, y^\ell$ that have the same coordinate-wise sums, i.e. that satisfy $x^1 + \cdots + x^\ell = y^1 + \cdots + y^\ell$.[4] Indeed, considering the same example as earlier, in which $\ell = 2$ and $x^1 = 0^n$, $x^2 = 1^n, y^1 = 0^{n/2}1^{n/2}$ and $y^2 = 1^{n/2}0^{n/2}$, the summed k-deck is $(\binom{n}{k}, \dots, \binom{n}{k})/2 \in \mathbb{R}^k$ in both cases.

At a high level our Lemma IV.6 can be viewed as a *robust* generalization of the [KR97] result. A key technical ingredient in its proof is a robust generalization of the [BEK99] result to *multivariate* polynomials. (The summed k-deck corresponds to univariate polynomials, so at a high level our analysis involving multivariate polynomials can be viewed as how we get around the obstacle noted in the previous paragraph.) The proof of Lemma IV.6 consists of three steps which we outline below.

The first conceptual step of our argument is to show that if two support-ℓ distributions $\mathbf{X}$ and $\mathbf{Y}$ over $\{0,1\}^n$ satisfy $d_{\mathrm{TV}}(\mathbf{X}, \mathbf{Y}) \geq \varepsilon$, then there exists a subset $T \subset [0 : n-1]$ of size $d = \lfloor \log(2\ell) \rfloor$ such that $\mathbf{X}$ and $\mathbf{Y}$ "differ significantly" just on the coordinates in T. In particular, there is some $|T|$-bit string c such

[3] By a weighted-multiset we mean a multiset in which each element has a weight. Alternatively, one can interpret (after normalization) $\mathsf{D}_k(x)$ as a probability distribution over the 2^k strings in $\{0,1\}^k$ and in this case, $\mathsf{D}_k(\mathbf{X})$ can be viewed as a probability distribution that is the $\mathbf{X}$-mixture of $\mathsf{D}_k(x^1), \dots, \mathsf{D}_k(x^\ell)$.

[4] This is conceptually similar to the inability of mean-based algorithms to handle multiple strings noted earlier.

that $\mathbf{Pr}_{\boldsymbol{x}\sim\mathbf{X}}[\boldsymbol{x}_T = c]$ is significantly different from $\mathbf{Pr}_{\boldsymbol{y}\sim\mathbf{Y}}[\boldsymbol{y}_T = c]$, where we use x_T to denote the restriction of a string $x \in \{0,1\}^n$ on coordinates in T. (This is made precise in Lemma IV.1.) Let $\Delta : \binom{[0:n-1]}{d} \to \mathbb{R}$ be the following function over size-d subsets of $[0 : n-1]$:

$$\Delta(S) = \mathbf{Pr}_{\boldsymbol{x}\sim\mathbf{X}}\big[\boldsymbol{x}_S = c\big] - \mathbf{Pr}_{\boldsymbol{y}\sim\mathbf{Y}}\big[\boldsymbol{y}_S = c\big]. \qquad (2)$$

Then Lemma IV.1 implies that $\|\Delta\|_\infty$ is not too small.

The second (and central) conceptual step of our argument can be viewed as a robust generalization of the [BEK99] result to d-variate polynomials, as alluded to earlier. The key result giving this step, Lemma IV.7, roughly speaking states the following:

> Given the Δ as defined in (2), there is a d-variate polynomial ϕ of not-too-high degree (roughly $\sqrt{n}$) such that[5]
>
> $$\left|\sum_{0\le t_1<\cdots<t_d<n} \phi(t_1,\ldots,t_d)\cdot\Delta\big(\{t_1,\ldots,t_d\}\big)\right| \qquad (\dagger\dagger)$$
>
> can be lower bounded in terms of $\|\Delta\|_\infty$, which is not too small by Lemma IV.1.

The third conceptual step relates (††) to the distance between the k-decks $\mathsf{D}_k(\mathbf{X})$ and $\mathsf{D}_k(\mathbf{Y})$, by showing that if (††) is not too small then $\mathsf{D}_k(\mathbf{X})$ and $\mathsf{D}_k(\mathbf{Y})$ must be "noticeably different" when k is chosen to be $\deg(\phi) + d$. We refer the reader to Lemma IV.8. At a high level this is analogous to, but technically more involved than, the [KR97] proof that the inequality (†) for $\delta = x - y$ implies that $\mathsf{SD}_k(x) \neq \mathsf{SD}_k(y)$ with $k = \deg(p)+1$. Lemma IV.6 then follows by combining all three steps, i.e. $d_{\mathrm{TV}}(\mathbf{X},\mathbf{Y})$ being large implies that $\mathsf{D}_k(\mathbf{X})$ is "noticeably different" from $\mathsf{D}_k(\mathbf{Y})$ for k that is roughly $\sqrt{n}$. Below we outline the main ingredients needed in the second step.

In the search for a low-degree polynomial ϕ such that the sum in (††) has large magnitude, it is natural to define $\phi(t_1,\ldots,t_d)$ by first projecting $(t_1,\ldots,t_d)$ to a line and then applying a univariate polynomial similar to the p used in (†). To make this more precise, we will look for ϕ of the form

$$\phi(t_1,\ldots,t_d) = f\big(w_1t_1 + \cdots + w_dt_d\big), \qquad (3)$$

where $w_1,\ldots,w_d$ are positive integers (so the line is along the direction $w = (w_1,\ldots,w_d)$) and f is a low-degree univariate polynomial to be specified later. With (3), we rewrite the sum in (††) as

$$\sum_{0\le t_1<\cdots<t_d<n} \phi(t_1,\ldots,t_d)\cdot\Delta\big(\{t_1,\ldots,t_d\}\big) = \sum_{b=0}^{nd\,\|w\|_\infty} f(b)\cdot\Gamma(b), \qquad (4)$$

where $\Gamma(b)$ is the sum of $\Delta(T)$ over all d-subsets $T = \{t_1,\ldots,t_d\}$ such that $0 \le t_1 < \cdots < t_d < n$ and $b = w_1t_1 + \cdots + w_dt_d$. Comparing (4) with (†), our goal would follow directly from the [BEK99] result by choosing f to be p if Γ is nonzero and takes values in $\{-1,0,1\}$ (or even $\{-c,0,c\}$ for some not too small $c > 0$). However, the main difficulty we encounter is that Γ is much more complex than the $\{-1,0,1\}^n$ vectors that can be handled by techniques of [BEK99]; for example, Γ in general may contain a large number (depending on n) of distinct values.

There are three ingredients we use in choosing $w_1,\ldots,w_d$ and f to overcome this difficulty:

(A) We first observe that Δ has a combinatorial "rectangular" structure, which implies that the support of Δ can be partitioned into a small number of sets $\mathcal{S}_1, \mathcal{S}_2, \ldots$ (each element of $\mathcal{S}_a$ is a size-d subset of $[0 : n-1]$) such that all $T \in \mathcal{S}_a$ share the same value of $\Delta(T)$ and there is a set $T_a \in \mathcal{S}_a$ that is *dominated*[6] by every $T \in \mathcal{S}_a$. We refer to T_a as the *anchor set* of $\mathcal{S}_a$. This is made precise in Lemma IV.3. Moreover, we show in Lemma IV.4 that the collections $\mathcal{S}_a$ can be divided into an *even smaller* number of groups such that, for any $\mathcal{S}_a, \mathcal{S}_{a'}$ that belong to the same group, the ratio of $|\Delta(T_a)|$ and $|\Delta(T_{a'})|$ is bounded from above by a small number.

(B) Next we observe that when $w_1,\ldots,w_d$ are drawn from a suitable distribution, all anchor sets in (A) have distinct images after the projection. (See Claim IV.9.) We fix such a tuple $(w_1,\ldots,w_d)$. (A) and (B) together are then used to obtain (see Lemma IV.11) a strong structural characterization of Γ.

(C) Finally we define a new univariate polynomial f based on Chebyshev polynomials and the construction of p in [BEK99]. (See Lemma IV.10.) The characterization of Γ and properties of f are then combined to finish the proof by showing that

[5] The reader who has peeked ahead to the statement of Lemma IV.7 may have noticed that the lemma statement also bounds the magnitudes of coefficients of the polynomial ϕ. This is done for technical reasons, and we skip these technical details in the high-level description here.

[6] Given two size-d subsets $S = \{s_1,\ldots,s_d\}$ and $T = \{t_1,\ldots,t_d\}$ of $[0 : n-1]$ with $s_1 < \cdots < s_d$ and $t_1 < \cdots < t_d$, we say that S is dominated by T if $s_i \le t_i$ for all i.

the sum in (††) has not too small magnitude when we apply the polynomial ϕ given in (3).

C. Our lower bounds

We begin by recalling the $\Omega(n)$ lower bound of McGregor et al. [MPV14]. This lower bound is obtained via a simple analysis of the two distributions of traces resulting from the two strings $x^1 = 0^{n/2}10^{n/2-1}$ and $x^2 = 0^{n/2-1}10^{n/2}$. The starting point of the [MPV14] analysis is the observation that under the δ-deletion channel, conditioned on the sole "1" coordinate being retained, the distribution of a trace of x^1 corresponds to $(\boldsymbol{a}, \boldsymbol{b})$ where $\boldsymbol{a}$ and $\boldsymbol{b}$ are independent draws from $\mathrm{Bin}(n/2, 1-\delta)$ and $\mathrm{Bin}(n/2-1, 1-\delta)$ respectively, whereas the distribution of a trace of x^2 corresponds to $(\boldsymbol{b}, \boldsymbol{a})$. [MPV14] used this to show that the squared Hellinger distance between these two distributions of traces is $O(1/n)$, and in turn use this squared Hellinger distance bound to infer an $\Omega(n)$ sample complexity lower bound for determining whether a collection of received traces came from x^1 or from x^2.

Our lower bound approach may be viewed as an extension of the [MPV14] lower bound to *mixtures* of distributions similar to the ones they consider. The high-level idea of our lower bound proof is as follows: we show that there exist two distributions $\mathbf{X}, \mathbf{Y}$ over $\{0,1\}^n$ (in fact, over n-bit strings with precisely one 1) which have disjoint supports, each of size at most 2ℓ, but are such that the total variation distance $d_{\mathrm{TV}}(\mathrm{Del}_\delta(\mathbf{X}), \mathrm{Del}_\delta(\mathbf{Y}))$, between traces of strings drawn from $\mathbf{X}$ versus traces of strings drawn from $\mathbf{Y}$, is very small. This is easily seen to imply Theorem 2.

For simplicity in introducing the main ideas of our analysis, in this expository overview we will first consider an "$n = +\infty$" version of our population recovery scenario. We begin by considering the distribution $\mathrm{Del}_\delta(\tilde{e}_{m+i})$ where m is some fixed value and $\tilde{e}_{m+i}$ is an infinite string with a single 1 in position $m+i$ and all other coordinates 0. A δ fraction of the outcomes of $\mathrm{Del}_\delta(\tilde{e}_{m+i})$ are the infinite all-0 string, which conveys no information. The other $1-\delta$ fraction of the outcomes each have precisely one 1, occurring in position $1+\boldsymbol{a}$ where $\boldsymbol{a}$ is distributed according to the binomial distribution $\mathrm{Bin}(m+i, 1-\delta)$. In this infinite-$n$ setting, two distributions $\mathbf{X}, \mathbf{Y}$ over strings of the form $\tilde{e}_{m+i}$ with disjoint supports correspond to two mixtures of distinct binomial distributions (all with second parameter $1-\delta$, but with a set of first parameters in the first mixture that is disjoint from the set of first parameters in the second mixture). The animating idea behind our construction and analysis is that it is possible for two distinct mixtures of binomials like this to be very close to each other in total variation distance.[7]

In order to show that two distinct mixtures of binomial distributions as described above can be very close to each other in total variation distance, our lower bounds employ technical machinery due to Roos [Roo00] and Daskalakis and Papadimitriou [DP15]. Roos [Roo00] developed a "Krawtchouk expansion" which provides an *exact* expression for the probability that a Poisson binomial distribution (a sum of n independent Bernoulli random variables with expectations $p_1, \ldots, p_n$) puts on any given outcome in $\{0, 1, \cdots, n\}$. Daskalakis and Papadimitriou [DP15] used Roos's Krawtchouk expansion to show that under mild technical conditions, low-order moments of any Poisson binomial distribution essentially determine the entire distribution. In more detail, their main result is that if $\mathbf{X}, \mathbf{Y}$ are two Poisson binomial distributions (satisfying mild technical conditions) whose t-th moments match, i.e. $\mathbf{E}[\mathbf{X}^t] = \mathbf{E}[\mathbf{Y}^t]$ for $t = 1, \ldots, O(\log(1/\varepsilon))$, then the total variation distance between $\mathbf{X}$ and $\mathbf{Y}$ is at most ε.

Our analysis proceeds in two main steps. In the first step, we show that there exist two mixtures of pairs of binomial distributions, which we denote by $\mathbf{D}_S$ and $\mathbf{D}_T$, with certain desirable properties. S and T are both subsets of $\{0, \ldots, 2\ell\}$, and $\mathbf{D}_S$ is a certain mixture of pairs of binomial distributions $(\mathrm{Bin}(n/2+i, 1-\delta), \mathrm{Bin}(n/2-i, 1-\delta))$ for $i \in S$ while $\mathbf{D}_T$ is a certain mixture of pairs of binomial distributions $(\mathrm{Bin}(n/2+j, 1-\delta), \mathrm{Bin}(n/2-j, 1-\delta))$ for $j \in T$. We establish the existence of *disjoint* sets S, T such that the resulting mixtures $\mathbf{D}_S$ and $\mathbf{D}_T$ have matching t-th moments for all $t = 1, \ldots, \ell$. This is proved using known algebraic expressions for the moments of binomial distributions and simple linear algebraic arguments. In the second main step, we extend the analysis of Daskalakis and Papadimitriou [DP15] and apply this extension to our setting, in which we are dealing with mixtures of (pairs of) binomial distributions (as opposed to their and Roos's setting of Poisson binomial distributions). We show that the matching first ℓ moments of $\mathbf{D}_S$ and $\mathbf{D}_T$ imply that the distributions $\mathrm{Del}_\delta(\mathbf{X})$ and $\mathrm{Del}_\delta(\mathbf{Y})$

[7] We remark that our actual scenario is more complicated than this idealized version because n is a finite value rather than $+\infty$. For $n = 2m+1$, this means that a received trace $0^{\boldsymbol{a}}10^{\boldsymbol{b}}$ which contains a 1 and came from $\mathrm{Del}_\delta(e_{m+i})$ provides a pair of values $(\boldsymbol{a}, \boldsymbol{b})$ where $\boldsymbol{a}$ is distributed according to $\mathrm{Bin}(m+i, \rho)$ and $\boldsymbol{b}$ is independently distributed according to $\mathrm{Bin}(m-i, \rho)$ where $\rho = 1-\delta$ is the retention probability. This second value $\boldsymbol{b}$ provides additional information which is not present in the $n = +\infty$ version of the problem, and this makes it more challenging and more technically involved to prove a lower bound. We deal with these issues in Section V-B.

are very close, where $\mathbf{X}$ corresponds to the mixture of Hamming-weight-one strings in $\{0,1\}^n$ corresponding to $\mathbf{D}_S$ and $\mathbf{Y}$ likewise corresponds to the mixture of Hamming-weight-one strings corresponding to $\mathbf{D}_T$. (In fact, in our setting having ℓ matching moments leads to $n^{-\Omega(\ell)}$-closeness in total variation distance, whereas in [DP15] the resulting closeness from ℓ matching moments was $2^{-\Omega(\ell)}$.)

We close this subsection by observing that while the results of [Roo00], [DP15] were used in a crucial way in subsequent work of Daskalakis et al. [DDS15] to obtain a sample complexity *upper bound* on learning Poisson binomial distributions, in our context we use these results to obtain a sample complexity *lower bound* for population recovery. Intuitively, the difference is that in the [DDS15] scenario of learning an unknown Poisson binomial distribution, there is no noise process affecting the samples: the learning algorithm is assumed to directly receive draws from the underlying Poisson binomial distribution being learned. In such a noise-free setting, the existence of a small ε-cover for the space of all Poisson binomial distributions (which is established in [DP15] as a consequence of their moment-matching result) means, at least on a conceptual level, that a learning algorithm "need only search a small space of candidates" to find a high-accuracy hypothesis. In contrast, in our context of deletion-channel noise, our arguments show that it is possible for two underlying true distributions $\mathbf{X}, \mathbf{Y}$ over $\{0,1\}^n$ to be very different (indeed, to have disjoint supports) but to be such that their deletion-noise-corrupted versions have low-order moments which match each other exactly. In this scenario, the [Roo00], [DP15] results can be used to show that the variation distance between the two distributions of noisy samples received by the learner is very small, and this gives a sample complexity lower bound for distinguishing $\mathbf{X}$ and $\mathbf{Y}$ on the basis of such noisy samples.

III. Preliminaries

Notation. Given a nonnegative integer n, we write $[n]$ to denote $\{1,\dots,n\}$. Given integers $a \le b$ we write $[a:b]$ to denote $\{a,\dots,b\}$. It will be convenient for us to index a binary string $x \in \{0,1\}^n$ using $[0:n-1]$ as $x = (x_0,\dots,x_{n-1})$. Given a vector $v = (v_1,\dots,v_d) \in \mathbb{R}^d$, we write $\|v\|_\infty$ to denote $\max_{i\in[d]} |v_i|$. Given a function $\Delta : A \to \mathbb{R}$ over a finite domain A, we write $\|\Delta\|_\infty = \max_{a\in A} |\Delta(a)|$. Given a polynomial p (which may be univariate or multivariate), we write $\|p\|_1$ to denote the sum of magnitudes of p's coefficients. All logarithms and exponents are binary (base 2) unless otherwise specified.

Distributions. We use bold font letters to denote probability distributions and random variables, which should be clear from the context. We write "$\boldsymbol{x} \sim \mathbf{X}$" to indicate that random variable $\boldsymbol{x}$ is distributed according to distribution $\mathbf{X}$. The total variation distance between two distributions $\mathbf{X}$ and $\tilde{\mathbf{X}}$ over a finite set $\mathcal{X}$ is defined as

$$d_{\mathrm{TV}}(\mathbf{X}, \tilde{\mathbf{X}}) = \frac{1}{2}\sum_{x\in\mathcal{X}} \big|\mathbf{X}(x) - \tilde{\mathbf{X}}(x)\big|,$$

where $\mathbf{X}(x)$ denotes the amount of probability mass that the distribution $\mathbf{X}$ puts on outcome x.

Population recovery from the deletion channel. Throughout this paper the parameter $0 < \delta < 1$ denotes the *deletion probability*. Given a string $x \in \{0,1\}^n$, we write $\mathrm{Del}_\delta(x)$ to denote the distribution of a random trace of x after it has been passed through the δ-deletion channel (so the distribution $\mathrm{Del}_\delta(x)$ is supported on $\{0,1\}^{\le n}$). Recall that a random trace $\boldsymbol{y} \sim \mathrm{Del}_\delta(x)$ is obtained by independently deleting each bit of x with probability δ and concatenating the surviving bits.[8]

We now define the problem of population recovery from the deletion channel that we will study in this paper. In this problem the goal is to learn an unknown *target distribution* $\mathbf{X}$ supported on at most ℓ strings from $\{0,1\}^n$. The learning algorithm has access to independent samples, each of which is generated independently by first drawing a string $\boldsymbol{x} \sim \mathbf{X}$ and then outputting a trace from $\mathrm{Del}_\delta(\boldsymbol{x})$. For conciseness we write $\mathrm{Del}_\delta(\mathbf{X})$ to denote this distribution. The goal for the learning algorithm is to output with high probability (say at least 0.99) a *hypothesis distribution* $\tilde{\mathbf{X}}$ for $\mathbf{X}$ which is ε-accurate in total variation distance: $d_{\mathrm{TV}}(\mathbf{X}, \tilde{\mathbf{X}}) \le \varepsilon$. We are interested in the number of samples needed for this learning task in terms of n, ℓ, ε and δ.

Decks. Given a subset $T = \{t_1,\dots,t_k\} \subseteq [0:n-1]$ of size k with $t_1 < \cdots < t_k$, and two strings $v \in \{0,1\}^k$, $x \in \{0,1\}^n$, we say that v *matches* x *at* T if $x_T = v$, where $x_T = (x_{t_1},\dots,x_{t_k}) \in \{0,1\}^k$ denotes the string x restricted to positions in T. We say that the *number of occurrences of* v *in* x is the number of size-k subsets $T \subseteq [0:n-1]$ such that v matches x at T, and we write $\#(v,x)$ to denote this quantity. Given a distribution $\mathbf{X}$

[8] For simplicity in this work we assume that the deletion probability δ is known to the learning algorithm. We note that it is possible to obtain a high-accuracy estimate of δ simply by measuring the average length of traces received from the deletion channel.

over $\{0,1\}^n$, we write $\#(v, \mathbf{X})$ to denote the expected number of occurrences of v in $\boldsymbol{x} \sim \mathbf{X}$, i.e.

$$\#(v, \mathbf{X}) = \mathop{\mathbf{E}}_{\boldsymbol{x} \sim \mathbf{X}} \big[\#(v, \boldsymbol{x})\big].$$

Given a string $x \in \{0,1\}^n$, we write $\mathsf{D}_k(x)$ to denote the (*normalized*[9]) *k-deck* of x. This is a 2^k-dimensional vector indexed by strings $v \in \{0,1\}^k$ such that

$$\big(\mathsf{D}_k(x)\big)_v = \frac{\#(v,x)}{\binom{n}{k}}.$$

So $\mathsf{D}_k(x)$ is a nonnegative vector that sums to 1. Similarly, for a distribution $\mathbf{X}$ over strings from $\{0,1\}^n$, we write $\mathsf{D}_k(\mathbf{X})$ to denote the (*normalized*[10]) *k-deck* of $\mathbf{X}$, given by

$$\big(\mathsf{D}_k(\mathbf{X})\big)_v = \frac{\#(v,\mathbf{X})}{\binom{n}{k}},$$

for each $v \in \{0,1\}^k$. So $\mathsf{D}_k(\mathbf{X})$ is also a 2^k-dimensional nonnegative vector that sums to 1.

IV. Upper Bounds for Distributions Supported on at Most ℓ Strings

Our goal is to prove Theorem 3, which is restated below:

Theorem 3. *There is an algorithm A which has the following performance guarantee: For any distribution $\mathbf{X}$ supported over at most ℓ strings in $\{0,1\}^n$, if A is given*

$$\frac{1}{\varepsilon^2} \cdot \left(\frac{2}{1-\delta}\right)^{\sqrt{n}\cdot(\log n)^{O(\ell)}} \tag{5}$$

many samples from $\mathrm{Del}_\delta(\mathbf{X})$, then with probability at least 0.99 the algorithm outputs a probability distribution $\tilde{\mathbf{X}}$ supported over at most ℓ strings such that $d_{\mathrm{TV}}(\mathbf{X}, \tilde{\mathbf{X}}) \le \varepsilon$.

In Section IV-A we introduce the notion of a *restriction*, which is a "local view" of a distribution $\mathbf{X}$ confined to a specific subset of coordinates and a specific outcome for those coordinates. We then provide some terminology and prove three useful lemmas about restrictions in Section IV-A. Next in Section IV-B we describe the algorithm A, state our main technical lemma, Lemma IV.6, and use it to prove the correctness of algorithm A. We prove Lemma IV.6 in Sections IV-C and IV-D.

Notational convention. Our argument below involves many integer-valued index variables which take values in a range of different intervals. To help the reader keep track, we will use the following convention (the values L and m will be defined later):

- $s, t, s_1, t_1, \ldots$ will denote an index ranging over $[0 : n-1]$;
- $j, j_1, \ldots$ will denote an index ranging over $[0 : k-1]$;
- $a, a', a_1, \ldots$ will denote an index ranging over $[L]$;
- $b, b', b_1, \ldots$ will denote an index ranging over $[0 : m]$;
- $i, i_1, \ldots,\ \alpha, \alpha_1, \ldots$ and $\beta, \beta_1, \ldots$ will denote an index in all other places.

A. Restrictions

Let $\mathbf{X}$ be a distribution over strings from $\{0,1\}^n$ and let $d \in [n]$ be a parameter (which should be thought of as quite small; we will set $d = O(\log \ell)$ below). Given a size-d subset $T = \{t_1, \ldots, t_d\}$ of $[0 : n-1]$ with $0 \le t_1 < \cdots < t_d < n$ and a string $c \in \{0,1\}^d$, we define

$$\mathrm{restrict}(\mathbf{X}, T, c) := \mathbf{Pr}_{\boldsymbol{x} \sim \mathbf{X}} \big[(\boldsymbol{x}_{t_1}, \ldots, \boldsymbol{x}_{t_d}) = c\big],$$

the probability that a draw of $\boldsymbol{x} \sim \mathbf{X}$ matches c in the coordinates of T.

Let $\mathbf{X}$ and $\mathbf{Y}$ be two distributions, each supported over at most ℓ strings from $\{0,1\}^n$. Our first lemma shows that if $d_{\mathrm{TV}}(\mathbf{X}, \mathbf{Y})$ is large, then there are a size-d subset T and a string $c \in \{0,1\}^d$ with $d = \lfloor \log(2\ell) \rfloor$ such that there is a reasonably big gap between $\mathrm{restrict}(\mathbf{X}, T, c)$ and $\mathrm{restrict}(\mathbf{Y}, T, c)$.

Lemma IV.1. *Let $\mathbf{X}$ and $\mathbf{Y}$ be two distributions, each supported over at most ℓ strings from $\{0,1\}^n$. Then there exist a size-d subset T of $[0 : n-1]$ and a string $c \in \{0,1\}^d$ with $d = \lfloor \log(2\ell) \rfloor$ such that*

$$\Big|\mathrm{restrict}(\mathbf{X}, T, c) - \mathrm{restrict}(\mathbf{Y}, T, c)\Big| \ge \frac{d_{\mathrm{TV}}(\mathbf{X}, \mathbf{Y})}{\ell^{O(\ell)}}.$$

Proof. Let $\mathrm{supp}(\mathbf{X}) \cup \mathrm{supp}(\mathbf{Y}) = \{z^1, \ldots, z^{\ell'}\}$ for some $\ell' \le 2\ell$. For each $i \in [\ell']$, let $p_i \ge 0$ be the magnitude of the difference between the probabilities of z^i in $\mathbf{X}$ and in $\mathbf{Y}$. Let $\varepsilon = d_{\mathrm{TV}}(\mathbf{X}, \mathbf{Y})$. Then by definition we have $\sum_i p_i = 2\varepsilon$. Without loss of generality we assume that $p_1 \ge \cdots \ge p_{\ell'} \ge 0$ and prove the following claim (where we set $p_{\ell'+1} = 0$ by default for convenience):

Claim IV.2. *There exists an $i^* \in [\ell']$ such that $p_{i^*} \ge \varepsilon/(4\ell)^{\ell'}$ and $p_{i^*+1} \le p_{i^*}/(4\ell)$.*

[9] It will be more convenient for us to use the notion of (normalized) k-decks defined here; note that we can recover from it the multi-set of all subsequences of x with length k, and vice versa.

[10] Similarly, the (normalized) k-deck here is equivalent to the weighted multi-set version used in the introduction up to a simple rescaling.

Proof. First we notice that $p_1 \geq \varepsilon/\ell$ given that $\sum_i p_i = 2\varepsilon$ and $\ell' \leq 2\ell$. Now given that the p_i's are nonnegative, there exists an $i \in [\ell']$ (e.g., by taking $i = \ell'$) such that $p_{i+1} \leq p_i/(4\ell)$. Take i^* to be the smallest such index i. Then we have

$$\frac{p_{i^*}}{p_1} = \frac{p_{i^*}}{p_{i^*-1}} \cdots \frac{p_2}{p_1} > \frac{1}{(4\ell)^{i^*-1}}$$

by the choice of i^* as the smallest such index. As a result, we have

$$p_{i^*} \geq \frac{\varepsilon}{(4\ell)^{i^*}} \geq \frac{\varepsilon}{(4\ell)^{\ell'}}.$$

This finishes the proof of the claim. □

Let $i^* \in [\ell']$ be the integer given by the claim above, and we consider the first i^* strings $z^1, \ldots, z^{i^*}$. Given that $i^* \leq \ell' \leq 2\ell$, there exist a d-subset T of $[0 : n-1]$ with $d = \lfloor \log(2\ell) \rfloor$, a string $c \in \{0,1\}^d$ and an $i' \leq i^*$ such that the restriction of $z^{i'}$ matches c but the restriction of z^i does not match c for any other $i \leq i^*$. (This can be achieved by repeatedly selecting a coordinate that splits the remaining strings into two nonempty subsets and setting c to reduce the size by at least half each time.) Using properties of i^* given in the claim above, we have

$$\begin{aligned}&\Big|\mathrm{restrict}(\mathbf{X}, T, c) - \mathrm{restrict}(\mathbf{Y}, T, c)\Big|\\ &\quad\geq p_{i^*} - \sum_{i > i^*} p_i \geq p_{i^*} - 2\ell \cdot \frac{p_{i^*}}{4\ell} = \frac{p_{i^*}}{2} \geq \frac{\varepsilon}{\ell^{O(\ell)}}.\end{aligned}$$

This finishes the proof of the lemma. □

Given two size-d subsets $S = \{s_1, \ldots, s_d\}$ and $T = \{t_1, \ldots, t_d\}$ of $[0 : n-1]$ with $s_1 < \cdots < s_d$ and $t_1 < \cdots < t_d$, we say that S is *dominated* by T if $s_i \leq t_i$ for every $i \in [d]$. Let $\Delta : \binom{[0:n-1]}{d} \to \mathbb{R}$ be a function over size-d subsets of $[0 : n-1]$. We use $\mathrm{supp}(\Delta)$ to denote the set of subsets T with $\Delta(T) \neq 0$. We need the following definitions of a *cover* and a *group cover* of such a function Δ.

Definition 1 (Covers and group covers)**.** *We say that a function* $\Delta : \binom{[0:n-1]}{d} \to \mathbb{R}$ *has an* L-cover $\{(T_a, \mathcal{S}_a) : a \in [L]\}$ *for some* $L \geq 0$ *if*

1) $\mathcal{S}_1, \ldots, \mathcal{S}_L$ *form an* L*-way partition of* $\mathrm{supp}(\Delta)$*;*
2) $T_a \in \mathcal{S}_a$ *for each* $a \in [L]$*;*
3) $\Delta(T) = \Delta(T_a)$ *for every* $T \in \mathcal{S}_a$*; and*
4) T_a *is dominated by every* $T \in \mathcal{S}_a$*.*

We refer to the set T_a *as the* anchor set *of the collection* $\mathcal{S}_a$*.*

Furthermore we say that Δ *has an* (L, q, λ)-group cover *if* Δ *has an* L*-cover* $\{(T_a, \mathcal{S}_a) : a \in [L]\}$ *and a* q*-way partition of* $[L]$ *into* $A_1, \ldots, A_q$ *such that for each* $i \in [q]$*, for all* $a, a' \in A_i$ *we have*

$$\frac{|\Delta(T_a)|}{|\Delta(T_{a'})|} \leq \lambda.$$

Given distributions $\mathbf{X}$ and $\mathbf{Y}$ over strings from $\{0,1\}^n$ and a string $c \in \{0,1\}^d$, we write $\Delta_{\mathbf{X},\mathbf{Y},c}$ to denote the function over size-d subsets of $[0 : n-1]$ that maps a size-d subset T to

$$\Delta_{\mathbf{X},\mathbf{Y},c}(T) := \mathrm{restrict}(\mathbf{X}, T, c) - \mathrm{restrict}(\mathbf{Y}, T, c).$$

The second lemma shows that when d and the supports of $\mathbf{X}, \mathbf{Y}$ are small, the function $\Delta_{\mathbf{X},\mathbf{Y},c}$ has a small cover for any string $c \in \{0,1\}^d$. Taking as an example when $\ell = d = 2$ and $\mathrm{supp}(\mathbf{X}) = \{x^1, x^2\}$, we have that $\mathrm{restrict}(\mathbf{X}, S, c) = \mathrm{restrict}(\mathbf{X}, T, c)$ if $x^1_S = x^1_T$ and $x^2_S = x^2_T$ (note that this is a sufficient but not necessary condition in general). Letting $S = \{s_1, s_2\}$ for some $s_1 < s_2$ and $T = \{t_1, t_2\}$ for some $t_1 < t_2$, this condition can be written equivalently as

$$(x^1_{s_1}, x^2_{s_1}) = (x^1_{t_1}, x^2_{t_1}) \quad \text{and} \quad (x^1_{s_2}, x^2_{s_2}) = (x^1_{t_2}, x^2_{t_2}).$$

This implies that $\mathrm{restrict}(\mathbf{X}, \cdot, c)$, as a function over size-2 subsets, has the following combinatorial "rectangular" structure: one can partition indices $t \in [0 : n-1]$ into four types 00,01,10,11 according to values of x^1_t and x^2_t; this induces a partition of all size-2 subsets into 16 "rectangles,"[11] where $S = \{s_1 < s_2\}$ and $T = \{t_1 < t_2\}$ belong to the same "rectangle" iff the type of s_1 is the same as that of t_1 and the type of s_2 is the same as that of t_2. It follows that all T in the same "rectangle" share the same value $\mathrm{restrict}(\mathbf{X}, T, c)$. We use this observation to obtain a small cover for $\Delta_{\mathbf{X},\mathbf{Y},c}$.

Lemma IV.3. *Let* $\mathbf{X}$ *and* $\mathbf{Y}$ *be two distributions, each supported over at most* ℓ *strings from* $\{0,1\}^n$*. For any* $d \in [n]$ *and any string* $c \in \{0,1\}^d$*,* $\Delta_{\mathbf{X},\mathbf{Y},c}$ *has an* L*-cover for some* $L \leq 2^{2d\ell}$*.*

Proof. Suppose that $\mathbf{X}$ is supported on $x^1, \ldots, x^{\ell'}$ and $\mathbf{Y}$ is supported on $y^1, \ldots, y^{\ell''}$ with $\ell', \ell'' \leq \ell$. We say an index $t \in [0 : n-1]$ is of *type*-(u, v), where $u \in \{0,1\}^{\ell'}$ and $v \in \{0,1\}^{\ell''}$, if

$$(x^1_i, \ldots, x^{\ell'}_i) = u \quad \text{and} \quad (y^1_i, \ldots, y^{\ell''}_i) = v.$$

This allows us to classify size-d subsets of $[0 : n-1]$ into at most $(2^{\ell'+\ell''})^d \leq 2^{2d\ell}$ many equivalence classes: $S \sim T$ if $S = \{s_1, \ldots, s_d\}$ with $s_1 < \cdots < s_d$ and

[11] Strictly speaking, these are not rectangles since we always need to order indices of a subset in ascending order.

$T = \{t_1, \ldots, t_d\}$ with $t_1 < \cdots < t_d$ are such that s_i and t_i are of the same type for all $i \in [d]$.

Let $\mathcal{S}_a$ be a nonempty equivalence class of $\sim$ such that $S = \{s_1, \ldots, s_d\} \in \mathcal{S}_a$ if $s_1 < \cdots < s_d$ and s_i has type-$(u^{(i)}, v^{(i)})$ for each $i \in [d]$. It follows from the definition of $\sim$ that all $S \in \mathcal{S}_a$ have the same $\mathrm{restrict}(\mathbf{X}, S, c)$ and $\mathrm{restrict}(\mathbf{Y}, S, c)$, and hence the same value of $\Delta_{\mathbf{X},\mathbf{Y},c}(S)$. Moreover, we let $T_a = \{t_1, \ldots, t_d\}$ be the following set: t_1 is the smallest index of type-$(u^{(1)}, v^{(1)})$ and for each i from 2 to d, t_i is the smallest index that is larger than t_{i-1} and has type-$(u^{(i)}, v^{(i)})$. Because $\mathcal{S}_a$ is nonempty, T_a is well defined and it is easy to verify that T_a is dominated by every $S \in \mathcal{S}_a$. As a result, $\Delta_{\mathbf{X},\mathbf{Y},c}$ has the following L-cover:

$$\big\{(T_a, \mathcal{S}_a) : \mathcal{S}_a \text{ is nonempty and } \Delta_{\mathbf{X},\mathbf{Y},c}(T_a) \neq 0\big\},$$

for some $L \leq 2^{2d\ell}$. This finishes the proof of the lemma. □

The last lemma shows that the function $\Delta_{\mathbf{X},\mathbf{Y},c}$ actually has an (L, q, λ)-group cover, for some parameters $L \leq 2^{2d\ell}$, $q \leq \ell$ and $\lambda \leq \ell^{O(\ell)}$.

Lemma IV.4. *Let $\mathbf{X}$ and $\mathbf{Y}$ be two distributions, each supported over at most ℓ strings from $\{0,1\}^n$. For any $d \in [n]$ and $c \in \{0,1\}^d$, $\Delta_{\mathbf{X},\mathbf{Y},c}$ has an $(L, q, \ell^{O(\ell)})$-group cover for some $L \leq 2^{2d\ell}$ and $q \leq \ell$.*

Proof. First we apply Lemma IV.3 to obtain an L-cover $\{(T_a, \mathcal{S}_a) : a \in [L]\}$ of $\Delta := \Delta_{\mathbf{X},\mathbf{Y},c}$ for some $L \leq 2^{2d\ell}$. It suffices to show that the L positive numbers $|\Delta(T_a)|$, $a \in [L]$, can be divided into at most ℓ groups such that any two in the same group have the ratio bounded from above by $\ell^{O(\ell)}$.

Let $p_1, \ldots, p_{\ell'} > 0$ be probabilities of strings in $\mathbf{X}$ for some $\ell' \leq \ell$ and $q_1, \ldots, q_{\ell''} > 0$ be probabilities of strings in $\mathbf{Y}$ for some $\ell'' \leq \ell$. The observation is that every number $|\Delta(T_a)|$ is a linear form over the p_i's and q_i's with coefficients $-1, 0$ or 1. This motivates the following claim:

Claim IV.5. *Let $u_1, \ldots, u_g > 0$ be g* (not necessarily distinct) *positive numbers. Let V be the set of all positive values v of the form $v = c_1 u_1 + \cdots + c_g u_g$ for some $c_1, \ldots, c_g \in \{-1, 0, 1\}$. Then there cannot exist $g+1$ numbers $v_1, \ldots, v_{g+1}$ in V satisfying $v_{g+1} > \cdots > v_1$ and*

$$\frac{v_{i+1}}{v_i} \geq (g+2)!, \quad \textit{for all } i \in [g].$$

Proof. Assume for a contradiction that such $g+1$ numbers $v_1, \ldots, v_{g+1}$ exist in V and let

$$v_i = c_{i,1} u_1 + \cdots + c_{i,g} u_g$$

where $c_{i,j} \in \{-1, 0, 1\}$ for each $i \in [g+1]$. Given that these are $g+1$ many g-dimensional vectors $c_i = (c_{i,1}, \ldots, c_{i,g})$, let $i^* \leq g+1$ be the smallest integer such that c_{i^*} can be written as a linear combination of $c_1, \ldots, c_{i^*-1}$: $c_{i^*} = \alpha_1 c_1 + \cdots + \alpha_{i^*-1} c_{i^*-1}$, which implies that

$$\begin{aligned} v_{i^*} &= \alpha_1 v_1 + \cdots + \alpha_{i^*-1} v_{i^*-1} \\ &\leq |\alpha_1| \cdot v_1 + \cdots + |\alpha_{i^*-1}| \cdot v_{i^*-1}. \end{aligned} \tag{6}$$

We show below that the magnitude of coefficients $\alpha_1, \ldots, \alpha_{i^*-1}$ is relatively small, which leads to a contradiction because we assumed that v_{i^*} is much bigger than $v_{i^*-1}, \ldots, v_1$.

To see this, note that $(\alpha_1, \ldots, \alpha_{i^*-1})$ is the solution to a $(i^*-1) \times (i^*-1)$ linear system $Ax = b$ where A is a $\{-1, 0, 1\}$-valued $(i^*-1) \times (i^*-1)$ full-rank matrix and b is a $\{-1, 0, 1\}$-valued vector. (In more detail, one can take A to be a full-rank $(i^*-1) \times (i^*-1)$ submatrix of the matrix that consists of $c_1, \ldots, c_{i^*-1}$ as columns and take the vector b to be the corresponding entries of c_{i^*}.) It follows from Cramer's rule that each entry of A^{-1} has magnitude at most $(i^*-1)!$ and thus, each entry of $A^{-1}b$ has absolute value at most $(i^*-1) \cdot (i^*-1)! < i^*! \leq (g+1)!$ This contradicts with (6) and the assumption that $v_1 < \ldots < v_{i^*-1} \leq v_{i^*}/(g+2)!$. □

Claim IV.5 gives us the following procedure to partition $[L]$ into $A_1, \ldots, A_q$ for some $q \leq \ell$:

```
1) Set $i = 1$ and $\mathcal{L} = [L]$.
2) While $\mathcal{L}$ is nonempty do
3)     Let $v$ be the smallest $|\Delta(T_a)|$, $a \in \mathcal{L}$.
4)     Remove from $\mathcal{L}$ and add to $A_i$ every $a \in \mathcal{L}$ with
       $|\Delta(T_a)| \leq (2\ell+2)! \cdot v$, and increment $i$.
```

It follows from Claim IV.5 that when $\mathcal{L}$ becomes empty at the end, the number of A_i's we created can be no more than ℓ. Furthermore, every a and a' that belong to the same A_i have the ratio of $|\Delta(T_a)|$ and $|\Delta(T_{a'})|$ bounded by $(2\ell+2)! = \ell^{O(\ell)}$. This finishes the proof of the lemma. □

B. Main Algorithm

We start with an algorithm, based on dynamic programming, for estimating the k-deck of a distribution $\mathbf{X}$ over $\{0,1\}^n$.

Theorem 4. *Let $k \in [n]$. There is an algorithm with the following performance guarantee: for any distribution $\mathbf{X}$ over strings in $\{0,1\}^n$, if the algorithm is given*

$$M = O\left(\frac{k}{\xi^2 (1-\delta)^{2k}}\right)$$

many samples from $\mathrm{Del}_\delta(\mathbf{X})$ *then with probability at least* 0.99 *the algorithm outputs a nonnegative* 2^k*-dimensional vector* Q *with* $\|Q - \mathsf{D}_k(\mathbf{X})\|_\infty \le \xi$. *Its running time is* $2^k M \cdot \mathrm{poly}(n)$.

Proof. Let $x^1, \ldots, x^p$ be the support of $\mathbf{X}$. Then for each string $v \in \{0,1\}^k$, we have

$$\begin{aligned}&\mathop{\mathbf{E}}_{\boldsymbol{z}\sim\mathrm{Del}_\delta(\mathbf{X})}\big[\#(v,\boldsymbol{z})\big]\\&= (1-\delta)^k \cdot \sum_{i=1}^{p} \mathbf{X}(x^i)\cdot \#(v,x^i)\\&= (1-\delta)^k \cdot \mathop{\mathbf{E}}_{\boldsymbol{x}\sim\mathbf{X}}\big[\#(v,\boldsymbol{x})\big]\\&= (1-\delta)^k \cdot \#(v,\mathbf{X}) = (1-\delta)^k \cdot \binom{n}{k}\cdot \big(\mathsf{D}_k(\mathbf{X})\big)_v.\end{aligned}$$

The first equation is because for a given size-k subset $S \subseteq [0:n-1]$ of indices at which v matches x^i, all of the positions in S "survive" into a string $\boldsymbol{z} \sim \mathrm{Del}_\delta(x^i)$ with probability exactly $(1-\delta)^k$.

As a result, it suffices to estimate $\mathbf{E}[\#(v,\boldsymbol{z})]$ to additive accuracy $\pm\xi(1-\delta)^k\binom{n}{k}$ for every string $v \in \{0,1\}^k$. For any fixed string $v \in \{0,1\}^k$, by a standard Chernoff bound, using

$$M = O\left(\frac{k}{\xi^2(1-\delta)^{2k}}\right)$$

samples the empirical estimate of $\mathbf{E}[\#(v,\boldsymbol{z})]$ will have the desired additive $\xi(1-\delta)^k\binom{n}{k}$ accuracy except with failure probability $0.01/2^k$. The success probability of 0.99 follows from union bound.

The running time of the algorithm uses the following simple observation: given $z \in \{0,1\}^{n'}$ and $v \in \{0,1\}^k$, there is a $\mathrm{poly}(n',k)$-time procedure that computes $\#(v,z)$. The procedure works by straightforward dynamic programming: For each $j \in [0:k-1]$ and $i \in [0:n'-1]$, the algorithm maintains a count of the number $\#(v_0 \ldots v_j, z_0 \ldots z_i)$. This then implies that the running time of the overall algorithm is $M \cdot 2^k \cdot \mathrm{poly}(n)$. This finishes the proof of the lemma. □

We prove the following main technical lemma in Sections IV-C and IV-D. Intuitively, this lemma says that if the total variation distance between $\mathbf{X}$ and $\mathbf{Y}$ is not too small, then for a suitable (not too large) value of k^*, the distance between the k^*-decks of $\mathbf{X}$ and $\mathbf{Y}$ also cannot be too small.

Lemma IV.6. *Let* ℓ *be a positive integer with* $\ell \le \log n$. *Let* $\mathbf{X}$ *and* $\mathbf{Y}$ *be two distributions, each supported over at most* ℓ *strings from* $\{0,1\}^n$. *Then there is a positive integer*

$$k^* = \sqrt{n}\cdot(\log n)^{O(\ell)} \tag{7}$$

such that

$$d_{\mathrm{TV}}(\mathbf{X},\mathbf{Y}) \le \exp\left(\sqrt{n}\cdot(\log n)^{O(\ell)}\right)\cdot\|\mathsf{D}_{k^*}(\mathbf{X}) - \mathsf{D}_{k^*}(\mathbf{Y})\|_\infty.$$

We now present our algorithm A and use Lemma IV.6 to prove Theorem 3:

Proof of Theorem 3. The bound (5) we aim for holds trivially when $\ell \ge \log n$. To see this, we first notice that when $\ell \ge \log n$, the sample complexity bound (5) we aim for is at least

$$\frac{\mathrm{poly}(\ell)}{\varepsilon^2}\cdot\left(\frac{1}{1-\delta}\right)^n. \tag{8}$$

With $(1/(1-\delta))^n$ samples from $\mathrm{Del}_\delta(\mathbf{X})$, we expect to see a full string of length n where no bits are deleted and we know that such a string is drawn directly from $\mathbf{X}$. This means that, with (8) many samples, we receive $\mathrm{poly}(\ell)/\varepsilon^2$ draws from $\mathbf{X}$ with high probability. When the latter happens, the empirical estimation $\tilde{\mathbf{X}}$ of $\mathbf{X}$ satisfies $d_{\mathrm{TV}}(\mathbf{X},\tilde{\mathbf{X}}) \le \varepsilon$ with high probability. This allows us to focus on the case when $\ell \le \log n$ in the rest of the proof (so Lemma IV.6 applies).

Let ε be the total variation distance we aim for in Theorem 3. Let k^* be the parameter in (7). Let ξ be a parameter to be specified later. By Theorem 4, the algorithm A can first use

$$M^* = O\left(\frac{k^*}{\xi^2(1-\delta)^{2k^*}}\right) \tag{9}$$

samples to obtain an estimate Q of $\mathsf{D}_{k^*}(\mathbf{X})$ such that

$$\|Q - \mathsf{D}_{k^*}(\mathbf{X})\|_\infty \le \xi, \tag{10}$$

and it succeeds in obtaining such an estimate with probability at least 0.99.

With Q in hand the algorithm A computes $\|Q - \mathsf{D}_{k^*}(\mathbf{Y})\|_\infty$ for every distribution $\mathbf{Y}$ supported on at most ℓ strings such that the probability of each string in $\mathbf{Y}$ is an integer multiple of ξ/ℓ. Finally the algorithm outputs the distribution $\mathbf{X}^*$ that minimizes the distance (breaking ties arbitrarily).

We show that when Q satisfies (10), $\mathbf{X}^*$ must be close to $\mathbf{X}$. We start with a simple observation that one can round $\mathbf{X}$ to get a distribution $\mathbf{X}'$ in which the probability of each string is an integer multiple of ξ/ℓ and $d_{\mathrm{TV}}(\mathbf{X},\mathbf{X}') \le \xi$. This can be done by rounding the probability of every string except one to the nearest multiple of ξ/ℓ and setting the last probability as required so that the total probability is 1. We have

$$\begin{aligned}&\big\|Q - \mathsf{D}_{k^*}(\mathbf{X}')\big\|_\infty\\&\le \big\|Q - \mathsf{D}_{k^*}(\mathbf{X})\big\|_\infty + \big\|\mathsf{D}_{k^*}(\mathbf{X}) - \mathsf{D}_{k^*}(\mathbf{X}')\big\|_\infty\\&\le \big\|Q - \mathsf{D}_{k^*}(\mathbf{X})\big\|_\infty + d_{\mathrm{TV}}(\mathbf{X},\mathbf{X}') \le 2\xi.\end{aligned}$$

By definition of $\mathbf{X}^*$ and $\mathbf{X}'$, we have $\|Q - \mathsf{D}_{k^*}(\mathbf{X}^*)\|_\infty \le \|Q - \mathsf{D}_{k^*}(\mathbf{X}')\|_\infty \le 2\xi$. As a result,

$$\begin{aligned}&\left\|\mathsf{D}_{k^*}(\mathbf{X}) - \mathsf{D}_{k^*}(\mathbf{X}^*)\right\|_\infty \\ &\quad\le \left\|Q - \mathsf{D}_{k^*}(\mathbf{X}^*)\right\|_\infty + \left\|Q - \mathsf{D}_{k^*}(\mathbf{X})\right\|_\infty \le 3\xi.\end{aligned}$$

It follows from Lemma IV.6 that

$$d_{\mathrm{TV}}(\mathbf{X}, \mathbf{X}^*) \le 3\xi \cdot \exp\left(\sqrt{n} \cdot (\log n)^{O(\ell)}\right).$$

Finally we choose ξ so that the RHS becomes ε. The number of samples needed in (9) becomes

$$\left(\frac{1}{\varepsilon}\right)^2 \cdot \left(\frac{2}{1-\delta}\right)^{\sqrt{n}\cdot(\log n)^{O(\ell)}}.$$

This finishes the proof of Theorem 3. □

We use the following two lemmas to prove Lemma IV.6. They are proved in Section IV-C and IV-D.

Lemma IV.7. *Let d, q, L and λ be positive integers satisfying*

$$d, q \le \log n \quad \textit{and} \quad L, \lambda \le (\log n)^{O(\log n)}.$$

Let $\Delta : \binom{[0:n-1]}{d} \to \mathbb{R}$ be a function that is not identically zero and has an (L, q, λ)-group cover. Let $m = d(n-1)L^2$. Then there exists a d-variate polynomial ϕ with degree at most $O(\sqrt{m} \cdot \log^{4q+1} m)$ and $\|\phi\|_1 = \exp(O(\sqrt{m} \cdot \log^{4q+3} m))$ such that

$$\begin{aligned}&\left|\sum_{0 \le t_1 < \cdots < t_d < n} \phi(t_1, \ldots, t_d) \cdot \Delta\big(\{t_1, \ldots, t_d\}\big)\right| \\ &\ge \frac{\|\Delta\|_\infty}{\exp(O(\sqrt{m} \cdot \log^{4q-1} m))}.\end{aligned}$$

We note that the following lemma holds for any two distributions $\mathbf{X}, \mathbf{Y}$ over $\{0,1\}^n$ regardless of their support size.

Lemma IV.8. *Let $d, k \in [n]$ with $k \ge d$. Let $\mathbf{X}, \mathbf{Y}$ be distributions each supported over strings from $\{0,1\}^n$. Then for any string $c \in \{0,1\}^d$ and d-variate polynomial ϕ of degree at most $k - d$,*

$$\begin{aligned}&\left|\sum_{0 \le t_1 < \cdots < t_d < n} \phi(t_1, \ldots, t_d) \cdot \Delta_{\mathbf{X},\mathbf{Y},c}\big(\{t_1, \ldots, t_d\}\big)\right| \\ &\le \|\phi\|_1 \cdot n^{O(k)} \cdot \|\mathsf{D}_k(\mathbf{X}) - \mathsf{D}_k(\mathbf{Y})\|_\infty.\end{aligned}$$

Proof of Lemma IV.6. Let $\mathbf{X}$ and $\mathbf{Y}$ be two distributions each supported over at most ℓ strings from $\{0,1\}^n$. It then follows from Lemma IV.1 and Lemma IV.4 that there exists a string $c \in \{0,1\}^d$ with $d = \lfloor \log(2\ell) \rfloor$ such that $\Delta := \Delta_{\mathbf{X},\mathbf{Y},c}$ satisfies $\|\Delta\|_\infty \ge d_{\mathrm{TV}}(\mathbf{X}, \mathbf{Y})/\ell^{O(\ell)}$ and has an (L, q, λ)-group cover for some $L \le 2^{2d\ell}$, $q \le \ell$, and $\lambda = \ell^{O(\ell)}$. As we assumed that $\ell \le \log n$, both d and q are at most $\log n$ and $L, \lambda \le \ell^{O(\ell)} \le (\log n)^{O(\log n)}$ (so Lemma IV.7 applies).

Let $m = d(n-1)L^2$ and ϕ be the polynomial given in Lemma IV.7. Let $k^* = \deg(\phi) + d$ (we set $k = k^*$ in Lemma IV.8; the choice of k^* ensures that $\deg(\phi) \le k^* - d$ as required in Lemma IV.8) with

$$k^* = O(\sqrt{m} \cdot \log^{4q+1} m) = \sqrt{n} \cdot (\log n)^{O(\ell)}.$$

Combining Lemma IV.7 and Lemma IV.8, we have

$$\begin{aligned}&\frac{\|\Delta\|_\infty}{\exp(\sqrt{n} \cdot (\log n)^{O(\ell)})} \\ &\le \exp\left(\sqrt{n} \cdot (\log n)^{O(\ell)}\right) \cdot n^{\sqrt{n}\cdot(\log n)^{O(\ell)}} \cdot \\ &\quad \|\mathsf{D}_{k^*}(\mathbf{X}) - \mathsf{D}_{k^*}(\mathbf{Y})\|_\infty.\end{aligned}$$

The lemma follows from the fact that $\|\Delta\|_\infty \ge d_{\mathrm{TV}}(\mathbf{X}, \mathbf{Y})/\ell^{O(\ell)}$. □

C. *Proof of Lemma IV.7*

Let Δ be a function over d-subsets of $[0 : n-1]$ that is not identically zero and has an (L, q, λ)-group cover $\{(T_a, \mathcal{S}_a) : a \in [L]\}$ with a q-way partition $A_1, \ldots, A_q$ of $[L]$. We start with a high-level description of the d-variate polynomial ϕ.

To evaluate ϕ on a tuple $(t_1, \ldots, t_d)$, we first project $(t_1, \ldots, t_d)$ onto a line along the direction of $(w_1, \ldots, w_d)$ for some relatively small positive integers $w_1, \ldots, w_d$ to be specified later, and then apply a univariate polynomial $f(\cdot)$ on the image of the projection. In other words ϕ takes the form

$$\phi(t_1, \ldots, t_d) = f\big(w_1 t_1 + \cdots + w_d t_d\big) \tag{11}$$

for some positive integers $w_1, \ldots, w_d \in [L^2]$. We give details below.

1) The projection : Let $m = d(n-1)L^2$ and let w be the following function from size-d subsets of $[0 : n-1]$ to $[0 : m]$:

$$w(T) = w_1 t_1 + \cdots + w_d t_d,$$

where $T = \{t_1, \ldots, t_d\}$ with $t_1 < \cdots < t_d$. So w is the projection function that maps a size-d subset T of $[0 : n-1]$ (or equivalently, a sorted d-tuple of distinct values from $[0 : n-1]$) to a location on the real line. Claim IV.9 implies that there exist $w_1, \ldots, w_d \in [L^2]$ such that all anchor sets in the L-cover are mapped to distinct locations.

Claim IV.9. *If $w_1, \dots, w_d$ are drawn independently and uniformly at random from $[L^2]$ then $w(T_a) \neq w(T_{a'})$ for all $a \neq a' \in [L]$ with probability at least $1/2$.*

Proof. Let $S = \{s_1, \dots, s_d\}$ and $T = \{t_1, \dots, t_d\}$ denote two size-d subsets of $[0 : n-1]$ that satisfy $s_1 < \cdots < s_d$, $t_1 < \cdots < t_d$ and $S \neq T$. Then the probability that $w(S) = w(T)$ equals

$$\mathbf{Pr}\left[w_1 s_1 + \cdots + w_d s_d = w_1 t_1 + \cdots + w_d t_d\right]. \quad (12)$$

As $(s_1, \dots, s_d) \neq (t_1, \dots, t_d)$, one of the d quantities $s_i - t_i$ is nonzero; say without loss of generality $s_1 \neq t_1$. Fixing any outcomes of random draws of $w_2, \dots, w_d$, there is a unique outcome of w_1 which would result in the equation in (12), and the probability that w_1 takes this particular outcome is either $1/L^2$ or zero (if it is not in $[L^2]$). As a result, the probability in (12) is at most $1/L^2$, and the claim follows from a union bound over $\binom{L}{2}$ events. □

We fix such a tuple $w_1, \dots, w_d \in [L^2]$ that satisfies Claim IV.9 for the rest of the proof.

2) The univariate polynomial : Now we move to the more difficult part of choosing the univariate polynomial f in (11).

A useful tool. A key tool for our construction of f is a univariate polynomial h with several useful properties described below. Figure 1 gives a schematic representation of the key upper bounds on $|h(b)|$ provided by item (2) in Lemma IV.10.

Lemma IV.10. *There is a univariate polynomial h with the following properties:*

1) *h has degree $O(\sqrt{m}\log m)$.*
2) *$h(0) = 1$ and for each $b \in [m]$,*
$$|h(b)| \leq \frac{1}{2^{\sqrt{b}}} \quad \text{and} \quad |h(-b)| \leq e^{6\sqrt{b}\log m}.$$
3) *h satisfies $\|h\|_1 \leq \exp(O(\sqrt{m}\log m))$.*

Our construction of the polynomial h is based on the Chebyshev polynomial and builds on an earlier construction due to Borwein et al. [BEK99]. We prove Lemma IV.10 in Appendix A, and we explain the role that h plays in the construction of our desired univariate polynomial f under the heading "**The high-level idea**" below, after first providing some useful preliminary explanation.

Given that our polynomial ϕ takes the form of (11), the crucial quantity whose magnitude we are trying to lower bound, namely

$$\sum_{0 \leq t_1 < \cdots < t_d < n} \phi(t_1, \dots, t_d) \cdot \Delta\big(\{t_1, \dots, t_d\}\big)$$

(recall the LHS of Lemma IV.7), can be written as

$$\sum_{b \in [0:m]} f(b) \cdot \Gamma(b), \quad (13)$$

where $\Gamma : [0 : m] \to \mathbb{R}$ is a function that is defined using Δ as follows:

$$\Gamma(b) = \sum_{T:\, w(T) = b} \Delta(T), \quad (14)$$

where the sum is over all d-subsets T of $[0 : n-1]$.

To better understand Γ, we use the (L, q, λ)-group cover of Δ to introduce two new sequences $\tau_0, \dots, \tau_r$ and $m_0, \dots, m_r$, for some value $r \in [0 : q-1]$ that is defined below. We start with some notation. For each $i \in [q]$, we let $\mathcal{G}_i = \cup_{a \in A_i} \mathcal{S}_a$ and refer to $\mathcal{G}_i$ as *group* i. We refer to the T_a with the smallest $w(T_a)$ among all $a \in A_i$ as the *anchor* of group i and denote it by V_i. (By Claim IV.9, each group has a unique anchor and we have $w(T) > w(V_i)$ for all $T \in \mathcal{G}_i$ other than V_i.) We let $v_i = |\Delta(V_i)|$ and $\kappa_i = w(V_i)$, so κ_i is the location that the anchor V_i of $\mathcal{G}_i$ is projected to. By the definition of an (L, q, λ)-group cover and Claim IV.9, we have that each $v_i > 0$, the κ_i's are distinct,

$$\max_{i \in [q]} v_i \geq \frac{\|\Delta\|_\infty}{\lambda}.$$

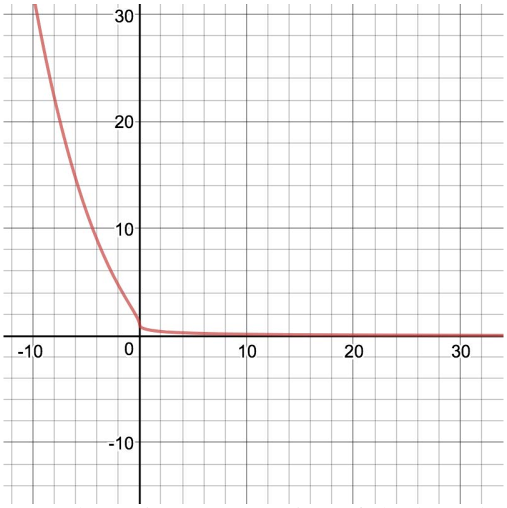

Fig. 1: A schematic representation of the bounds on $|h(b)|$ given by item (2) of Lemma IV.10. The three key properties are that (i) $h(0) = 1$; (ii) for $b \in [m]$, the upper bound on $|h(b)|$ is very small and decreases rapidly as we move away from 0; and (iii) for $b \in [-m : -1]$, the upper bound on $|h(b)|$ is not too large and does not increase too rapidly as we move away from 0.

Now we are ready to define r and the two sequences. See Figure 2 for an illustration of these sequences. First we let $\tau_0 = \max_{i \in [q]} v_i$ and also let $m_0 \in [0 : m]$ denote the smallest κ_i (among all groups $i \in [q]$) with $v_i = \tau_0$. We are done and the value of r is 0 if no κ_i is smaller than m_0 (i.e. the anchor of every other group is projected to a larger location value than m_0); otherwise, we let $\tau_1 < \tau_0$ be the largest value of v_i over those $i \in [q]$ that have $\kappa_i < m_0$ and also let $m_1 < m_0$ be the smallest κ_i such that $v_i = \tau_1$. We are done and the value of r is 1 if no κ_i is smaller than m_1 (i.e. every other group anchor is projected to a larger location value than m_1); otherwise we repeat the process. Continuing in this way, at the end we obtain two sequences:

$$0 < \tau_r < \cdots < \tau_0, \quad \text{with } \tau_0 = \max_{i \in [q]} v_i \geq \frac{\|\Delta\|_\infty}{\lambda} \text{ and}$$
$$0 \leq m_r < \cdots < m_0 \leq m,$$

for some value $r \in [0 : q-1]$. We say that $(\tau_0, \ldots, \tau_r)$ is the *τ-step-sequence* and that $(m_0, \ldots, m_r)$ is the *m-step-sequence* for Γ.

The high-level idea. Before entering into further details we give intuition for the polynomial f. Looking ahead to (18), the polynomial f is essentially a translation of the polynomial h depicted in Figure 1, i.e. $f(x)$ is essentially $h(x - m_\alpha)$ for some $\alpha \in [0 : r]$.[12] Recalling the key properties of h, we see that

- $f(m_\alpha) = 1$.
- $|f(b)|$ is "very small" for $b > m_\alpha$; and
- $|f(b)|$ is "not too large" for $b < m_\alpha$.

The crux of our analysis below is to establish that there is a suitable value m_α in the m-step-sequence which is such that the magnitude of the single summand $f(m_\alpha) \cdot \Gamma(m_\alpha)$ in (13) is greater than the contribution of all other summands in (13).

To gain some intuition for why this is the case, let us pretend that instead of the Γ being defined as in (13), the definition of Γ instead only took a sum over the q anchors $V_1, \ldots, V_q$ of the q groups $\mathcal{G}_1, \ldots, \mathcal{G}_q$ (i.e. Γ is supported on κ_i, $i \in [q]$, with $|\Gamma(\kappa_i)| = v_i$). Of course this is not actually the case since each group $\mathcal{G}_i$ in general contains many more sets than just its anchor T_i, but it turns out that the effect of other sets in $\mathrm{supp}(\Delta)$ will only cost us some extra $n^d\lambda$ factors in the analysis (corresponding to the $n^d\lambda$ factors in properties (ii) and (iii) of $\Gamma_0, \ldots, \Gamma_r$ as described below, where $n^d \geq \binom{n}{d}$ just serves as a bound for the number of size-d subsets) which turn out to be manageable.

In this hypothetical scenario the only nonzero values of $\Gamma(b)$ that would enter the picture would be the v_i values at locations κ_i, $i \in [q]$, which are the heights of the bars in Figure 2. The desired m_α could then be identified as follows:

- We proceed in an inductive fashion. For each $p \in [0 : r]$, we show that there is a choice of $\alpha \in [0 : p]$ such that, by setting $f(x) = h(x - m_\alpha)$, the value of $|f(m_\alpha) \cdot \Gamma(m_\alpha)|$ outweighs $|f(b) \cdot \Gamma(b)|$ for every other $b \in [m_p : m]$. The choice of α at the end of the induction when p reaches r gives us the desired location m_α for the translation of h to define f.
- The base case when $p = 0$ is trivial by setting $\alpha = 0$ and $f(x) = h(x - m_0)$. Here we have that $|f(m_0) \cdot \Gamma(m_0)|$ outweighs $|f(b) \cdot \Gamma(b)|$ for all $b > m_0$ because $|\Gamma(m_0)| = \tau_0 \geq |\Gamma(b)|$ by the definition of our step-sequences and the fact that $f(m_0) = 1$ is "much larger" than $|f(b)|$ for $b > m_0$.
- Next we move to $p = 1$, and now we need to take $\Gamma(b)$, $b \in [m_1 : m_0 - 1]$, into consideration. To this end we compare τ_0/τ_1 with $\exp(\sqrt{m_0 - m_1})$ and consider the following two cases.
  - If τ_0/τ_1 is larger then we can keep $\alpha = 0$ and $f(x) = h(x - m_0)$ because $|f(m_0) \cdot \Gamma(m_0)|$ outweighs $|f(m_1) \cdot \Gamma(m_1)|$ (since $\Gamma(m_1) = \tau_1$ and $f(m_1)$ is roughly[13] $\exp(\sqrt{m_0 - m_1})$) as well as $|f(b) \cdot \Gamma(b)|$ for all $b \in [m_1 + 1 : m_0 - 1]$ (since $|f(m_1)| > |f(b)|$ and by the definition of our step-sequences, $|\Gamma(m_1)| \geq |\Gamma(b)|$). By the inductive hypothesis we also know that $|f(m_0) \cdot \Gamma(m_0)|$ outweighs $|f(b) \cdot \Gamma(b)|$ for all $b > m_0$.
  - Otherwise (if τ_1 is larger than $\tau_0/\exp(\sqrt{m_0 - m_1})$) we show that setting $\alpha = 1$ and $f(x) = h(x - m_1)$ works. On the one hand, $|f(m_1) \cdot \Gamma(m_1)|$ outweighs $|f(b) \cdot \Gamma(b)|$ for $b \in [m_1 + 1 : m_0 - 1]$ since $|\Gamma(b)| \leq |\Gamma(m_1)|$ by the definition of our step-sequences and the fact that $f(m_1) = 1$ is "much larger" than $|f(b)|$ (similar to the base case). On the other hand, $|f(m_1) \cdot \Gamma(m_1)| = \tau_1$ outweighs $|f(m_0) \cdot \Gamma(m_0)| = |f(m_0)| \cdot \tau_0$

[12] The exponent of h in the exact definition of our f given in Equation (18) is needed for technical reasons that are not important for this intuitive explanation.

[13] This is not entirely precise because in (2) of Lemma IV.10 there is indeed an extra factor of $\log m$ in the exponent on the left side of 0; overcoming this factor of $\log m$ is the reason why we end up with the exponent as in (18).

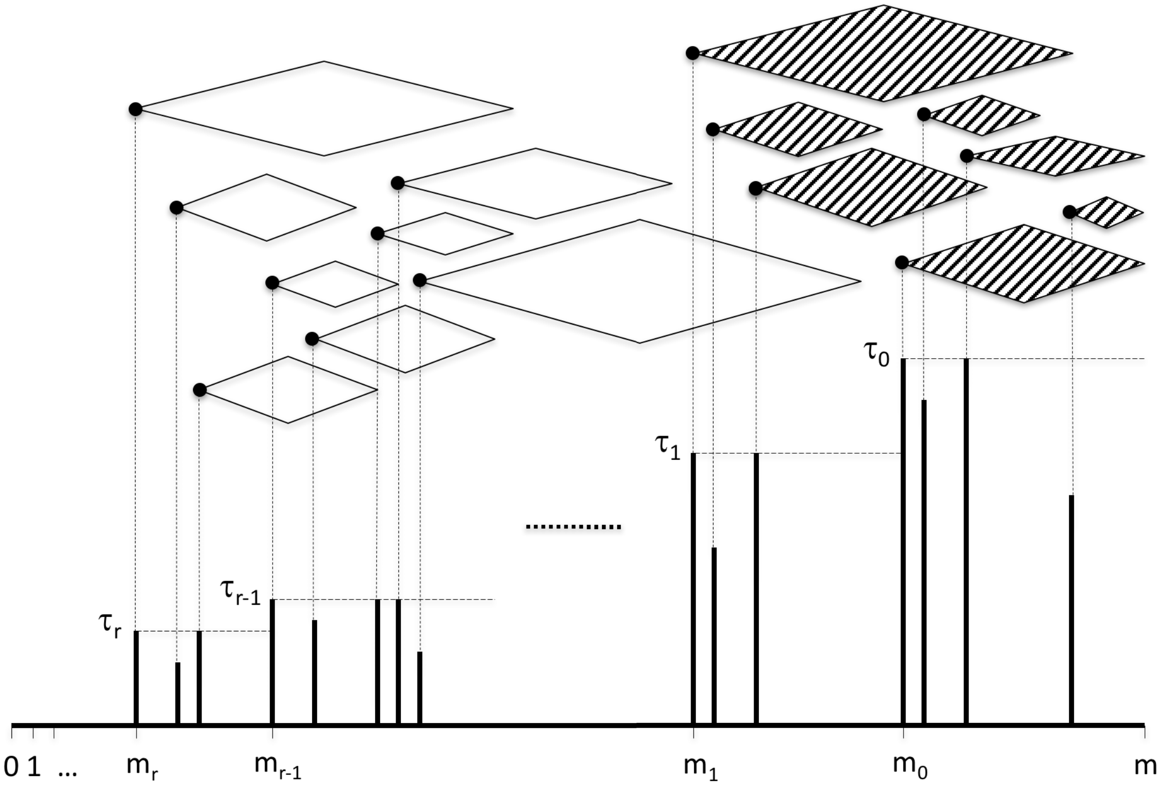

Fig. 2: An illustration of a τ-step-sequence and its associated m-step-sequence. The values $\tau_r < \tau_{r-1} < \cdots < \tau_1 < \tau_0$ (which may be arbitrary real positive values) are the heights of the bars at locations $0 \leq m_r < m_{r-1} < \cdots < m_1 < m_0 \leq m$ (these locations are integers). The location of each vertical bar corresponds to some $\kappa_i, i \in [q]$, and its height is v_i; the corresponding group $\mathcal{G}_i$ is illustrated as a diamond, with V_i being its left corner. Note that all the bars between locations m_i and m_{i-1} have height at most τ_i. See Example 5 for an explanation of why certain diamonds are shaded in the figure.

(since $|f(m_0)|$ is, roughly speaking, $\exp(-\sqrt{m_0 - m_1})$) as well as $|f(b) \cdot \Gamma(b)|$ for $b > m_0$ (since $|f(m_0)| > |f(b)|$ and $|\Gamma(m_0)| \geq |\Gamma(b)|$ so the contribution from b is smaller than that from m_0).

- Continuing in this fashion, we show that, if α is the choice for some $p \in [0 : r-1]$, then for $p+1$ we can either keep the same choice of α or move α to $p+1$, depending on the result of a similar comparison between τ_α / τ_{p+1} and $\exp(\sqrt{m_\alpha - m_{p+1}})$. This finishes the induction.

The above reasoning is formalized in the statement and proof of the (crucial) Lemma IV.11, which additionally has to deal with the complication that it must address the real scenario rather than the hypothetical simplification considered in the informal description above.

Now we turn to the details. For each $b \in [0 : m]$, we let

$$\mathcal{G}_{\geq b} = \bigcup_{i \in [q] \colon \kappa_i \geq b} \mathcal{G}_i.$$

For each $p \in [0 : r]$ let Γ_p denote the following function on $[0 : m]$:

$$\Gamma_p(b) = \sum_{T \in \mathcal{G}_{\geq m_p} \colon w(T) = b} \Delta(T).$$

In words, the value of Γ_p evaluated at a location value b is obtained as follows: for each group $\mathcal{G}_i$ for which the location κ_i that the anchor set V_i is projected to is at least m_p, we sum the value of $\Delta(T)$ over all $T \in \mathcal{G}_i$ which are mapped by the projection function w to the location b.

Example 5. In Figure 2, only $\mathcal{G}_i$'s that correspond to shaded diamonds are considered in Γ_1.

We have the following properties from our choices of τ_i's and m_i's:

(i) $\Gamma = \Gamma_r$. This is because every location κ_i is at least m_r.

(ii) Γ_0 is such that $\Gamma_0(b) = 0$ for all $b < m_0$, $|\Gamma_0(m_0)| = \tau_0$, and

$$|\Gamma_0(b)| \le n^d \lambda \tau_0$$

for all $b > m_0$. (The last bound holds just because there are at most n^d many size-d subsets and the maximum value of $|\Delta(T)|$ on any T contributing to the sum $\Gamma_0(b)$ is at most $\lambda\tau_0$.)

(iii) Generalizing the previous property, for each $p \in [r]$, $\Gamma_p(b) = 0$ for $b < m_p$, $|\Gamma_p(m_p)| = \tau_p$ and

$$\left|\Gamma_p(b) - \Gamma_{p-1}(b)\right| \le n^d \lambda \tau_p$$

for all $b > m_p$ (since the maximum magnitude of $\Delta(T)$ on any subset T contributing to the sum $\Gamma_p(b)$ but not to $\Gamma_{p-1}(b)$ is at most $\lambda\tau_p$).

We require the following crucial lemma, Lemma IV.11, concerning Γ_p. Intuitively, the lemma states that for each p there is a suitable index $\alpha \le p$ (so $m_\alpha \ge m_p$) such that (a) the magnitude of $\Gamma_p(m_\alpha)$ is not too small compared to τ_0 (this is given by (15)); (b) for locations $b > m_\alpha$ the magnitude of $\Gamma_p(b)$ is not too large compared to the magnitude of $\Gamma_p(m_\alpha)$ (this is given by (16)); and (c) for locations b between m_p and m_α the magnitude of $\Gamma_p(b)$ is small compared to the magnitude of $\Gamma_p(m_\alpha)$ (this is given by (17)). In all three places the meaning of "small" or "large" is specified by a second parameter β which can grow slowly with p. We defer the proof, which proceeds by induction on p and makes the high-level idea (described earlier) precise, to the full version of the paper [BCF+19].

Lemma IV.11. *Assume that $d \le \log n$ and $\lambda \le (\log n)^{O(\log n)}$. Then for each $p \in [0:r]$ there are two parameters $\alpha_p \in [0:p]$ and $\beta_p \in [0:4p+3]$ (letting α denote α_p and β denote β_p below for convenience) such that*

$$\left|\Gamma_p(m_\alpha)\right| \cdot 2^p \cdot \exp\left(\sqrt{m} \cdot \log^\beta m\right) \ge \tau_0 \qquad (15)$$

and every index $b \in [m_p : m]$ satisfies

1) If $b \ge m_\alpha$, then

$$\left|\Gamma_p(b)\right| \le \left|\Gamma_p(m_\alpha)\right| \cdot 2^p \cdot \exp\left(\sqrt{b - m_\alpha} \cdot \log^\beta m\right); \qquad (16)$$

2) If $m_p \le b < m_\alpha$, then

$$\left|\Gamma_p(b)\right| \cdot \exp\left(\sqrt{m_\alpha - b} \cdot \log^{\beta+3} m\right) \le 2^p \cdot \left|\Gamma_p(m_\alpha)\right|. \qquad (17)$$

Finally we combine Lemma IV.10 and Lemma IV.11 to prove Lemma IV.7.

Proof of Lemma IV.7. Recall that $d, q \le \log n$ and $\lambda, L \le (\log n)^{O(\log n)}$.

Let $\alpha \in [0:r]$ and $\beta \in [0:4r+3]$ be the final parameters that satisfy Lemma IV.11 for $\Gamma_r = \Gamma$. We define the polynomial f using h from Lemma IV.10 as follows

$$f(x) = \Big(h(x - m_\alpha)\Big)^{\left\lceil 3\log^{\beta+1} m \right\rceil}. \qquad (18)$$

It follows from Lemma IV.10 that f has degree

$$\deg(f) = O\big(\sqrt{m} \cdot \log^{\beta+2} m\big) = O\big(\sqrt{m} \cdot \log^{4q+1} m\big).$$

using $r < q$ and $\beta \le 4r+3$. Moreover, we have

$$\begin{aligned} \|f\|_1 &\le \exp\big(O(\sqrt{m} \cdot \log^{4q+1} m)\big) \cdot (m+1)^{\deg(f)} \\ &= \exp\big(O(\sqrt{m} \cdot \log^{4q+2} m)\big). \end{aligned}$$

It follows from the definition of ϕ in (11) and $w_1, \ldots, w_d \in [L^2]$ that the same degree upper bound holds for ϕ and

$$\begin{aligned} \|\phi\|_1 &\le \exp\big(O(\sqrt{m} \cdot \log^{4q+2} m)\big) \cdot (dL^2)^{\deg(f)} \\ &\le \exp\big(O(\sqrt{m} \cdot \log^{4q+3} m)\big). \end{aligned}$$

To analyze $\sum_b f(b) \cdot \Gamma(b)$, we show that $|f(b) \cdot \Gamma(b)| \le |\Gamma(m_\alpha)|/(2m)$ for all $b \ne m_\alpha$ and thus,

$$\begin{aligned} \left|\sum_{b \in [0:m]} f(b) \cdot \Gamma(b)\right| &\ge \frac{|\Gamma(m_\alpha)|}{2} \overset{(15)}{\ge} \frac{\tau_0}{2^q \cdot \exp(\sqrt{m} \cdot \log^\beta m)} \\ &\ge \frac{\|\Delta\|_\infty}{\exp(O(\sqrt{m} \cdot \log^{4q-1} m))}, \end{aligned}$$

using $\tau_0 \ge \|\Delta\|_\infty/\lambda$. For each $b > m_\alpha$, by Lemma IV.11 and Lemma IV.10 (and $q \le \log n$),

$$\begin{aligned} \left|f(b) \cdot \Gamma(b)\right| &\le \left|\Gamma(m_\alpha)\right| \cdot 2^q \cdot \frac{\exp\left(\sqrt{b - m_\alpha} \cdot \log^\beta m\right)}{\exp\left(\sqrt{b - m_\alpha} \cdot 3\log^{\beta+1} m\right)} \\ &\le \frac{|\Gamma(m_\alpha)|}{2m}. \end{aligned}$$

For each $b < m_\alpha$ we have from Lemma IV.11 and Lemma IV.10 that

$$\begin{aligned} \left|f(b) \cdot \Gamma(b)\right| &\le 2^q \cdot |\Gamma(m_\alpha)| \cdot \frac{\exp\left(\sqrt{m_\alpha - b} \cdot \log^{\beta+2} m\right)}{\exp(\sqrt{m_\alpha - b} \cdot \log^{\beta+3} m)} \\ &\le \frac{|\Gamma(m_\alpha)|}{2m}. \end{aligned}$$

This finishes the proof of Lemma IV.7. □

D. *Proof of Lemma IV.8*

Let $\mathbf{X}$ and $\mathbf{Y}$ be two distributions each supported over strings from $\{0,1\}^n$.

Given $0 \le j_1 < \cdots < j_d \le k-1$, we use $g_{j_1,\dots,j_d}$ to denote the following d-variate polynomial,

$$g_{j_1,\dots,j_d}(t_1,\dots,t_d) := \binom{t_1}{j_1}\cdot\binom{t_2-t_1-1}{j_2-j_1-1}\cdots \binom{t_d-t_{d-1}-1}{j_d-j_{d-1}-1}\binom{n-t_d-1}{k-j_d-1}. \tag{19}$$

To see the relevance of this polynomial to the k-deck, we note that given any $0 \le t_1 < \cdots < t_d < n$ the quantity $g_{j_1,\dots,j_d}(t_1,\dots,t_d)$ is the number of ways to pick k indices from $[0:n-1]$ such that each t_i is the (j_i+1)th smallest index picked.

We first show that the following sum

$$\sum g_{j_1,\dots,j_d}(t_1,\dots,t_d)\cdot \mathrm{restrict}\big(\mathbf{X},\{t_1,\dots,t_d\},c\big) \tag{20}$$

(where the sum is over all $0 \le t_1 < \cdots < t_d < n$) can be written as a low-weight linear combination of entries of $\mathsf{D}_k(\mathbf{X})$.

Lemma IV.12. *For any $0 \le j_1 < \cdots < j_d \le k-1$ and any $c \in \{0,1\}^d$, the sum (20) can be written as a linear combination of entries of $\mathsf{D}_k(\mathbf{X})$ in which each coefficient is either 0 or $\binom{n}{k}$.*

Proof. Recalling the combinatorial interpretation of $g_{j_1,\dots,j_d}(t_1,\dots,t_d)$ given after (19), we see that if we divide the sum in (20) by $\binom{n}{k}$, the result is precisely the probability that $(\boldsymbol{z}_{j_1},\dots,\boldsymbol{z}_{j_d}) = c$ when we draw $\boldsymbol{x} \sim \mathbf{X}$, draw a size-$k$ subset $\mathbf{T}$ of $[0:n-1]$ uniformly at random, and then set $\boldsymbol{z} = \boldsymbol{x}_{\mathbf{T}}$. The latter probability can also be expressed using entries of $\mathsf{D}_k(\mathbf{X})$ as

$$\sum_{\substack{z\in\{0,1\}^k \\ (z_{j_1},\dots,z_{j_d})=c}} \big(\mathsf{D}_k(\mathbf{X})\big)_z,$$

as $(\mathsf{D}_k(\mathbf{X}))_z$ is the probability of $\boldsymbol{x}_{\mathbf{T}} = z$ with $\boldsymbol{x}$ and $\mathbf{T}$ drawn as above. This finishes the proof. □

Next we show that, for every monomial $t_1^{r_1}\cdots t_d^{r_d}$ of degree $r_1+\cdots+r_d \le k-d$, there exists a low-weight linear combination of polynomials $g_{j_1,\dots,j_d}$ that agrees with $t_1^{r_1}\cdots t_d^{r_d}$ over $t_1,\dots,t_d$ that satisfy $0 \le t_1 < \cdots < t_d < n$.

Lemma IV.13. *For any nonnegative integers $r_1,\dots,r_d$ with $r_1+\cdots+r_d \le k-d$, we have that*

$$t_1^{r_1}\cdots t_d^{r_d} = \sum_{0\le j_1<\cdots<j_d<k} w_{j_1,\dots,j_d}\cdot g_{j_1,\dots,j_d}(t_1,\dots,t_d),$$

for all $0 \le t_1 < \cdots < t_d < n$, where the coefficients $w_{j_1,\dots,j_d}$ satisfy $\sum |w_{j_1,\dots,j_d}| \le k^{O(k)}$.

Before proving Lemma IV.13, we use Lemma IV.12 and Lemma IV.13 to prove Lemma IV.8.

Proof of Lemma IV.8. Combining Lemma IV.12 and Lemma IV.13, we have that

$$\begin{aligned}\sum_{0\le t_1<\cdots<t_d<n} & t_1^{r_1}\cdots t_d^{r_d}\cdot \mathrm{restrict}\big(\mathbf{X},\{t_1,\dots,t_d\},c\big) \\ = \sum_{0\le j_1<\cdots<j_d<k} & w_{j_1,\dots,j_d} \sum_{0\le t_1<\cdots<t_d<n} g_{j_1,\dots,j_d}(t_1,\dots,t_d)\cdot \\ & \mathrm{restrict}\big(\mathbf{X},\{t_1,\dots,t_d\},c\big)\end{aligned}$$

can be written as a linear combination of entries of $\mathsf{D}_k(\mathbf{X})$ in which each coefficient has magnitude at most $k^{O(k)}\cdot\binom{n}{k} = n^{O(k)}$. As a result, we have

$$\begin{aligned}&\left|\sum_{0\le t_1<\cdots<t_d<n} t_1^{r_1}\cdots t_d^{r_d}\cdot \Delta_{\mathbf{X},\mathbf{Y},c}\big(\{t_1,\dots,t_d\}\big)\right| \\ &\le n^{O(k)}\cdot \|\mathsf{D}_k(\mathbf{X})-\mathsf{D}_k(\mathbf{Y})\|_\infty.\end{aligned}$$

This finishes the proof of the lemma. □

Finally we prove Lemma IV.13. We follow a three-step approach. We say that a *quasimonomial* is a polynomial of the form

$$t_1^{\alpha_1}\cdot(t_2-t_1-1)^{\alpha_2}\cdot(t_3-t_2-1)^{\alpha_3}\cdots(t_d-t_{d-1}-1)^{\alpha_d}$$

for some nonnegative integers $\alpha_1,\dots,\alpha_d$; the degree of this quasimonomial is $\alpha_1+\cdots+\alpha_d$. And we say that a *PBC* (*Product of Binomial Coefficients*) is a polynomial of the form

$$\binom{t_1}{\beta_1}\binom{t_2-t_1-1}{\beta_2}\cdots\binom{t_d-t_{d-1}-1}{\beta_d}$$

for some nonnegative integers $\beta_1,\dots,\beta_d$; the degree of this PBC is $\beta_1+\cdots+\beta_d$. We observe that, compared to PBCs, the polynomials $g_{j_1,\dots,j_d}$ have an extra binomial coefficient that involves t_d at the end. The three steps of our approach are as follows:

- **First step:** Express each d-variable monomial $t_1^{r_1}\cdots t_d^{r_d}$ with $r_1+\cdots+r_d \le k-d$ as a low-weight linear combination of quasimonomials of degree at most $k-d$.
- **Second step:** Express each quasimonomial of degree at most $k-d$ as a low-weight linear combination of PBCs of degree at most $k-d$.
- **Third step:** Finally, express each PBC of degree at most $k-d$ as a low-weight linear combination of polynomials $g_{j_1,\dots,j_d}$.

For each step, we bound the sum of magnitudes of coefficients in the linear combination. The rest of the proof of Lemma IV.13, including the details of each step, is deferred to the full version of our paper [BCF$^+$19].

V. Lower bounds for distributions supported on at most 2ℓ strings

Our main result in this section is Theorem 6, given below, which establishes a lower bound on the sample complexity of population recovery under the deletion channel which is exponential in the population size for a wide range of population sizes:

Theorem 6. *Fix any constant deletion probability $\delta \in (0,1)$. Suppose that A is an algorithm which, when run on i.i.d. samples drawn from a distribution $\mathrm{Del}_\delta(\mathbf{X})$ with $|\mathrm{supp}(\mathbf{X})| \le 2\ell$, outputs a hypothesis $\tilde{\mathbf{X}}$ which satisfies $d_{\mathrm{TV}}(\mathbf{X}, \tilde{\mathbf{X}}) \le 0.49$ with probability at least 0.51. Then A must use*

$$\frac{\Omega\left(n/\ell^2\right)^{\frac{\ell+1}{2}}}{\ell^{\frac{3}{2}}}$$

many samples.

If the population size upper bound 2ℓ is a constant this gives a lower bound of $\Omega(n^{(\ell+1)/2})$ samples, and for any $\ell < n^{0.499}$ this gives a lower bound of $n^{\Omega(\ell)}$.

For the rest of this section fix $\delta \in (0,1)$ and let ρ denote $1-\delta$. The high-level idea of the proof is as follows: We show that there exist two distributions $\mathbf{X}, \mathbf{Y}$ over $\{0,1\}^n$ which have disjoint supports, each of size at most 2ℓ, but satisfy

$$d_{\mathrm{TV}}(\mathrm{Del}_\delta(\mathbf{X}), \mathrm{Del}_\delta(\mathbf{Y})) = O\left(\frac{\ell^2}{n}\right)^{\frac{\ell+1}{2}} \cdot \ell^{\frac{3}{2}} \cdot (1-\delta) \tag{21}$$

which clearly implies Theorem 6.

For simplicity throughout this section we assume that n is odd, and we write m to denote $(n-1)/2$. The following notation will be useful: For $0 \le i \le 2\ell$ we write e_{m+i} to denote the string $0^{m+i}10^{m-i}$. The two distributions $\mathbf{X}$ and $\mathbf{Y}$ that we consider will be supported on disjoint subsets of $\{e_{m+i}\}_{i\in[0:2\ell]}$ (and hence each distribution has support size at most $2\ell+1$, but in our proofs neither will have full support so their support size will be at most 2ℓ).

Notation and setup. For notational convenience, let $\mathrm{B}(r)$ denote the binomial distribution $\mathrm{Bin}(r,\rho)$.

Let S be a set of indices, π_S be a distribution over S, and $\{\mathbf{V}_i\}_{i\in S}$ be a set of random variables indexed by S. We write $\mathrm{Mix}(\pi_S; \{\mathbf{V}_i\}_{i\in S})$ to denote the mixture over $\{\mathbf{V}_i\}_{i\in S}$ with each $\mathbf{V}_i$ weighted by $\pi_S(i)$.

For conciseness we write $\mathbf{Z}_n$ to denote a random variable which is distributed according to the binomial distribution $\mathrm{B}(n)$. We recall the following convenient expression for the falling moments of the binomial distribution: for any $t = 0, 1, \ldots$, we have

$$\mathbf{E}[\mathbf{Z}_n(\mathbf{Z}_n-1)\cdots(\mathbf{Z}_n-t)] = P_t(n) \tag{22}$$

where $P_t(n) = n(n-1)\cdots(n-t)\rho^{t+1}$.

For completeness we include the derivation below:

$$\begin{aligned}
&\mathbf{E}[\mathbf{Z}_n(\mathbf{Z}_n-1)\cdots(\mathbf{Z}_n-t)]\\
&= \sum_{i=0}^{n} i(i-1)\cdots(i-t)\cdot\binom{n}{i}\rho^i(1-\rho)^{n-i}\\
&= \sum_{i=t+1}^{n} \frac{n!}{(n-i)!(i-t-1)!}\cdot\rho^i\cdot(1-\rho)^{n-i}\\
&= n\cdots(n-t)\rho^{t+1}\sum_{j=0}^{n-t-1}\binom{n-t-1}{j}\rho^j(1-\rho)^{n-t-1-j}\\
&= P_t(n).
\end{aligned}$$

The key lemmas. The first main lemma makes precise the moment-matching property of π_S and π_T that we require:

Lemma V.1 (Matching moments of mixtures of disjointly supported binomial distributions)**.** *Let $\ell \le O(\sqrt{n})$.*[14] *There are two disjoint subsets $S, T \subset [0:2\ell]$ and two distributions π_S, π_T supported on $\{e_{m+i}\}_{i\in S}$ and $\{e_{m+j}\}_{j\in T}$ respectively with the following property (which we call the "matching moment property"):*

Let $\tilde{\mathbf{D}}_S$ be a random variable whose distribution is the mixture of $\{\mathbf{Z}_{m+i}\}_{i\in S}$ in which distribution $\mathbf{Z}_{m+i}$ has mixing weight $\pi_S(e_{m+i})$, and likewise $\tilde{\mathbf{D}}_T$ be a random variable whose distribution is the mixture of $\{\mathbf{Z}_{m+j}\}_{j\in T}$ in which distribution $\mathbf{Z}_{m+j}$ has mixing weight $\pi_T(e_{m+j})$. Then the first ℓ moments of $\tilde{\mathbf{D}}_S$ and $\tilde{\mathbf{D}}_T$ match each other, i.e. for all $t \in [\ell]$, we have

$$\mathbf{E}[(\tilde{\mathbf{D}}_S)^t] = \mathbf{E}[(\tilde{\mathbf{D}}_T)^t]. \tag{23}$$

The second main lemma (statement given in Lemma V.3 below) gives the desired upper bound on total variation distance. To prove Theorem 6 it suffices to prove Lemmas V.1 and V.3.

A. Proof of Lemma V.1

Proof. We defer the proof to the full version of the paper [BCF$^+$19]. □

[14] Note that if $\ell = \omega(\sqrt{n})$ then Theorem 6 holds trivially, so this assumption is without loss of generality.

We will use the following corollary of Lemma V.1:

Corollary V.2. *Let S, T, π_S, π_T be as in Lemma V.1. Then for any polynomial p of degree at most ℓ, we have*

$$\sum_{i\in\mathbb{N}} \pi_S(e_{m+i})p(m+i) = \sum_{j\in\mathbb{N}} \pi_S(e_{m+j})p(m+j). \quad (24)$$

Proof. Equation (23) can be rewritten as

$$\sum_{i\in\mathbb{N}} \pi_S(e_{m+i})\,\mathbf{E}[(\mathbf{Z}_{m+i})^t] = \sum_{j\in\mathbb{N}} \pi_S(e_{m+j})\,\mathbf{E}[(\mathbf{Z}_{m+j})^t],$$

which holds for all $t \leq \ell$.

This is equivalent to having equal falling moments, i.e. for all $t \in [\ell]$, $\sum_{i\in\mathbb{N}} \pi_S(e_{m+i})\,\mathbf{E}[P_{t-1}(\mathbf{Z}_{m+i})]$ equals $\sum_{j\in\mathbb{N}} \pi_S(e_{m+j})\,\mathbf{E}[P_{t-1}(\mathbf{Z}_{m+j})]$. Indeed, for a random variable $\mathbf{Z}$, $\mathbf{E}[P_{t-1}(\mathbf{Z})]$ can be written as a linear combination of $1, \mathbf{E}[\mathbf{Z}], \mathbf{E}[\mathbf{Z}^2], \ldots, \mathbf{E}[\mathbf{Z}^t]$ and since $1, P_0(\mathbf{Z}), P_1(\mathbf{Z}), \ldots, P_{\ell-1}(\mathbf{Z})$ form a set of ℓ polynomials in $\mathbf{Z}$ with degrees $0, 1, 2, \ldots, \ell$, then they form a basis for polynomials in $\mathbf{Z}$ with degree at most ℓ.

By (22), this is in turn equivalent to having, for all $t \in [\ell]$,

$$\sum_{i\in\mathbb{N}} \pi_S(e_{m+i})\cdot P_{t-1}(m+i) = \sum_{j\in\mathbb{N}} \pi_S(e_{m+j})\cdot P_{t-1}(m+j),$$

which is in turn equivalent to (24) by the reasoning in the above paragraph. □

B. Total Variation Distance Upper Bound

We state Lemma V.3 below. Informally, it says that if π_S, π_T have the matching moment property, then the variation distance between two corresponding mixtures of two-dimensional vector-valued random variables is small. (In the following, the notation $(\mathrm{B}(a), \mathrm{B}(b))$ stands for a vector-valued random variable in which the two coordinates are independently drawn from $\mathrm{B}(a)$ and $\mathrm{B}(b)$ respectively.)

Lemma V.3. *Let $\mathbf{X}, \mathbf{Y}$ be two distributions with disjoint supports $\{e_{m+i}\}_{i\in S}$ and $\{e_{m+j}\}_{j\in T}$ respectively, where $S \cup T \subset [0 : 2\ell]$, with the matching moment property from Lemma V.1 above. Then*

$$d_{\mathrm{TV}}(\mathrm{Del}_\delta(\mathbf{X}), \mathrm{Del}_\delta(\mathbf{Y})) \leq O\left(\frac{\ell^2}{n}\right)^{\frac{\ell+1}{2}} \cdot \ell^{\frac{3}{2}} \cdot (1-\delta). \quad (25)$$

Setup and useful results. Our proof of Lemma V.3 is based on "moment-matching" results for Poisson binomial distributions which were proved by Roos [Roo00] and subsequently used by Daskalakis and Papadimitriou [DP15]. Our approach is similar to the approach used in [DP15]. To state these results, recall that a *Poisson binomial distribution* (PBD) is a sum $\mathbf{U} = \mathbf{A}_1 + \cdots + \mathbf{A}_n$ of independent Bernoulli random variables (so each $\mathbf{A}_i$ is a random variable taking value 1 with some probability $p_i \in [0,1]$ and taking value 0 with probability $1 - p_i$).

In [DP15], it is shown that if two PBDs satisfy some mild technical condition and have matching first ℓ moments, then they have total variation distance at most $2^{-\Omega(\ell)}$. We show that two mixtures of pairs of suitable binomially distributed variables that have matching first ℓ moments will have total variation distance at most $n^{-\Omega(\ell)}$.

We recall Theorem 1 of [DP15], which gives a "Krawtchouk expansion" for any Poisson binomial distribution. This provides an expression for the exact probability that the Poisson binomial distribution puts on any outcome in its support. (We state the theorem for PBDs which are a sum of n' many random variables, as when we apply it later it will be for such PBDs where $n' = m + \ell = (n-1)/2 + \ell$.)

Theorem 7 (Theorem 1 of [Roo00], see also Theorem 7 of [DP15]). *Let $\mathbf{U} = \mathbf{A}_1 + \cdots + \mathbf{A}_{n'}$ be a Poisson binomial distribution in which each $\mathbf{A}_i$ takes value 1 with probability $p_i \in [0,1]$. Then for all $r \in [n']$ and all $p \in [0,1]$, we have*

$$\mathbf{Pr}[\mathbf{U} = r] = \sum_{t=0}^{n'} \alpha_t(p_1, \ldots, p_{n'}; p) \cdot \Delta^t B_{n'-t,p}(r), \quad (26)$$

where

- *$\alpha_0(p_1, \ldots, p_{n'}; p) = 1$ and for $t \in [0 : n']$,*
$$\alpha_t(p_1, \ldots, p_{n'}; p) := \sum_{1\leq u(1)<\cdots<u(t)\leq n'} \prod_{r=1}^{t} (p_{u(r)} - p),$$
- *and for all $t \in [0 : n']$,*
$$\Delta^t B_{n'-t,p}(r) := \frac{(n'-t)!}{n'!} \cdot \frac{d^t}{dp^t} B_{n',p}(r),$$
where in the last expression $B_{n',p}(r)$ denotes the value $\binom{n'}{r} p^r (1-p)^{n'-r}$, the probability that the binomial distribution $\mathrm{Bin}(n', p)$ puts on the outcome r, viewed as a function of p.

We highlight the fact that $\Delta^t B_{n',p}(r)$ has no dependence on the parameters $p_1, \ldots, p_{n'}$; this will be important for us later.

The following result, deduced from [Roo00], is very useful in analyzing (26). It bounds each of the $n' + 1$ summands in (26) which add up to $\mathbf{Pr}[\mathbf{U} = r]$.

Theorem 8. *Let $(p_1, \ldots, p_{n'}) \in [0,1]^{n'}$, $p \in [0,1]$, and $\alpha_t(\cdot,\cdot)$ be as in the statement of Theorem 7. Define*

$$\theta(p_1, \ldots, p_{n'}; p) := \frac{2\sum_{i=1}^{n'}(p_i - p)^2 + (\sum_{i=1}^{n'}(p_i - p))^2}{2n'p^2(1-p)^2}. \tag{27}$$

For $t \in [n']$,

$$\begin{aligned}&|\alpha_t(p_1, \ldots, p_{n'}; p)| \cdot \|\Delta^t B_{n'-t,p}(\cdot)\|_1 \\ &\le \sqrt{e} \cdot \theta(p_1, \ldots, p_{n'}; p)^{\frac{t}{2}} t^{\frac{1}{4}}\end{aligned}$$

where $\|\Delta^t B_{n'-t,p}(\cdot)\|_1$ denotes the 1-norm of $\Delta^t B_{n'-t,p}(\cdot)$ viewed as an $(n'+1)$-dimensional vector, i.e. $\|\Delta^t B_{n'-t,p}(\cdot)\|_1 := \sum_{r=0}^{n'} |\Delta^t B_{n'-t,p}(r)|$.

Proof. Inequality (30) in [Roo00] says $|\alpha_t(p_1, \ldots, p_{n'}; p)|$ is at most

$$p^{\frac{t}{2}}(1-p)^{\frac{t}{2}}\theta(p_1, \ldots, p_{n'}; p)^{\frac{t}{2}} \left(\frac{n'}{n'-t}\right)^{\frac{n'-t}{2}}$$

for $t \in [n']$.

Inequality (38) in [Roo00] gives

$$\|\Delta^t B_{n'-t,p}(\cdot)\|_1 \le \sqrt{e}\cdot t^{\frac{1}{4}} \left(\frac{n'-t}{n'}\right)^{\frac{n'-t}{2}} \left(\frac{t}{n'p(1-p)}\right)^{\frac{t}{2}}$$

for $t \in [n']$.

By multiplying the above two inequalities together we get the desired result because $t \le n'$. □

For conciseness we now let $\mathbf{D}_S$ denote $\mathrm{Mix}(\pi_S; ((\mathrm{Bin}(m+i,\rho), \mathrm{Bin}(m-i,\rho)))_{i\in S})$ where in each component two-dimensional distribution the two distributions $\mathrm{Bin}(m+i,\rho)$ and $\mathrm{Bin}(m-i,\rho)$ are independent, and similarly we let $\mathbf{D}_T$ denote $\mathrm{Mix}(\pi_T; ((\mathrm{Bin}(m+j,\rho), \mathrm{Bin}(m-j,\rho)))_{j\in T})$. In the rest of the proof we will argue that

$$d_{\mathrm{TV}}(\mathbf{D}_S, \mathbf{D}_T) \le O\left(\frac{\ell^2}{n}\right)^{\frac{\ell+1}{2}} \cdot \ell^{\frac{3}{2}} \tag{28}$$

This establishes the claimed upper bound on $d_{\mathrm{TV}}(\mathrm{Del}_\delta(\mathbf{X}), \mathrm{Del}_\delta(\mathbf{Y}))$ given in (25). To see this, observe that for any outcome in $\mathrm{supp}(\mathbf{X})$ or $\mathrm{supp}(\mathbf{Y})$, with probability δ the one 1-coordinate is deleted under Del_δ (in which case the distributions resulting from $\mathrm{Del}_\delta(\mathbf{X})$ and $\mathrm{Del}_\delta(\mathbf{Y})$ are identical), and that with the remaining $1-\delta$ probability (when the one 1-coordinate is not deleted) there is an exact correspondence between $\mathrm{Del}_\delta(\mathbf{X})$ and $\mathbf{D}_S$ and between $\mathrm{Del}_\delta(\mathbf{Y})$ and $\mathbf{D}_T$.

For an index $c \le n'$, let $v^{(c)}$ denote the n'-dimensional real vector whose first c values are ρ and whose remaining values are 0.

For $t, t' \in [0:n']$ we define

$$C_{t,t'}(p) = \sum_{i\in\mathbb{N}} \pi_S(e_{m+i}) \cdot \alpha_t(v^{(m+i)}; p) \cdot \alpha_{t'}(v^{(m-i)}; p),$$

$$D_{t,t'}(p) = \sum_{j\in\mathbb{N}} \pi_T(e_{m+j}) \cdot \alpha_t(v^{(m+j)}; p) \cdot \alpha_{t'}(v^{(m-j)}; p).$$

The following lemma is crucial for us. Recall that $n' = m + \ell$.

Lemma V.4. *Let π_S, π_T be as in the statement of Lemma V.3. Then for any $p \in [0,1]$, the values $C_{t,t'}(p)$ and $D_{t,t'}(p)$ are identical for $t, t' \ge 0$ and $t + t' \le \ell$.*

Proof. Let p be any value in $[0,1]$. If $t + t' = 0$, then $t = t' = 0$. Recalling that $\alpha_0(\cdot,\cdot) \equiv 1$ we have that $C_{0,0}(p) = \sum_{i\in\mathbb{N}} \pi_S(e_{m+i}) = 1 = \sum_{j\in\mathbb{N}} \pi_T(e_{m+j}) = D_{0,0}(p)$ as desired.

Let $\kappa(k) = \binom{m+k}{c}\binom{n'-m-k}{t-c}\binom{m-k}{c'}\binom{n'-m+k}{t'-c'}$.

For $t + t' \ge 1$, we observe that $\alpha_t(v^{(m+i)}; p) \cdot \alpha_{t'}(v^{(m-i)}; p)$ is composed of summands of the form $(\rho - p)^{c+c'}(-p)^{t+t'-c-c'}$ for $c \in [0,t], c' \in [0,t']$.

In particular, we have $C_{t,t'}(p) = \sum_{i\in\mathbb{N}} \pi_S(e_{m+i}) \cdot \sum_{c=0}^{t}\sum_{c'=0}^{t'} \kappa(i) \cdot (\rho - p)^{c+c'} \cdot (-p)^{t+t'-c-c'}$, in which each $\pi_S(e_{m+i})$ is multiplied by a polynomial in m of degree at most $t + t' \le \ell$.

Similarly, we have $D_{t,t'}(p) = \sum_{j\in\mathbb{N}} \pi_T(e_{m+j}) \cdot \sum_{c=0}^{t}\sum_{c'=0}^{t'} \kappa(j) \cdot (\rho - p)^{c+c'} \cdot (-p)^{t+t'-c-c'}$ and by Corollary V.2, we see that $C_{t,t'}(p) = D_{t,t'}(p)$. □

We can now prove Lemma V.3 using a similar argument to the one used in the proof of Theorem 3 in [DP15]. Our proof will upper bound $\mathbf{Pr}[\mathbf{D}_S = (r,s)] - \mathbf{Pr}[\mathbf{D}_T = (r,s)]$ by using Theorem 7, Lemma V.4, and Theorem 8. The proof is deferred to the full version of the paper [BCF+19].

Acknowledgment

Frank Ban is supported by NSF award CCF-1819935. Xi Chen is supported by NSF awards CCF-1703925 and IIS-1838154. Adam Freilich is supported by NSF award CCF-1563155. Rocco A. Servedio is supported by NSF awards CCF-1563155, CCF-1814873 and IIS-1838154. Sandip Sinha is supported by NSF awards CCF-1563155, CCF-1420349, CCF-1617955, CCF-1740833, CCF-1421161, CCF-1714818 and Simons Foundation (#491119).

References

[BCF+19] Frank Ban, Xi Chen, Adam Freilich, Rocco A. Servedio, and Sandip Sinha. Beyond trace reconstruction: Population recovery from the deletion channel. *CoRR*, abs/1904.05532, 2019. IV-C2, IV-D, V-A, V-B

[BEK99] Peter Borwein, Tamás Erdélyi, and Géza Kós. Littlewood-type problems on $[0, 1]$. *Proceedings of the London Mathematical Society*, 3(79):22–46, 1999. II-B, II-B, II-B, II-B, IV-C2, A

[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 - 16th International Workshop, APPROX 2013, and 17th International Workshop, RANDOM 2013, Berkeley, CA, USA, August 21-23, 2013. Proceedings*, pages 347–362, 2013. I

[BKKM04] T. Batu, S. Kannan, S. Khanna, and A. McGregor. Reconstructing strings from random traces. In *Proceedings of the Fifteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2004*, pages 910–918, 2004. I

[BP92] Robert Brawer and Magnus Pirovino. The linear algebra of the pascal matrix. *Linear Algebra and its Applications*, 174:13–23, 1992.

[CK97] Christian Choffrut and Juhani Karhumäki. Combinatorics of words. In *Handbook of Formal Languages, Volume I*, pages 329–438. Springer, 1997. II-A

[DDS15] C. Daskalakis, I. Diakonikolas, and R. A. Servedio. Learning Poisson Binomial Distributions. *Algorithmica*, 72(1):316–357, 2015. II-C

[DOS17a] Anindya De, Ryan O'Donnell, and Rocco A. Servedio. Optimal mean-based algorithms for trace reconstruction. In *Proceedings of the 49th ACM Symposium on Theory of Computing (STOC)*, pages 1047–1056, 2017. I, I-A, II, II, II-A

[DOS17b] Anindya De, Ryan O'Donnell, and Rocco A. Servedio. Sharp bounds for population recovery. *CoRR*, abs/1703.01474, 2017. I

[DP15] Constantinos Daskalakis and Christos Papadimitriou. Sparse covers for sums of indicators. *Probability Theory & Related Fields*, 162:679–705, 2015. (document), II-C, V-B, 7, V-B

[DRWY12] Z. Dvir, A. Rao, A. Wigderson, and A. Yehudayoff. Restriction access. In *Innovations in Theoretical Computer Science*, pages 19–33, 2012. I

[DS03] Miroslav Dudík and Leonard Schulman. Reconstruction from subsequences. *Journal of Combinatorial Theory, Series A*, 103(2):337–348, 2003. II-A

[DST16] A. De, M. Saks, and S. Tang. Noisy population recovery in polynomial time. Technical Report TR-16-026, Electronic Colloquium on Computational Complexity, 2016. To appear in FOCS 2016. I

[HHP18] Lisa Hartung, Nina Holden, and Yuval Peres. Trace reconstruction with varying deletion probabilities. In *Proceedings of the Fifteenth Workshop on Analytic Algorithmics and Combinatorics, ANALCO 2018, New Orleans, LA, USA, January 8-9, 2018.*, pages 54–61, 2018. I, 1, I-A

[HL18] N. Holden and R. Lyons. Lower bounds for trace reconstruction. *CoRR*, abs/1808.02336, 2018. I

[HMPW08] T. Holenstein, M. Mitzenmacher, R. Panigrahy, and U. Wieder. Trace reconstruction with constant deletion probability and related results. In *Proceedings of the Nineteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2008*, pages 389–398, 2008. I, I-A, II, II

[HPP18] Nina Holden, Robin Pemantle, and Yuval Peres. Subpolynomial trace reconstruction for random strings and arbitrary deletion probability. *CoRR*, abs/1801.04783, 2018. I, I-A

[Kal73] V. V. Kalashnik. Reconstruction of a word from its fragments. *Computational Mathematics and Computer Science (Vychislitel'naya matematika i vychislitel'naya tekhnika), Kharkov*, 4:56–57, 1973. I, II-A

[KM05] Sampath Kannan and Andrew McGregor. More on reconstructing strings from random traces: Insertions and deletions. In *IEEE International Symposium on Information Theory*, pages 297–301, 2005. I

[KR97] Ilia Krasikov and Yehuda Roditty. On a reconstruction problem for sequences,. *Journal of Combinatorial Theory, Series A*, 77(2):344–348, 1997. (document), II-A, II-B, II-B, II-B

[Lev01a] Vladimir Levenshtein. Efficient reconstruction of sequences. *IEEE Transactions on Information Theory*, 47(1):2–22, 2001. I

[Lev01b] Vladimir Levenshtein. Efficient reconstruction of sequences from their subsequences or supersequences. *Journal of Combinatorial Theory Series A*, 93(2):310–332, 2001. I

[LZ15] S. Lovett and J. 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, STOC 2015, Portland, OR, USA, June 14-17, 2015*, pages 137–142, 2015. I

[MMS$^+$91] Bennet Manvel, Aaron Meyerowitz, Allen Schwenk, Ken Smith, and Paul Stockmeyer. Reconstruction of sequences. *Discrete Mathematics*, 94(3):209–219, 1991. II-A

[MPV14] Andrew McGregor, Eric Price, and Sofya Vorotnikova. Trace reconstruction revisited. In *Proceedings of the 22nd Annual European Symposium on Algorithms*, pages 689–700, 2014. I, II-A, II-C

[MS13] Ankur Moitra and Michael E. Saks. A polynomial time algorithm for lossy population recovery. In *54th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2013, 26-29 October, 2013, Berkeley, CA, USA*, pages 110–116, 2013. I

[NP17] Fedor Nazarov and Yuval Peres. Trace reconstruction with $\exp(o(n^{1/3}))$ samples. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2017*, pages 1042–1046, 2017. I, I-A, II, II, II-A

[OAC$^+$18] Lee Organick, Siena Dumas Ang, Yuan-Jyue Chen, Randolph Lopez, Sergey Yekhanin, Konstantin Makarychev, Miklos Z Racz, Govinda Kamath, Parikshit Gopalan, Bichlien Nguyen, et al. Random access in large-scale dna data storage. *Nature biotechnology*, 36(3):242, 2018. I

[PSW17] Yury Polyanskiy, Ananda Theertha Suresh, and Yihong Wu. Sample complexity of population recovery. In *Proceedings of the 30th Conference on Learning Theory, COLT 2017, Amsterdam, The Netherlands, 7-10 July 2017*, pages 1589–1618, 2017. I

[PZ17] Yuval Peres and Alex Zhai. Average-case reconstruction for the deletion channel: subpolynomially many traces suffice, 2017. Available at https://arxiv.org/abs/1708.00854. I, I-A

[Roo00] B. Roos. Binomial approximation to the Poisson binomial distribution: The Krawtchouk expansion. *Theory Probab. Appl.*, 45:328–344, 2000. (document), II-C, V-B, 7, V-B, V-B

[Sco97] Alexander Scott. Reconstructing sequences. *Discrete Mathematics*, 175(1):231–238, 1997. II-A, II-B

[VS08] Krishnamurthy Viswanathan and Ram Swaminathan. Improved string reconstruction over insertion-deletion channels. In *Proceedings of the 19th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 399–408, 2008. I

[WY16] A. Wigderson and A. Yehudayoff. Population recovery and partial identification. *Machine Learning*, 102(1):29–56, 2016. Preliminary version in FOCS 2012. I

[YGM17] S.M. Hossein Tabatabaei Yazdi, Ryan Gabrys, and Olgica Milenkovic. Portable and error-free DNA-based data storage. *Scientific Reports*, 7(1):5011, 2017. I

APPENDIX

A. Chebyshev polynomials

Let $T_r(x)$ denote the rth Chebyshev polynomial of the first kind. Then T_r has degree r and satisfies the following property:

Property 9. *$T_r(1) = 1$ and $|T_r(x)| \le 1$ for all $|x| \le 1$. If $x > 1$ then $T_r(x) > 1$.*

We will need an upper bound for $T_r(x)$ over $x \in [1,2]$. For this purpose we recall the following explicit form of $T_r(x)$ for $|x| \ge 1$:

$$T_r(x) = \frac{(x-\sqrt{x^2-1})^r + (x+\sqrt{x^2-1})^r}{2}. \tag{29}$$

Property 10. *For $a \in [0,1]$, we have $T_r(1+a) \le e^{3r\sqrt{a}}$.*

Proof. Using (29) we have

$$T_r(1+a) \le \left(1+a+\sqrt{2a+a^2}\right)^r \le (1+3\sqrt{a})^r \le e^{3r\sqrt{a}},$$

where we used $a^2 \le a \le \sqrt{a}$ when $a \in [0,1]$ and $1+x \le e^x$. □

The next property follows from the recurrence relation

$$T_{r+1}(x) = 2x \cdot T_r(x) - T_{r-1}(x)$$

with initial conditions $T_0(x) = 0$ and $T_1(x) = 1$.

Property 11. *For all $r \ge 0$, we have $\|T_r\|_1 \le 3^r$.*

Following [BEK99], we write g_r to denote the following polynomial of degree r:

$$g_r(x) = \frac{1}{r+0.5} \cdot \left(\frac{T_0(x)}{2} + T_1(x) + \cdots + T_r(x)\right).$$

We need the following properties of the polynomial g_r. Items 1, 2 and 3 of Property 12 follow directly from Properties 9, 10 and 11, respectively. For item 4 we have

$$\begin{aligned} g_r(\cos y) &= \frac{1}{r+0.5}(0.5 + \cos y + \cos 2y + \cdots + \cos ry) \\ &= \frac{1}{r+0.5} \cdot \frac{\sin(r+0.5)y}{\sqrt{2(1-\cos y)}}, \end{aligned}$$

for all $0 < y \le \pi$. This implies that for all $x \in [-1,1)$, we have

$$|g_r(x)| \le \frac{1}{r+0.5} \cdot \frac{1}{\sqrt{2(1-x)}} \le \frac{1}{r\sqrt{2(1-x)}}.$$

Property 12. *The polynomial g_r satisfies the following properties.*

1) *$g_r(1) = 1$ and $|g_r(x)| \le 1$ for all $|x| \le 1$;*
2) *$1 \le g_r(1+a) \le e^{3r\sqrt{a}}$ for all $a \in [0,1]$;*
3) *$\|g_r\|_1 \le 3^r$; and*
4) *$|g_r(x)| \le \frac{1}{r\sqrt{2(1-x)}}$ for all $x \in [-1,1)$.*

B. Proof of Lemma IV.10

Recall the statement of Lemma IV.10:

Lemma IV.10, restated. *There is a univariate polynomial h with the following properties:*

1) *h has degree $O(\sqrt{m}\log m)$.*
2) *$h(0) = 1$ and for each $b \in [m]$,*

$$|h(b)| \le \frac{1}{2^{\sqrt{b}}} \quad \textit{and} \quad |h(-b)| \le e^{6\sqrt{b}\log m}.$$

3) *h satisfies $\|h\|_1 \le \exp(O(\sqrt{m}\log m))$.*

Proof. Recall the polynomial g_r in Section A. We use it to define a degree-r polynomial ψ_r:

$$\psi_r(x) = g_r\left(1 - \frac{x}{m}\right).$$

Properties of g_r directly imply the following properties of ψ_r:

1) $\psi_r(0) = 1$;
2) For each $b \in [m]$, we have

$$|\psi_r(b)| \le \min\left(1, \frac{1}{r}\sqrt{\frac{m}{2b}}\right) \text{ and } 1 \le \psi_r(-b) \le e^{3r\sqrt{b/m}};$$

3) Finally, ψ_r satisfies

$$\|\psi_r\|_1 \le 3^r \cdot \left(1 + \frac{1}{m}\right)^r.$$

Let $\tilde{m} = 4^\beta$ be the smallest power of 4 with $\tilde{m} \ge m$. We use ψ_r to define our h as follows:

$$h(x) = \prod_{i\in[\beta]} \left(\psi_{\sqrt{\tilde{m}/4^{i-2}}}(x)\right)^{\sqrt{4^i}}.$$

First we have $h(0) = 1$ and the degree of h is at most

$$\sum_{i\in[\beta]} \sqrt{\frac{\tilde{m}}{4^{i-2}}} \cdot \sqrt{4^i} = O(\sqrt{\tilde{m}}\log_4 \tilde{m}) = O(\sqrt{m}\log m).$$

Next, given $b \in [m]$, let $i \in [\beta]$ be an integer such that $4^{i-1} \le b \le 4^i$. Then (using $\tilde{m} \ge m$)

$$\left|\psi_{\sqrt{\tilde{m}/4^{i-2}}}(b)\right| \le \sqrt{\frac{4^{i-2}}{\tilde{m}}} \cdot \sqrt{\frac{m}{2b}} \le \sqrt{\frac{4^{i-2}}{m}} \cdot \sqrt{\frac{m}{2 \cdot 4^{i-1}}} < \frac{1}{2}.$$

Using $|\psi_r(b)| \le 1$ for all r, we have that

$$|h(b)| \le \frac{1}{2^{\sqrt{4^i}}} \le \frac{1}{2^{\sqrt{b}}}.$$

On the other hand, we have for each $b \in [m]$ (using $\tilde{m} \le 4m$ and that m is asymptotically large),

$$\begin{aligned} h(-b) &\le \exp\left(3\sqrt{b/m} \sum_{i \in [\beta]} \sqrt{\tilde{m}/4^{i-2}} \cdot \sqrt{4^i}\right) \\ &= \exp\left(24\sqrt{b}\,\log_4 \tilde{m}\right) \le \exp\left(24\sqrt{b}\log m\right). \end{aligned}$$

Finally, the sum of magnitudes of coefficients of h is at most

$$\prod_{i \in [\beta]} \left(3^{\sqrt{\tilde{m}/4^{i-2}}} \cdot 2\right)^{\sqrt{4^i}} = \exp\left(O(\sqrt{m}\log m)\right).$$

This finishes the proof of the lemma. □