Two Party Distribution Testing: Communication and Security

Alexandr Andoni
Columbia University
andoni@cs.columbia.edu

Tal Malkin
Columbia University
tal@cs.columbia.edu

Negev Shekel Nosatzki
Columbia University
ns3049@columbia.edu

November 8, 2018

Abstract

We study the problem of discrete distribution testing in the *two-party setting*. For example, in the standard closeness testing problem, Alice and Bob each have t samples from, respectively, distributions a and b over $[n]$, and they need to test whether $a = b$ or a, b are ϵ-far (in the ℓ_1 distance) for some fixed $\epsilon > 0$. This is in contrast to the well-studied one-party case, where the tester has unrestricted access to samples of both distributions, for which optimal bounds are known for a number of variations. Despite being a natural constraint in applications, the two-party setting has evaded attention so far.

We address two fundamental aspects of the two-party setting: 1) what is the communication complexity, and 2) can it be accomplished securely, without Alice and Bob learning extra information about each other's input. Besides closeness testing, we also study the independence testing problem, where Alice and Bob have t samples from distributions a and b respectively, which may be correlated; the question is whether a, b are independent of ϵ-far from being independent. Our contribution is three-fold:

- **Communication**: we show how to gain communication efficiency as we have more samples, beyond the information-theoretic bound on t. Furthermore, the gain is polynomially better than what one may obtain by adapting one-party algorithms.

  For the closeness testing, our protocol has communication $s = \tilde{\Theta}_\epsilon\left(n^2/t^2\right)$ as long as t is at least the information-theoretic minimum number of samples. For the independence testing over domain $[n] \times [m]$, where $n \geq m$, we obtain $s = \tilde{O}_\epsilon(n^2 m/t^2 + nm/t + \sqrt{m})$.
- **Lower bounds**: we prove *tightness* of our trade-off for the closeness testing, as well as that the independence testing requires tight $\Omega(\sqrt{m})$ communication for unbounded number of samples. These lower bounds are of independent interest as, to the best of our knowledge, these are the first 2-party communication lower bounds for testing problems, where the inputs represent a set of *i.i.d. samples*.
- **Security**: we define the concept of secure distribution testing and argue that it must leak at least some minimal information when the promise is not satisfied. We then provide secure versions of the above protocols with an overhead that is only polynomial in the security parameter.

1 Introduction

Distribution property testing is a sub-area of statistical hypothesis testing, which has enjoyed continuously growing interest in the theoretical computer science community, especially since the 2000 papers [GR00, BFR$^+$00]. One of the most basic problems is closeness testing, also known as the *homogeneity testing* or *two sample problem*; see [Gut89, Sha11, Unn12]. Here, given two distributions a, b and t samples from each of them, distinguish between the cases where $a = b$ versus a and b are ϵ-far, which usually means $\|a - b\|_1 > \epsilon$.[1] For this specific problem, the extensive research lead to algorithms with optimal sample complexity [BFR$^+$00, Val11, BFR$^+$13, CDVV14, DK16, DGPP16], including when the number of samples from the two distributions is unequal [AJOS14, BV15, DK16]. Further research directions of interest include obtaining instance-optimal algorithms, which depend on further properties of the distributions a, b [ADJ$^+$11, ADJ$^+$12, DK16], quantum algorithms [BHH11], as well as algorithms whose output is differentially-private [DHS15, CDK17, ASZ17, ADR17]. An even larger body of work studied numerous related problems such as independence testing, among many others. We refer the reader to the surveys [Gol17, Can15, Rub12, Rub06] for further references.

Focusing on testing two distributions, such as in the closeness problem, a very natural aspect has, surprisingly, evaded attention so far: that such a testing task would be often run by two players, each with access to their own distribution. Specifically, Alice has samples from the distribution a, Bob has samples from the distribution b, and they need to jointly solve a distribution testing problem on (a, b). This is a natural setting that models many of the envisioned usage scenarios of distribution testing, where different parties wish to jointly perform a statistical hypothesis testing task on their distributions. For example, [Sha11] describes the scenario where two distinct sensors need to test whether they sample from the same distribution ("noise") or not.

This 2-party setting raises the following standard theoretical challenges, neither of which has been previously studied in the context of distribution testing:

- What is the *communication complexity* of the testing problem? In particular, can we do better than the straightforward approach, where Alice sends her samples to Bob who then runs an offline algorithm? Can we prove matching communication lower bounds for such testing problems?

  This aspect parallels the quest for low memory or communication usage for hypothesis testing on a *single distribution*, initiated in the statistics community [Cov69, HC70] and [AC86, Han87, A$^+$98].

- Is it possible to design a distribution testing protocol that is *secure*, i.e., where Alice and Bob do not learn anything about each other's samples, besides the mere fact of whether their distributions are same or ϵ-far?

  While more modern, this question is highly relevant in today's push for doing statistics that is more privacy-respecting.

1.1 Our Contributions

In this paper, we initiate the study of testing problems in the two-party model, and design protocols which are both communication-efficient and secure. We do so for two basic problems on pairs of distributions (i.e., where the two-party setting is natural): 1) closeness testing, and 2) independence testing.

Our main finding is that, once the number of samples exceeds the information-theoretic minimum, we can obtain protocols with polynomially smaller communication than the naïve adaptation of existing algorithms. We complement our protocols with lower bounds on the communication complexity of such problems that are near-optimal for closeness testing, as well as for independence in an intriguing

[1] This is equivalent to saying that the total variation distance is more than $\epsilon/2$.

parameter regime. Our upper and lower bounds on communication are novel even without any security considerations.

To argue security, we also put forth a definition for secure distribution testing in the multi-party model. Our definition differs from the standard secure computation setting due to two unique features of the considered setting. First, this is "testing" and not "computing"; second, the function of interest is defined with respect to distributions, but the inputs that the parties use in the computation are samples. These features do not come into play if the distributions satisfy the promise (e.g., they are either identical or ϵ-far), in which case the security guarantee matches the standard cryptographic one (no information is leaked beyond the output). However, the crux is when the promise is not satisfied, in which case we need to allow for some information on the parties' samples to be leaked by the protocol. Our definition permits leakage of at most one bit in this case, and leaks nothing when the promise is satisfied. See the formal definition and discussion in Section 5.

Closeness Testing. In the *2-party closeness testing* problem $2\text{PCT}_{n,t,\epsilon}$, Alice and Bob each have access to t samples from some distributions respectively a, b over alphabet $[n]$. Their goal is to distinguish between $a = b$ and $\|a - b\|_1 \geq \epsilon$ with probability $\geq 2/3$.

We first give a non-secure near-optimal communication protocol, and then show how to make it secure with only a small overhead (polynomial in the security parameter). Our secure version is based on the existence of a PRG that stretches from polylog(m) bits to m bits, and of an OT protocol with polylog communication. We elaborate on these standard cryptographic assumptions in a later section. Overall, we prove the following theorem.

Theorem (Closeness, Secure; see Theorem 5.4)**.** *Fix a security parameter $k > 1$. Fix $n > 1$ and $\epsilon \in (0, 2)$, and let t be such that $t \geq C \cdot k \cdot \max\left(n^{2/3} \cdot \epsilon^{-4/3}, \sqrt{n} \cdot \epsilon^{-2}\right)$ for some (universal) constant $C > 0$. Then, assuming PRG and OT as above, there exists a* secure distribution testing protocol *for $2\text{PCT}_{n,t,\epsilon}$ which uses $\tilde{O}_k\left(\frac{n^2}{t^2\epsilon^4} + 1\right)$ communication.*

To contrast the communication bounds of our protocol to the classic 1-party setting, consider what happens in the extreme settings of the parameters s, t, for a fixed ϵ. When $t \approx \Theta(n^{2/3})$, the communication is $\tilde{O}(n^{2/3})$ as well, i.e., Alice may as well just send all the samples over to Bob. However the communication decreases as the players have more samples. This may not be surprising given the testing results with unequal number of samples [BV15, DK16]: indeed, Alice can send $\approx \max\{n/\sqrt{t}, \sqrt{n}\}$ samples to Bob, and Bob can run the tester. In contrast, our protocol obtains a *polynomially smaller* complexity, $\approx n^2/t^2$, whenever $t \gg n^{2/3}$. Intuitively, considering the extreme of $t \gg n$, we can obtain near-constant communication: with so many samples, we can *learn* the distribution, and then use *sketching* tools, such as the ℓ_1 sketching algorithm of [AMS99, Ind06].

We prove a *near-tight* lower bound on the above communication complexity trade-off (even without security considerations) in Section 4. We note that this lower bound differs from standard communication complexity lower bounds as the players' inputs are *i.i.d. samples* and not worst-case.

Theorem (Closeness lower bound; see Theorem 4.1)**.** *Any two-way communication protocol for $2\text{PCT}_{n,t,1/2}$ requires $s = \tilde{\Omega}\left(\frac{n^2}{t^2}\right)$ communication.*

Independence Testing. The second problem we consider is the independence testing problem in the 2-party model, denoted $2\text{PIT}_{n,m,t,\epsilon}$. Let $p = (a, b)$ be some joint distribution over $[n] \times [m]$, where $n \geq m$, and for $i \in [t]$, let ζ_i be a sample drawn from p. Now we provide Alice with the first coordinates of ζ_i's and Bob with the second coordinates. Alice and Bob's goal is to test whether p is a product distribution or ϵ-far from any product distribution. We prove the following:

Theorem (Independence, Secure; see Theorem 6.1)**.** *Fix a security parameter $k > 1$. Fix $\epsilon \in (0, 2)$, $1 \le m \le n$, and let t be such that $t \ge C \cdot k \cdot \left(n^{2/3} m^{1/3} \epsilon^{-4/3} + \sqrt{nm}/\epsilon^2\right)$, for some (universal) constant C, and assuming OT, there is a secure distribution testing protocol for $2\mathrm{PIT}_{n,m,t,\epsilon}$ using $\tilde{O}_k\left(\frac{n^2 \cdot m}{t^2 \epsilon^4} + \frac{n \cdot m}{t \epsilon^4} + \frac{\sqrt{m}}{\epsilon^3}\right)$ bits of communication.*

We note that the lower bound on t from the above theorem is necessary as it is the information-theoretic bound, as proven in [DK16].

An important qualitative aspect of the communication complexity for 2PIT is that, when the number of samples $t \to \infty$, the protocol uses $\tilde{\Theta}_\epsilon(\sqrt{m})$ bits communication. This is in contrast to 2PCT, where the communication becomes poly-logarithmic for $t \to \infty$. Indeed, we show that $\Omega(\sqrt{m})$ communication is necessary for one-way communication for 2PIT. Since our non-secure protocol can be made one-way (see Remark 6.6), this lower bound is tight for one-way protocols. We conjecture that the bound on s from the above theorem is near-tight in n, m, t for 2-way communication protocols, even without security.

Theorem (Independence lower bound; see Theorem 7.1)**.** *For $n, t \in \mathbf{N}$, any one-way protocol for $2\mathrm{PIT}_{n,n,t,1}$ requires $\Omega(\sqrt{n})$ bits of communication.*

1.2 Related work

Our work bridges three separate areas and models: distribution testing, streaming/sketching, and secure computation. There's a large body of work in each of these areas, and we mention the branches most relevant to us.

Distribution Testing. The most basic problem for distribution testing is identity testing: where one distribution (say, a) is fixed/universal (e.g., uniform). There is a large body of work on this problem for general a, starting with [BFF+01]; see surveys [Gol17, Can15, Rub12]. As this problem is fundamentally about one unknown distribution a, we do not consider it in the 2-party model.

For the closeness testing problem, in addition to the aforementioned references, we highlight the result of [DK16] (see also [Gol16]), whose techniques are the starting point of our protocols. That result introduces a clean framework for testing, by reducing ℓ_1 testing to ℓ_2 testing and checking if two distributions are the same or ϵ-far from each other with respect to the ℓ_2 norm.

The independence testing problem has been studied in [BFF+01, LRR13, ADK15, DK16]. In the standard setting (1 party) the problem is defined as: given samples from a distribution π over pairs with marginals a, b, determine whether a, b are independent ($\pi = a \times b$) or π is ϵ-far from being a product distribution. This line of work culminated in the work of [DK16], who show the tight sample complexity of $\Theta(\sqrt{nm}/\epsilon^2 + n^{2/3} m^{1/3}/\epsilon^{4/3})$, where $n \ge m$ are the cardinalities of the two sets.

Other related questions include: testing over d-tuples of distributions [LRR13, LRR14, ADK15, DK16], testing independence of monotone distributions [BKR04], and testing k-wise independence [AAK+07, RX10, RX13]. To what extent the samples–communication trade-off from this paper extend to these problems is an interesting open question.

Connection to communication and sketching. As already alluded to, our 2-party testing setup turns out to be connected to sketching. In particular, for the closeness problem, when the number of samples is $t \approx n$, Alice and Bob can approximate respectively a, b (up to small ℓ_1 distance), and then just estimate $\|a - b\|_1$ using ℓ_1 sketching such as [Ind06], using *constant* communication.

More generally, the 2-party communication model is intimately connected to the *streaming model*, where the input is streamed over, while keeping small extra space for computation. Most relevant here is the work where the input is randomized in some way — this is similar to our setting, where the input consists of *samples* from a distribution. This includes the recent work on *stochastic streaming* [CMVW16], where the input is generated from a distribution. In particular, [CMVW16] analyze the space complexity for estimating the ℓ_2 norm of a probability distribution a given t samples presented as a stream. They show a trade-off between space s and number of samples $t \geq \sqrt{n}$ of $s \cdot t = \tilde{O}_\epsilon(n)$. They also show this is tight under a likely conjecture on the space complexity of estimating frequency moments in the *random order* streaming model.

For the independence problem, two variants were studied in the streaming model in [IM08, BO10]. In the first variant, termed *centralized model*, we stream over pairs $\zeta = (x, y)$, which implicitly define a joint distribution (a, b), and need to compute the distance between the joint distribution and the product-of-the-marginals distribution. Space complexity of this version is $(\log n)^{O(1)}$ [BO10]. The second variant, termed *distributed model*, is more similar to ours: we stream over items (t, i, p), where t is the time (index) of a sample $(x_t, y_t) \sim (a, b)$ and $i \in [n]$ is either x_t (if $p = 1$) or y_t (if $p = 2$), and the goal is again to estimate the distance between (a, b) and product distribution. For the latter problem, [IM08] show the complexity is between $\Omega(n)$ and $\tilde{O}_\epsilon(n^2)$.

In the context of connections between communication complexity and property testing, we also mention the work of [BBM12, BCG16]. They use communication tools to prove lower bounds on the *sample complexity* of testing problems. Their techniques however do not readily apply to proving *communication complexity* of (distribution) testing problems as we do here.

Testing and learning in distributed and streaming models. Communication lower bounds are also often related to those in the streaming and distributed models, which have received recent focus in the context of testing and learning questions. In particular, streaming (as a memory constraint) was considered as early as in [Cov69, HC70], for distribution (hypothesis) testing of one distribution. In recent years, a lot of attention has been drawn to streaming (memory) lower bounds for learning problems, such as parity learning [Raz16, Raz17, KRT17, MM17, GRT18]. These results also show a trade-off between number of samples and space complexity.

Another recent avenue is to study such problems in the distributed model, where there are many symmetric players, each with a number of i.i.d. samples from the same distribution. For learning problems (e.g., parameter or density estimation), see, e.g., [BGM$^+$16, DGL$^+$17, DS18]. We note that since learning is a much harder problem, typically proving lower bounds is easier (e.g., as shown in [DGL$^+$17], merely communicating the output requires $\Omega(n)$ communication). In contrast, for testing problems, the output is just one bit. For testing problems, see the recent (independent) manuscript [ACT18], who studies testing of one distribution, and focuses on reducing the communication per sample (from a max of $O(\log n)$).

None of the lower bounds from the above papers seem to be relevant here as they become vacuous for a 2-party setting. Indeed, when two players have two set of samples from the *same* distribution, then purely doubling the sample set of a player trivializes the question (she can solve it without communication).

Secure approximations. Our results on secure distribution testing can also be seen in the context of the area of secure computation of approximations. This is a framework introduced by [FIM$^+$06], allowing to combine the benefits of approximation algorithms and secure computation. This was considered in different settings [FIM$^+$06, HKKN01, BCNW08, IW06, BHN09, IMSW09, KMSZ08], but the most relevant to us is private approximation of distance between two input vectors. In particular, for ℓ_2 distance, Alice and Bob each have a vector $a, b \in \mathbb{R}^n$ and want to estimate $\|a - b\|_2$, without revealing any information that does not follow from the ℓ_2 distance itself. For this problem, [IW06] show that secure protocols are possible with only poly-logarithmic communication complexity. They also specifically look into the private near-neighbor problem and its approximation. We use some of their techniques in our secure protocols.

Approximation and testing have a similar flavor in that they both trade accuracy for efficiency, in different ways (computing an estimate of the solution in the former case, and computing the correct output bit if the promise is satisfied in the latter). The security goals are also similar (prevent leakage beyond the intended output). One important difference is that the intended output in secure testing is just the single bit of whether or not the test passed. Thus, for example, when approximating a distance function, even a secure protocol can leak any information that follows from the distance. In contrast, when testing for closeness, if the inputs are either identical or far, the protocol may only reveal this fact, but no other information about what the distance is.

Security and privacy of testing. While we are not aware of any work on secure testing, several recent papers address *differentially-private distribution testing* [DHS15, CDK17, ASZ17, ADR17]. Here the privacy guarantee relates to the value of the output after the computation is concluded, requiring it to be differentially-private with respect to the inputs. Our notion of security for distribution testing is different, in the same way that secure computation is different from differentially private computation. While differential privacy (DP) is concerned with what the intended output may leak about the inputs (even if the input came from a single party or the computation is done by a trusted curator), secure 2-party computation is concerned with how to compute an intended output without leaking any information beyond what the output itself reveals. The difference in goals also results in the difference that DP testing privacy guarantees are typically statistical in nature and provide a non-negligible adversarial advantage, while secure testing protocols rely on cryptographic assumptions and provide negligible advantage.

More recently, a more stringent model of Locally Differentially Private Testing was proposed [She18, ACFT18]. This model provides a stronger notion of differential privacy, where users send noisy samples to an untrusted curator, and the goal is to allow the curator to test the distribution of user inputs (for some property) without learning "too much" about the individual samples. For LDP, the main goal is to optimize the sample complexity as a function of the privacy guarantees. While this notion of privacy also incorporates some privacy of the individual inputs, it is much closer to DP than to our security notion. In addition, both DP and LDP does not provide sub-linear communication (in the sample size). In fact, their goal is to allow $O(1)$ communication per sample, with minimal sample overhead. In contrast, our protocols provide security "nearly for free" while allowing for faster communication with more samples. Finally, in the case of independence testing, our work assumes samples are distributed between the parties who need to test the joint distribution, while in the aforementioned work, each data point contains full sample information.

We also mention the work of [MNS11] on *sketching in adversarial environments*, which considers another threat model: here the two parties performing the computation are honest (and trust each other), but their (large) input arrives in a streaming fashion chosen adversarially, and they may only maintain a sketch of the inputs.

1.3 Our Techniques

We now outline the techniques used to establish our main results. Since our overall contribution is painting a big picture of the 2-party complexity of distribution testing problems, we appeal to a number of diverse tools. First, we design communication-efficient protocols whose communication improves as we have more samples. Second, we show that some of the established trade-offs are near-optimal by proving communication complexity lower bounds on the considered problems, which are near-tight in some of the parameter regimes. Third, we show how to transform our protocols into secure protocols, under standard cryptographic assumptions, without further loss in efficiency. All three of these contributions are independently first-of-a-kind, to the best of our knowledge.

Communication-efficient protocols. We start by noting that we can reduce the testing problem under the ℓ_1 distance (total variation distance) to the same problem under the ℓ_2 distance, using now-standard methods of [DK16, CDVV14]. Although this reduction introduces a few complications to deal with, our main challenge is actually testing under the ℓ_2 distance.

Closeness testing (2pCT) is technically the simpler problem, but it already illustrates some phenomena, in particular, how to leverage a larger number of samples to improve communication. To estimate the ℓ_2 distance between the 2 unknown distributions, we compute the ℓ_2 distance approximation between the given *samples* of these distributions. In order to approximate the latter in the 2-party setting, we use the ℓ_2 *sketching* tools [AMS99]. The crux is to show that we can tolerate a cruder ℓ_2 approximation if we are given a larger sample size. Since the complexity of $(1+\alpha)$-approximating the ℓ_2 distance is $\Theta(1/\alpha^2)$, we obtain an improvement in communication that is quadratic in the number of samples.

Independence Testing (2PIT) is more challenging since any distance approximation would need to be established based on the distribution(s) implicitly defined via the *joint samples*, split between Alice and Bob, and hence our approximation techniques above is not sufficient. Instead, we develop a reduction from a large, $[n] \times [m]$, alphabet problem, to a smaller alphabet problem, which can be efficiently solved by communicating fewer samples. This is accomplished by sampling a rectangle of the joint alphabet, and showing that such a process, when combined with the *split-set* technique from [DK16], generates *sub-distributions* (defined later) which satisfy some nice properties. We then show one can test the original distribution $p = (a, b)$ over a "large" domain of size $[n] \times [m]$ for independence by distinguishing closeness of 2 simulated distributions $\hat{p}, \hat{q}$, defined on a smaller domain of size $[l] \times [m]$, where $l = \tilde{\Theta}_\epsilon(n^3 m/t^3 + n^2 m/t^2 + 1)$. We show it is possible for Alice and Bob to simulate joint samples from $\hat{p}$ and $\hat{q}$ using $O(1)$ communication per sample, after they have down-sampled letters from one of the marginals.

The trade-off on communication–vs–samples emerges from two compounding effects: 1) balancing the size of the target rectangle with the expected number of available samples over such rectangle; and 2) the additional advantage from a tighter bound on the ℓ_2 norm of $\hat{q}$. Each of the above independently generates linear improvement in communication with more samples. The latter advantage, however, is helpful only while $t = O(n)$, and therefore we benefit from quadratic improvement in that regime, and linear improvement thereafter.

Lower bounds on communication. We note that the lower bounds on communication of testing problems present a particular technical challenge: for testing problems, the inputs are i.i.d. samples from some distributions. This is more akin to the average-case complexity setup, as opposed to "worst case" complexity as is standard for communication complexity lower bounds.

We manage to prove such testing lower bounds for the *Closeness Testing* problem (2PCT). While our lower bound is, at its core, a reduction from some "hard 2-party communication problem", our main contribution is dealing with the above challenge. One may observe that a "hard 2-party communication problem" is hard under a certain input distribution (by Yao's minimax theorem), and hence a reduction algorithm would also produce a hard distribution on the inputs to our problem. However, apriori, it is hard to ensure that the resulting input distribution resembles anything like a set of samples from distributions a, b. For example, the inputs may have statistical quirks that actually depend on whether it is a "close" or "ϵ-far" instance, which a reduction is not able to generate without knowing the output. Indeed, this is the major technical challenge to overcome in our reduction.

At a high level, the role of the "hard problem" is played by a variant of the well-known two-way Gap Hamming Distance (GHD) problem [CR12, She12]. The known GHD lower-bound variants are insufficient for us precisely because of the above challenge—we need a better control over the actual hard distribution, and in particular, for the $a = b$ instance, we need the GHD input vectors to be at some fixed distance. Therefore, we study the following *Exact GHD* variant: given $x, y \in \{0,1\}^n$, with $\|x\|_1 = \|y\|_1 = n/2$, distinguish between $\|x - y\|_1 = n/2$ versus $\|x - y\|_1 \in [n/2 + \beta, n/2 + 2\beta]$. We show there exists some $\beta \in [\Omega(\sqrt{n}), O(\sqrt{n \log n})]$ for which communication complexity must be $\tilde{\Omega}(n)$, by adapting the proof of [She12]. We note that such a lower bound is far from apparent even in the 1-way communication model (in contrast to the standard GHD, whose 1-way lower bound follows immediately from the Indexing problem).

Using one instance of *Exact GHD*, our reduction performs a careful embedding of this hard instance into the samples from distributions a, b, while patching the set of samples to look like i.i.d. samples from the two distributions. While we don't manage to get the output of the reduction to look precisely like i.i.d. samples from a, b, our reduction produces two sets of size $\mathrm{Poi}(t)$ whose distribution is within a small statistical distance from the distribution of two set of samples that would be drawn from two distributions a and b which are either "equal" (when $\|x - y\|_1 = n/2$) or "far" (when $\|x - y\|_1 \in [n/2 + \beta, n/2 + 2\beta]$).

We note that an immediate corollary of our lower bound is an alternative proof that $\tilde{\Omega}(n^{2/3})$ samples are necessary to solve closeness testing (in the vanilla testing setting), albeit the resulting bound is off by polylogarithmic factors from the best known ones [CDVV14, DK16].

For *Independence testing* (2pIT), we focus on the lower bound for unbounded number of samples, which is a bit simpler to deal with as it becomes a "worst-case problem" (worst-case input joint distributions (a, b)). We are able to show such a hardness result under one-way communication only. Our $\Omega(\sqrt{m})$ lower bound uses the Boolean Hidden Hypermatching (BHH) problem [VY11]. From a single BHH hard instance, we generate an unbounded number of sample pairs (A_i, B_i), where the samples are drawn from some joint distribution $p = (a, b)$. Depending on the BHH instance, p is either exactly product or far from product distribution.

We conjecture that our entire trade-off for the Independence problem is tight. The proof of this conjecture would have to overcome the above challenge of lower bounds for statistical inputs, for finite t.

Securing the communication protocols. Once low-communication insecure protocols have been designed, one may try to convert the protocols to secure ones using generic cryptographic techniques. The latter includes various techniques for secure two-party computation ([Yao82] and followup work), fully homomorphic encryption ([Gen09] and followup work), or homomorphic secret sharing ([BGI16, BGI17]). However, a naïve application of such techniques will blow up the communication bounds to be at least linear in the input size (which is prohibitive in our context), as well as possibly requiring strong assumptions, a high computation complexity, or not being applicable to arbitrary computations. The constraint of low-overhead, combined with other considerations, requires design of custom secure protocols.

Our starting point is a technique that falls into the latter category: secure circuits with ROM [NN01], a technique that can transform an insecure 2-party protocol to a secure one with a minimal blow-up in communication, and uses a weak assumption only (OT). In order to obtain an efficient protocol, however, it only applies to computations expressible via a very small circuit, whose size is proportional to the target communication, with access to a larger read-only-memory (ROM) table. Thus, the main challenge becomes to design two-party testing protocols that fit this required format.

For *Closeness Testing* (2pCT), we begin with our low-communication non-secure protocol, and adapt it to be secure by designing a small circuit. One of the main difficulties in designing such a circuit is that, in the ℓ_1- to ℓ_2-testing reduction, Alice and Bob need to agree on an alphabet, which depends on their inputs, without compromising the inputs themselves. To bypass this, and other issues, we allow Alice and Bob to perform some off-line work and prepare some polynomial-size inputs (in ROM). First, we devise a method for Alice and Bob to generate a combined split set S (discussed later) by having each of Alice and Bob contribute sampled letters to S. Second, we securely approximate the ℓ_2 distance of Alice and Bob's original, un-splitted samples using techniques from [IW06]. Finally, we adjust our approximation by accounting for some small number of letters which differ from the original alphabet or which cannot be approximated efficiently. To allow for such adjustment to be easily accessible by a bounded-size circuit, Alice and Bob prepare inputs for all possible scenarios. The main focus of our analysis goes into proving our construction adds only poly-logarithmic factors in communication over the non-secure protocol. Our adapted secure algorithm turns out to deviate significantly from the simpler, non-secure 2pCT algorithm; hence we present the two protocols separately.

For *Independence Testing* (2pIT), the main challenge is in designing the communication-efficient (insecure) protocol, while adapting it to a secure one is somewhat simpler. One challenge is to securely accomplish the aforementioned alphabet reduction (as the randomness, together with the final output, might compromise Bob's input). We address this issue by having the entire process of sampling a sub-distribution and pairing of samples to be done over a secure circuit. The sampled alphabet, however, might be too large to communicate, and to overcome this obstacle, we devise a method for sampling from the non-empty letters of Alice in a way that is distributed as if we would be sampling directly over the entire alphabet. For this problem, the insecure and secure protocol are more similar, and therefore we directly describe the secure version.

2 Preliminaries

Notation. Throughout this paper we denote distributions in small letters, and distribution samples in capital letters. Unless stated otherwise, any distribution is on alphabet $[n]$, and domain elements of $[n]$ are addressed as letters. In addition, whenever discussing secure protocols, we denote distributions as a, b, and for generic settings as p, q. Similarly, secure vectors of distribution samples are denoted as A, B while generic ones are denoted as X, Y.

We also denote any multiplicative error arising from approximation as $1+\alpha$, and any error or distance of/between distributions as ϵ. Unless stated otherwise, distance and norms are referring to the Euclidean distance and ℓ_2 norms.

Split Distributions. We use the concept of split distributions from [DK16] to essentially reduce testing under ℓ_1 distance to testing under ℓ_2 distance.

Definition 2.1. *Given a probability distribution p on $[n]$ and a multiset S of items from $[n]$, define the* split distribution p_S *on $[n+|S|]$ as follows. For $i \in [n]$, let a_i be equal to 1 plus the number of occurrences of i in S; note that $\sum_{i=1}^{n} a_i = n + |S|$. We associate the elements of $[n+|S|]$ to elements of the set $E = \{(i,j) : i \in [n], 1 \le j \le a_i\}$. Now the distribution p_S has support E and a random draw (i,j) from p_S is sampled by picking i randomly from p and j uniformly at random from $[a_i]$.*

Recall from [DK16] that split distributions are used to upper bound the ℓ_2 norm of an underlying distribution while maintaining its ℓ_1 distance to other distributions:

Fact 2.2 ([DK16])**.** *Let p and q be probability distributions on $[n]$, and S a given multiset of $[n]$. Then*

- *We can simulate a sample from p_S or q_S by taking a single sample from p or q, respectively.*
- $\|p_S - q_S\|_1 = \|p - q\|_1$.

Lemma 2.3 ([DK16])**.** *Let p be a distribution on $[n]$. Then: (i) For any multisets $S \subseteq S'$ of $[n]$, $\|p_{S'}\|_2 \le \|p_S\|_2$, and (ii) If S is obtained by taking $Poi(m)$ samples from p, then $\mathbb{E}[\|p_S\|_2^2] \le 1/m$.*

We provide further preliminaries needed for the secure versions of our protocols in Section 5.1.

3 Closeness Testing: Communication-Efficient Protocol

In this section we consider the closeness testing problem 2pCT, focusing first on the 2-party communication complexity only. In Section 5, we modify the protocol to make it secure.

As mentioned in the introduction, one way to obtain a protocol is to use unequal-size closeness testing, where Alice has s samples and Bob has t samples: Alice just sends her s samples to Bob, and Bob invokes a standard algorithm for closeness testing. Using the optimal bounds from, say, [DK16], we get the following trade-off for fixed ϵ: $s = \tilde{O}(n/\sqrt{t})$, with the condition that $s, t \geq \sqrt{n}$.

Here we obtain a *polynomially smaller* communication complexity, $s = \tilde{O}_\epsilon(n^2/t^2)$, whenever t is above the information-theoretic minimum on the number of samples. In Section 4, we show a nearly-matching lower bound.

3.1 Tool: approximation via occurrence vectors

Our protocol uses the framework introduced in [DK16], allowing us to focus on the ℓ_2 testing problem. For ℓ_2 testing, we show that, for two discrete distributions p, q, we can approximate their ℓ_2 distance by approximating the ℓ_2 distance of their respective sample occurrence vectors, defined as follows.

Definition 3.1. *Given t samples of some distribution p over $[n]$, we define* the occurrence vector $X \in [t]^n$ *such that X_i represent the number of occurrences of element $i \in [n]$ in the sample set.*

The following lemma bounds how well we need to estimate the ℓ_2 distance between occurrence vectors in order to distinguish between $p = q$ vs. $\|p - q\|_1 \geq \epsilon$. Overall it shows that the more samples we have, the less accurate the ℓ_2 estimation needs to be. Using the framework from [DK16], for now it is enough to assume that the ℓ_2 norm of both p and q is bounded by $U < 1$.

Lemma 3.2. *Let p, q be distributions over $[n]$ with $\|p\|_2, \|q\|_2 \leq U$ for some $U < 1$. There exists a constant $C > 0$, such that for $t = O(U \cdot n \cdot \epsilon^{-2})$, and $\alpha = \Omega(U)$, given Δ which is $(1 \pm \alpha)$-factor approximation of $\|X - Y\|_2^2$ where X, Y represent the occurrence vectors of t samples drawn from p, q respectively, then, using Δ, it is possible to distinguish whether $p = q$ versus $\|p - q\|_1 > \epsilon$ with 0.8 probability.*

The actual distinguishing algorithm is simple: for fixed $\alpha = \Omega(U)$, we merely compare Δ to some fixed threshold τ (fixed in the proof below). The intuition is that for a given number of samples, we have some gap between the range of possible distances $\|X - Y\|_2^2$ for each of the cases. If the number of samples is close to the information-theoretic minimum [CDVV14], then the gap is minimal and we need to calculate almost exactly the distance, hence approximating the distance between occurrence vectors doesn't help. However, as the number t of samples increases, so does gap between the ranges, and we can afford a looser distance approximation.

Proof of Lemma 3.2. Given $t = O(U/\epsilon'^2)$ samples from each p, q, according to [CDVV14, Proposition 3.1], the estimator $\frac{\sqrt{\sum_i (X_i - Y_i)^2 - X_i - Y_i}}{t}$ is a $\max\{\epsilon', \|p - q\|_2 / 8\}$ additive approximation of $\|p - q\|_2$ with 0.9 probability. Setting $\epsilon' = \epsilon/8\sqrt{n}$, we get that:

$$\|p - q\|_2 = 0 \Rightarrow \|X - Y\|_2^2 \leq \frac{\epsilon^2 t^2}{4n} + 2t$$

$$\|p - q\|_2 > \epsilon/\sqrt{n} \Rightarrow \|X - Y\|_2^2 \geq \frac{3\epsilon^2 t^2}{4n} + 2t$$

Suppose Δ is such that $\frac{\Delta}{\|X - Y\|_2^2} \in (1 - \alpha, 1 + \alpha)$. If $\|p - q\|_1 = 0$, then $\|p - q\|_2 = 0$ and hence

$$\Delta \leq (1 + \alpha)(\frac{\epsilon^2 t^2}{4n} + 2t) \leq t(\frac{\epsilon^2 t}{4n} + 2 + 2\alpha + \frac{\alpha \epsilon^2 t}{4n});$$

and if $\|p - q\|_1 > \epsilon$, then $\|p - q\|_2 > \epsilon/\sqrt{n}$ and hence

$$\Delta \geq (1 - \alpha)(\frac{3\epsilon^2 t^2}{4n} + 2t) \geq t(\frac{3\epsilon^2 t}{4n} + 2 - 2\alpha - \frac{3\alpha \epsilon^2 t}{4n}).$$

We distinguish the two cases, by comparing Δ to $\tau = \frac{\epsilon^2 t^2}{2n} + 2t$: namely $p = q$ iff $\Delta \leq \tau$. Hence we just need to ensure that

$$t(\frac{3\epsilon^2 t}{4n} + 2 - 2\alpha - \frac{3\alpha\epsilon^2 t}{4n}) - \tau \geq \tau - t(\frac{\epsilon^2 t}{4n} + 2 + 2\alpha + \frac{\alpha\epsilon^2 t}{4n})$$

Hence we need that $\frac{\epsilon^2 t}{4n} - 2\alpha - \frac{3\alpha\epsilon^2 t}{4n} \geq 0$, or $\alpha \leq \frac{\epsilon^2 t}{4n\cdot(2+3\epsilon^2 t/4n)}$. Since $t = O(Un\epsilon^{-2})$, the conclusion follows. $\square$

3.2 Communication vs number of samples

We now provide a (non-secure) protocol for 2pCT with a trade-off between communication and number of samples.

Theorem 3.3 (Closeness, insecure)**.** *Fix $n > 1$ and $\epsilon \leq 2$. There exists some constant $C > 0$ such that for all $t \geq C \cdot \max\left(n^{2/3} \cdot \epsilon^{-4/3}, \sqrt{n} \cdot \epsilon^{-2}\right)$, the problem $2\text{pCT}_{n,t,\epsilon}$ can be solved using $\tilde{O}\left(\frac{n^2}{t^2\epsilon^4} + 1\right)$ bits of communication.*

The protocol uses Lemma 3.2 as the main algorithmic tool and proceeds as follows. Bob generates multi-set S using samples from b and sends S to Alice. Then, Alice and Bob each simulates samples from a_S and b_S respectively, and they together approximate the ℓ_2 difference of the resulting occurrence vectors using sketching methods [AMS99].

Non-Secure 2pCT(a, b, t)
Alice's input: t samples from a
Bob's input: t samples from b

1. Fix $\alpha = \Omega(t \cdot \epsilon^2/n)$.
2. Bob generates multi-set S using $Poi(\frac{n^2}{t^2\epsilon^4})$ samples from b.
3. Bob sends S to Alice.
4. Alice and Bob recast their samples as being from distributions a_S, b_S (see Def. 2.1), and set A_S, B_S to be the respective occurrence vectors.
5. Alice and Bob each estimate $\|a_S\|_2$ and $\|b_S\|_2$ up to factor 2; if the two estimates are not within factor 4, output "ϵ-FAR";
6. Alice and Bob approximate $\Delta = \|A_S - B_S\|_2^2$ up to $(1 + \alpha)$ factor, using, say, [AMS99].
7. If Δ is less than $\tau = \frac{\epsilon^2 t^2}{2n} + 2t$ output "SAME", and, otherwise, output "ϵ-FAR".

Proof of Theorem 3.3. We note that, according to Lemma 2.3, $\mathbb{E}[\|b_S\|_2^2] = t^2\epsilon^4/n^2$ and hence $\|b_S\|_2^2 = O(t^2\epsilon^4/n^2)$ with at least 90% probability. Furthermore, since $t = \Omega(\sqrt{n}/\epsilon^2)$, we have that $|S| = O(n)$ with high probability. From now on, we condition on these two events.

If $\|a_S\|_2 \neq \Theta(\|b_S\|_2)$ then distributions are different and we output "ϵ-far" is step 5. Otherwise, we have that $\|a_S\|_2^2 = O(\|b_S\|_2^2) = O(t^2\epsilon^4/n^2)$. Hence we can use Lemma 3.2, where $U = O(t\epsilon^2/n)$ and $\alpha = \Omega(U)$, to claim the correctness of the protocol.

We now analyze the communication used by the protocol:

1. communicating S takes $|S| \log n = \tilde{O}(n^2/t^2\epsilon^4)$ bits with high probability.
2. estimating Δ up to approximation $1+\alpha$ takes $\tilde{O}(1/\alpha^2) = \tilde{O}(n^2/t^2\epsilon^4)$ bits, using standard ℓ_2 estimation algorithms [AMS99, KOR00].

$\square$

Remark 3.4. *Another application of this protocol is that it can be simulated by a* single party *to obtain space-bound* steaming algorithm *with the same space/sample trade-offs. While we are not formalizing this argument in this paper, this can essentially be done by storing S and sketching $\|A_S - B_S\|_2^2$.*

4 Closeness Testing: Communication Lower Bounds

We now prove that the protocol for 2pCT from Section 3 is near-tight, showing the following theorem:

Theorem 4.1. *Let a, b be some distributions over alphabet $[n]$, where Alice and Bob each receive* $\mathrm{Poi}(t)$ *samples from a, b respectively, for $t \le n/\log^c n$ for some large enough $c > 1$. Then any (two-way) communication protocol Π that distinguishes between $a = b$ and $\|a-b\|_1 \ge 1/2$ requires $s = \tilde{\Omega}(n^2/t^2)$ communication.*

Intuitively, our proof formalizes the concept that in testing distributions for closeness, "collisions is all that matters", *even in the communication model.* This is similar to the intuition from the "canonical tester" from [Val11], which shows a similar principle when all the samples are accessible. Our result can be seen to extending it to saying that the canonical tester is still the best even if we have more-than-strictly-necessary number of samples that we could potentially compress in a communication protocol.

To prove the theorem, we rely on the following communication complexity lower bound, which is a variant of the GapHamming lower bound [CR12, She12]. A somewhat surprising aspect of this GapHamming variant is that, unlike for the standard GapHamming, we are not aware of a lower bound for one-way communication that would be simpler than the two-way proof from the lemma below.

Lemma 4.2. *Let $n \ge 1$ be even. There exists some $\beta = \beta(n) \in [\Theta(\sqrt{n}), \Theta(\sqrt{n \log n})]$, satisfying the following. Consider a two-way communication protocol $\mathcal{A}$ that, with probability at least* 0.9, *for $x, y \in \{0,1\}^n$ with $\|x\|_1 = \|y\|_1 = n/2$, can distinguish between the case when $\|x-y\|_1 = n/2$ versus $\|x-y\|_1 - n/2 \in [\beta, 2\beta]$. Then $\mathcal{A}$ must exchange at least $\Omega(\frac{n}{\log n \cdot \log\log n \cdot \log\log\log n})$ bits of communication.*

The proof of this lemma is presented in Appendix A.

Now the idea is to reduce an instance of the GapHamming input from Lemma 4.2 to an instance of closeness testing by carefully molding the input (x, y) into a couple of related occurrence vectors (A, B).We use the following estimate on the statistical distance between Multinomial and Poisson random variables.

Definition 4.3. *Consider $n, k \ge 1$, as well as a vector $\vec{p} \in \mathbb{R}_+^k$, where $\sum_{i=1}^k p_i \le 1$. Then let $(M_1, \dots M_k) = \mathrm{Mult}_{\text{-}0}(n; \vec{p})$ be the k-dimensional random variable obtained by drawing a Multinomial random variable with parameters n and probability vector $(1 - \sum_{i=1}^k p_i, \vec{p})$, and dropping the first coordinate.*

Theorem 4.4 ([Bar88])**.** *Let $n, k \ge 1$, as well as a vector $\vec{p} \in \mathbb{R}_+^k$, where $p = \sum_{i=1}^k p_i \le 1$. Consider the random variable $(M_1, \dots M_k)$ to drawn from the Multinomial* $\mathrm{Mult}_{\text{-}0}(n; \vec{p})$*. Also consider the Poisson random variable $P = (P_1, \dots, P_k)$ where $P_i \sim$* $\mathrm{Poi}(np_i)$*. Then the variables $M = (M_1, \dots M_k)$ and $(P_1, \dots P_k)$ are at a statistical distance of $O(p \log n)$.*

Proof of Theorem 4.1. Consider some input vectors x, y, of length $m = \frac{n^2}{t^2 \log^3 n}$, to the GapHamming problem from above. Let $\Delta = \beta(m) = \Omega(\sqrt{m})$, and $\delta = \frac{1}{2}(\|x - y\|_1 - m/2) \in \{0\} \cup [\Delta/2, \Delta]$. The case of $\delta = 0$ will correspond to "same" case (i.e. $a = b$), and $\delta \in [\Delta/2, \Delta]$ — to "far" case (i.e. $\|a - b\|_1 \in [1/2, 1]$).

Fix $d = n/10$ and $l = C \cdot t \cdot \log n$ (where C is some constant that we shall fix later), which have the following meaning: each distribution a, b has half mass over $[d]$ items uniformly (called *dense items*), and the other half on $[l]$ items uniformly (called *large items*). When $a = b$, these are the same items, and when $a \neq b$, the large items are the same while the dense items have supports with a large difference. In particular, the dense items are supported on sets S_A, S_B respectively, with $|S_A| = |S_B| = d$, and $S_A \cap S_B = d \cdot \frac{\Delta - \delta}{\Delta}$; we hence also have that $|S_A \setminus S_B| = d \cdot \frac{\delta}{\Delta}$.

Now for $i \geq 0$, let $D(i) = \Pr[\mathrm{Poi}(t/2d) = i]$, i.e., probability that a dense number is sampled i times. For simplicity, we write $D(i,j) = D(i) \cdot D(j)$. Similarly we define $L(i) = \Pr[\mathrm{Poi}(t/2l) = i]$ and $L(i,j) = L(i) \cdot L(j)$. We also set $k = \Theta(\log n)$, which should be thought of as an upper bound on the count of any fixed item (with high probability).

The algorithm constructs the occurrence vectors A, B iteratively as follows. We note that all random variables are chosen using shared randomness. Let $m_c = m/4 - \Delta$.

1. For each $i, j \in \{1, \dots k\}$, and for each letter $c \in [n]$, we generate $\mathrm{Poi}(\frac{d}{\Delta} \cdot D(i,j))$ copies of letter c: Alice replaces 1 with i and Bob replaces 1 with j (both leaving 0s intact);

2. For each $i \in \{1, \dots k\}$, generate $\mathrm{Poi}(d \cdot D(i, 0))$ pairs $(i, 0)$, and similarly-distributed number of pairs $(0, i)$;

3. For each $i, j \in \{1, \dots k\}$, generate $\mathrm{Poi}(l \cdot L(i,j) - m_c \cdot \frac{d}{\Delta} D(i,j))$ pairs (i,j);

4. For each $i \in \{1, \dots k\}$, generate $\mathrm{Poi}(l \cdot L(i,0) - \frac{m}{4} \cdot \frac{d}{\Delta} \sum_{j=1}^{k} D(i,j))$ pairs $(i, 0)$, and similarly-distributed number of pairs $(0, i)$.

5. Fill in the required number of $(0,0)$ pairs so that A, B have length precisely n;

6. Randomly permute the letters of A, B (using shared randomness).

Claim 4.5. *All the Poisson random variables from above are properly defined—in particular, they have positive argument.*

Proof. We only need to prove this for steps 3 and 4 as the other ones are obvious. Indeed, for $i, j \geq 1$:

$$l \cdot L(i,j) = t \log n \cdot (\Omega(1/\log n))^{i+j} = t/\log n \cdot (\Omega(1/\log n))^{i+j-2},$$

whereas,

$$m_c \cdot \frac{d}{\Delta} D(i,j) = O(\sqrt{m} \cdot n \cdot (t/2d)^{i+j}) \leq \frac{n^2}{t \log^{1.5} n} \cdot O(t^2/n^2) \cdot (O(t/n))^{i+j-2} \leq \frac{t}{\log^{1.5} n} (O(t/n))^{i+j-2}.$$

Thus $l \cdot L(i,j) - m_c \cdot \frac{d}{\Delta} D(i,j) \geq 0$ for all $i, j \geq 1$.

Similarly, for step 5, for $i \geq 1$, we have:

$$l \cdot L(i, 0) = \Omega(t \cdot (O(1/\log n))^{i-1}),$$

whereas,

$$m/4 \cdot \frac{d}{\Delta} \sum_{j \geq 1} D(i,j) \leq O(\sqrt{m} \cdot n \cdot \sum_{j \geq 1} (O(t/n))^{i+j}) \leq O(\frac{n^2}{t \log^{1.5} n} \cdot (O(t/n))^{i+1}) \leq O(\frac{t}{\log^{1.5} n} \cdot (O(t/n))^{i-1}).$$

We again have $l \cdot L(i,0) - m/4 \cdot \frac{d}{\Delta} \sum_{j \geq 1} D(i,j) \geq 0$ as required. $\square$

We now prove the core of the reduction: that the distribution of (A, B) is close to occurrence vectors of $\mathrm{Poi}(t)$ i.i.d. samples from (a, b), such that $a = b$ if $\|x - y\|_1 = m/2$, and similarly, $\|a - b\|_1 \geq 1/2$ when $\|x - y\|_1 \geq m/2 + \beta$. We will prove that, for distribution of (co-)occurrences of large items is nearly same in the two instances; and similarly for the dense items.

We partition the coordinates of (x, y) in the following four groups, each corresponding to either occurrences of dense or large items:

- large: $m_c = m/4 - \Delta$ coordinates for each of $(1, 1)$ and $(0, 0)$ coordinate pairs (i.e., coordinates $i \in [m]$ where $(x_i, y_i) = (1, 1)$ or $(x_i, y_i) = (0, 0)$);
- large: $m/4$ coordinates for each of $(1, 0)$ and $(0, 1)$ pairs;
- dense: $\Delta - \delta$ coordinates for each of $(1, 1)$ and $(0, 0)$ pairs;
- dense: δ coordinates for each of $(1, 0)$ and $(0, 1)$ pairs.

Note that this accounts for all coordinates for a pair x, y such that $\|x - y\|_1 = m/2 + \delta$.

We now analyze the distribution of occurrences/collisions for each of large and dense items in the generated vectors (A, B), and show that, for each of large/dense items, the distribution is same as if these are occurrences of items coming from distributions a, b defined above. In particular, for, say, large items, we consider the distribution of counts $c_{i,j}$, where $i + j > 0$, where $c_{i,j}$ is the number of large items which where sampled i times on Alice's side and j times on the Bob's side; we will refer to them as (i, j) occurrence pairs. We then show that, the distribution of $(c_{i,j})_{i+j>0}$ in (A, B) is Poisson-distributed, whereas, if it were the occurrence pairs vector of samples drawn from a, b, then the distribution is a Multinomial. We then use Theorem 4.4 to conclude that the two distributions are statistically close. Note that the identify of items is not important, as the items are randomly permuted inside the domain, for both A, B as well as in distributions a, b.

Large items in (A, B)**.** We analyze the large items first. We use the fact that sum of Poisson distributions is again Poisson. For any $i, j \in \{1, \dots k\}$, the number of large (i, j) pairs is distributed as: $\mathrm{Poi}(m_c \cdot \frac{d}{\Delta} D(i, j))$ (from the first step: there are m_c coordinate pairs $(1, 1)$), plus $\mathrm{Poi}(l \cdot L(i, j) - m_c \cdot \frac{d}{\Delta} \cdot D(i, j))$ (from the third step). Thus the number of large (i, j) pairs is distributed as $\mathrm{Poi}(l \cdot L(i, j))$.

Similarly, say, considering occurrence pair $(i, 0)$ (a symmetric argument applies for $(0, i)$), the number of large $(i, 0)$ pairs is distributed as $\mathrm{Poi}(m/4 \cdot \frac{d}{\Delta} \sum_{j=1}^{k} D(i, j))$ (from the first step: there are $m/4$ coordinate pairs $(1, 0)$), plus $\mathrm{Poi}(l \cdot L(i, 0) - m/4 \cdot \frac{d}{\Delta} \sum_{j=1}^{k} D(i, j))$ (from step 4). This again amounts to $\mathrm{Poi}(l \cdot L(i, 0))$.

Large items in (a, b)**.** Let us now contrast these counts to the one would get from the "real counts" of the large items in the distribution (a, b) defined as above. The latter is a Multinomial $M^L = \mathrm{Mult}_{\text{-}0}(l; \vec{p}_L)$ where $\vec{p}_L = (L(i, j))_{i,j \geq 0; i+j>0} = (L(1, 0), L(0, 1), L(1, 1), L(2, 0), L(2, 1), ...)$. We now can use Theorem 4.4, to conclude that the the TV-distance between M^L and the distribution $\mathrm{Poi}(l \cdot \vec{p}_L)$ is bounded by: $O(\log n) \cdot \sum_{i,j \geq 0; i+j>0} L(i, j) \leq O(\log n) \cdot \sum_{i,j \geq 0; i+j>0} (t/2l)^{i+j} \leq O(1)/C \leq 0.01$ (by choosing C to be a large enough constant).

Dense items in (A, B)**.** Let's analyze the distribution of dense items now. For $i, j \geq 1$, the distribution of the number of (i, j) dense occurrence pairs is $\mathrm{Poi}((\Delta - \delta) \cdot \frac{d}{\Delta} D(i, j))$ as there are $\Delta - \delta$ coordinate pairs $(1, 1)$. Now consider the case of $(i, 0)$ occurrence pairs of dense items, for $i \in [k]$. Their count is distributed as: $\mathrm{Poi}(\delta \cdot \frac{d}{\Delta} \sum_{j=1}^{k} D(i, j))$ (from the first step: there are δ coordinate pairs $(1, 0)$), plus $\mathrm{Poi}(d \cdot D(i, 0))$ (from the second step). This amounts to:

$$\mathrm{Poi}(d \cdot \tfrac{\delta}{\Delta} \sum_{j=1}^{k} D(i, j) + d \cdot D(i, 0)).$$

Dense items in (a,b). Again, let's compare these counts to the "real counts" that would occur for the distributions (a,b). The latter distribution can be thought of as three distributions: corresponding to items in $S_A \cap S_B$, to items in $S_A \setminus S_B$, and items in $S_B \setminus S_A$. The occurrence counts for items in $S_A \cap S_B$ are distributed as a Multinomial $M^{D,i}$ with parameters $|S_A \cap S_B| = d \cdot \frac{\Delta-\delta}{\Delta}$ and probability vector $\vec{p}_D = (D(i,j))_{i,j\geq 0; i+j>0}$. By Theorem 4.4, the TV-distance between $M^{D,i}$ and the distribution $\mathrm{Poi}(d \cdot \frac{\Delta-\delta}{\Delta} \cdot \vec{p}_D)$ is bounded by: $O(\log n) \cdot \sum_{i,j\geq 0; i+j>0} D(i,j) \leq O(\log n) \cdot \sum_{i,j\geq 0; i+j>0} (t/2d)^{i+j} \leq O(1/\log n)$.

Note that the counts for $i,j \geq 1$ correspond to (1,1) pairs generating dense items in A, B above. It remains to analyze the case of $j = 0$ or $i = 0$. Wlog, consider $j = 0$ and $i > 0$ (ie, items that are only in Alice's distribution). For the distributions a, b, the occurrence pairs $(i, 0)$ are distributed as a Multinomial $M^{D,a} = \mathrm{Mult}_{\text{-}0}(|S_A \setminus S_B|; \vec{p}_{DA})$, where $\vec{p}_{DA} = (D(i))_{i\geq 1}$. By Theorem 4.4, the TV-distance between $M^{D,a}$ and $\mathrm{Poi}(d \cdot \frac{\delta}{\Delta} \cdot \vec{p}_{DA})$ is at most $O(\log n) \cdot \sum_{i\geq 1} D(i) \leq O(1/\log n)$. Summing up with the above, and focusing on $(i, 0)$ pairs, their distribution in (a, b) is at small distance to the distribution where each pair $(i, 0)$ is distributed as:

$$\mathrm{Poi}(d\cdot\tfrac{\Delta-\delta}{\Delta}\cdot D(i,0)+d\cdot\tfrac{\delta}{\Delta}\cdot D(i)) = \mathrm{Poi}(d\tfrac{\Delta-\delta}{\Delta}\cdot D(i,0)+d\tfrac{\delta}{\Delta}\cdot(D(i,0)+\sum_{j=1}^{k} D(i,j))) = \mathrm{Poi}(d\cdot\tfrac{\delta}{\Delta}\cdot\sum_{j=1}^{k} D(i,j)+d\cdot D(i,0)).$$

Thus we conclude that the distribution of occurrence pairs of the dense items in vectors A, B matches, up to a small total variation distance, the distribution for a, b as described above.

This completes the proof of the lower bound. $\square$

5 Closeness Testing: Secure Communication-Efficient Protocols

The protocol from Section 3 is clearly not secure (for example, Bob sends Alice his multi-set S, which reveals information about his input). In this section we show how to modify the protocol to make it secure, relying on standard cryptographic assumptions. Before proceeding, we provide some preliminaries, followed by our general definition for secure computation of distribution testing.

5.1 Cryptographic Tools and Preliminaries

We briefly review cryptographic tools and assumptions that we use. We keep the discussion largely informal, and focus on the aspects most relevant to our results. We refer the reader to, e.g., [Gol01, Gol04] for more details and formal definitions of standard primitives.

PRG, OT, and Our Assumptions. A pseudorandom generator (PRG) is a deterministic function G that stretches its input length, such that if the input is selected uniformly at random, the output is indistinguishable from uniform for an appropriate class of distinguishers.

In particular, the first cryptographic assumption that our secure protocols rely on, is that there exists a PRG G that can stretch $\mathrm{polylog}(m)$ bits to m bits, and fools $\mathrm{poly}(m)$-sized circuits. By default, this is what we mean when we refer to "PRG" in the rest of the paper.

A 1-out-of-m Oblivious Transfer (OT) protocol, allows one party holding input $i \in [m]$ and another party holding $s \in \{0,1\}^m$, to engage in a protocol where the first party obtains as output the bit s_i, the other party obtains no output, and no further information about s or i is leaked to the parties.

The second cryptographic assumption that our secure protocols rely on, is that there exists a 1-out-of-m OT protocol with communication complexity $\mathrm{polylog}(m)$. By default, this is what we mean when we refer to "OT" in the rest of the paper.

We note that it is easy to extend a 1-out-of-m OT protocol operating on bits (as we defined above) to one that operates on words of size r (namely, with $s \in \{\{0,1\}^r\}^m$), paying a communication overhead that is linear in r. Hence, under our OT assumption, there is such a protocol with communication complexity that is polylogarithmic in m and linear in r.

PRG and OT are both standard cryptographic primitives, that can be instantiated from various concrete number theoretic assumptions. The fact that we require super polynomial stretch for the PRG means that we need to assume subexponential hardness (and this is also the reason that we don't use the OT assumption to generically obtain PRG, as we only need OT with polynomial hardness).

We note that if we wish to weaken the assumptions, we may assume polynomial-stretch PRG (from input of size m^δ to output of size m for a constant δ) and sublinear-communication OT (1-out-of-m OT with communication complexity m^δ). Under these weaker assumptions, our secure protocols may incur higher communication complexity, but would still provide meaningful results (and remain secure). Specifically, under a weaker PRG assumption, our protocols would remain sub-linear always and would in fact keep the same complexity for part of the trade-off range. Under a weaker OT assumption, our protocols would remain sub-linear at least for some range of δ and t (the number of samples).

Secure Computation. Intuitively, secure computation allows two or more parties to evaluate some function of their inputs, such that no additional information is revealed to any party (or group of parties) beyond what follows from their own inputs and outputs.

Defining Security: Simulation Paradigm. Secure computation has been studied in many different settings. The idea underlying the security definition in all these settings, is trying to enforce that whatever the adversary can do in the real-world, can also be achieved in an ideal world, where the parties simply give their input to a trusted party, who hands them the output. Defining this formally is complex and various issues arise in different settings.

In this paper we focus on the simplest setting of a semi-honest (or "honest-but-curious") adversary, where the parties follow the protocol faithfully, but try to use their transcripts to glean more information than intended. Our defintions and protocols can be adapted to malicious adversary using standard techniques [GMW86, NN01, Gol04].

In our setting, if semi-honest Alice and Bob want to compute a function $f : \{0,1\}^* \times \{0,1\}^* \to \{0,1\}$, the definition of security boils down to requiring the existence of an efficient simulator for each party, which can simulate the view of the party (their input and transcript), from just the party's input, random input, and output.

Modular Composition. At times it is convenient to design a protocol in a modular way, where the computation of a function f may invoke a call to another function g. We will denote by Π_f^g a protocol computing f, with oracle gates to the function g. A composition theorem [Can00, Gol04] proves that if the protocol Π_f^g is a secure protocol for f, and Π_g is a secure protocol for g, then taking Π_f^g and replacing the oracle calls to g with an execution of Π_g results in a secure protocol Π_f for f (which doesn't make any oracle calls).

Feasibility Results. Starting with [Yao82], a large body of work has shown that any function that can be computed, can be computed securely in various settings. In particular, two parties holding inputs x and y can securely evaluate any circuit $C(x,y)$ with communication $O_k(|C|)$ (where k is the security parameter), under mild cryptographic assumptions. Note that this communication complexity is at least linear (since the circuit size is at least as large as the size of input and output), while we will need sublinear communication protocols. Other general techniques for secure function evaluation, where communication does *not* depend on the circuit size, include fully-homomorphic encryption (FHE) ([Gen09] and follow up work, see [Bra18] for a survey), and homomorphic secret sharing (HSS) [BGI16, BGI17]. However, these transformations still require communication that is linear in the length of input and output (prohibitive in our context). They also require stronger assumptions, and (for FHE) require a high computational overhead, or (for HSS) only apply for restricted classes of circuits. Naor and Nissim [NN01] showed a general way to transform insecure protocols to secure ones, while preserving communication. However, in general this may introduce an exponential blowup in computation, which again will not be sufficient for our needs.

While a naive application of these generic methods doesn't directly work for us, we will make use of another result by Naor and Nissim, adapting secure two party computation techniques to allow for communication-efficient protocols whenever the computation can be expressed as a (small) circuits with (large) ROM.

Secure Circuit with ROM. Consider the setting where each party has a table $R \in (\{0,1\}^r)^m$ (that is, m entries of size r each). Now consider a circuit C that, in addition to usual gates, has lookup gates which allow to access any of the parties' ROM tables (on input $i \in [m]$ the gate will return the r-bit record at the requested party's $R(i)$).

Theorem 5.1. *[NN01] If C is a circuit with ROM, then it can be securely computed with $\tilde{O}(|C| \cdot T(r,m))$ communication, where $T(r,m)$ is the communication of 1-out-of-m OT on words of size r.*

Thus, under our OT assumption, a circuit with ROM can be securely evaluated with communication complexity that is linear in $|C|$ and $|r|$, but polylogarithmic in m. We will rely on this theorem in both of our secure distribution testing constructions. Note that the main remaining challenge is to design the protocol that can be expressed in a form where this theorem can be applied.

Sampling an Orthonormal Matrix. We will use the following fact from [IW06], proven in the context of providing a secure approximation of the ℓ_2 distance.

Theorem 5.2. *[IW06] Suppose we sample a random orthonormal $n \times n$ matrix R (from a distribution defined by the Haar measure) but instead generate our randomness using a PRG G, rounding its entries to the nearest multiple of $2^{\Theta(K)}$, where $K = \Theta(k)$. Then we have for all $x \in [t]^n$:*

$$\Pr\left[\left(1-2^{-K}\right) \cdot \|x\|_2^2 \leq \|Rx\|_2 \ \ \textit{and}\ \forall_i (Rx)_i^2 < \frac{\|x\|_2^2}{n} K\right] > 1 - neg(k).$$

We note that the only place where we use a PRG in our constructions, is in order to be able to apply this theorem (in order to be able to obtain shared randomness with low communication, our parties will share a seed of a PRG that they will expand and use to sample).

5.2 Defining Secure Computation for Distribution Testing

Defining security for a distribution testing protocol requires some care, due to two new features that do not come up in the standard setting of secure computation of a function: first, this is "testing" and not "computing", and, second, the function of interest is defined with respect to distributions, but the inputs that the parties use in the computation are samples. Before providing our formal definition, we discuss these issues and our choices.

A testing problem can be described as a partial boolean function g, with the goal of computing $g(x)$ whenever g is defined on the given input x (e.g, when the input consists of two distributions that are either identical or ϵ-far). If the input x is such that $g(x)$ is not defined (we will refer to this as an *input in the gray zone*), the property testing definition (and literature) does not care about whether the output is 0 or 1. Indeed, this flexibility of having a gray zone with no correctness requirement imposed on it is precisely what allows for more efficient testing algorithms.

In contrast, for security purposes, we must care about what the protocol outputs when the input is in the gray zone, as this may reveal information about the inputs. Specifically, standard secure computation notions require that each party learns nothing beyond what follows from their own input and output.

For a protocol testing some property, when the output $g(x)$ is defined, we can (and will) follow this paradigm. When the input is in the gray zone and g is not defined, it is tempting to require that the protocol reveals this fact and no other information about the input. However, it's easy to see that such a protocol in fact is a secure computation of a complete function (where for each input x the protocol outputs whether $g(x) = 0$, $g(x) = 1$, or $g(x)$ is undefined). We could instead require that for inputs in the gray zone, the output of the protocol is some distribution over $\{0, 1\}$, independent of the input. But such a protocol again is in fact computing some complete function of the input, which intuitively defeats the point of using testing (versus computing) to gain efficiency. This intuition suggests that secure testing of any property (namely, secure computation of a partial function) cannot be achieved with better efficiency than secure computation of a (very related) complete function. We do not attempt to formalize (or refute) this intuition here.

Instead, we show that, for our more special case of secure *distribution* testing, some information must be leaked in the gray zone, *even if one disregards efficiency considerations.* Indeed, consider a closeness testing setting where Alice and Bob inputs are t samples from a and b (respectively) over $[n]$ which are either $(a, b) = (a_0, a_0)$, or $(a, b) = (a_0, b_\epsilon)$ for some distributions a_0, b_ϵ with $\|a_0 - b_\epsilon\|_1 = \epsilon$. Any correct protocols must have different outputs on such instances with, say, probability 0.99. Now define $\hat{a} = \delta \cdot a' + (1 - \delta) \cdot a$ for $\delta = 0.0001/t$, where a' is a defined on a new, unique set of letters; Clearly, the distribution of t-sample inputs from (a, b) and $(\hat{a}, \hat{b})$ are the same (for each instance), except with 0.0001 probability, and therefore the distribution over Alice and Bob views (and in particular, the output) in the latter case must be statistically close to the first one, and hence differ with at least 0.98 probability. However $(\hat{a}, \hat{b})$ in both (a_0, a_0) and (a_0, b_ϵ) instances are in the gray zone. It is therefore not possible to simulate Alice's view of Π in any statistically significant manner without additional information (besides the mere fact the distance is in the grey zone).

Following the above, we define a protocol to be a secure computation of a testing task (or partial function) $g(x)$, if it is a secure computation of some (complete, possibly randomized) function $f(x)$, where $f(x) = g(x)$ whenever $g(x)$ is defined; when $g(x)$ is not defined, $f(x)$ is some function of the input x. Thus, each party learns nothing beyond what follows from their own input and the output $f(x)$ (which means there's some leakage $f(x)$ in the gray zone, but no leakage beyond the testing output when the testing promise holds). We will require f to be a boolean function, so that the leakage in the gray zone is at most one bit of information about the input (we will also mention other possible generalizations of this definition). Finally, we note that for distribution testing, the function g is defined on the distributions, but the actual inputs of the parties are samples. This is not a major distinction for correctness, as with sufficiently many samples we can typically obtain the correct result with high probability. However, for security, the leakage f necessarily applies to the samples (our only access to the distributions), and not the distributions themselves.

We are now ready to provide our definition of secure computation of distribution testing.

Definition 5.3. *Let D be a set of input distributions over $\bigtimes_{i=1}^{d}[n_i]$, and let $g : D \to \{0, 1\}$ be a partial boolean function, defined on a subset $P \subseteq D$ (with $g(p) = \perp$ whenever $p \in D \setminus P$).*

*Let π be a d-party protocol, and let k be a security parameter. We say that π is a t-*sample secure distribution testing protocol *(for the testing task defined by g), if there exists a boolean function f : $\{\bigtimes_{i=1}^{d}[n_i]\}^t \to \{0, 1\}$ such that the following holds:*

Correctness: *for any $p \in P$,*

$$\Pr_{\zeta_1 \cdots \zeta_t \sim_{i.i.d}\, p}[f(\zeta) = g(p)] = 1 - neg(k)$$

Security: *For any $\zeta \in \{\bigtimes_{i=1}^{d}[n_i]\}^t$, if we give each player $i \in [d]$ the input $\left(1^k, \zeta_1(i), \ldots, \zeta_t(i)\right)$, then protocol π is a secure computation of the function $f(\zeta)$.*

We note that the security condition can be instantiated with any standard secure computation notion. In this paper we focus on the semi-honest model, although malicious security can be obtained by a standard transformation with low communication overhead.

More General Variants of Secure Distribution Testing. The definition can be naturally extended in various ways. We may allow non-boolean f (more bits of leakage), relax the requirement that $f = g$ whenever g is defined (allowing additional leakage even if the promise holds), or impose restrictions on f based on which leakage is deemed more or less reasonable.[2] We may also generalize the scope of distribution testing tasks modeled by the definition (e.g., we may allow one party to get more samples than the other, or give one party a complete description of a distribution rather than samples). For simplicity and ease of exposition, we do not develop these extensions here, and stick with the simpler definition above.

5.3 Secure Closeness Testing Protocol

We now describe a secure distribution testing protocol that achieves the same communication complexity as the insecure protocol from Section 3, up to poly-logarithmic factors. Specifically, we show the following theorem:

Theorem 5.4 (Closeness, Secure)**.** *Fix a security parameter $k > 1$. Fix $n > 1, \epsilon \in (0, 2)$, and let t be such that $t \geq C \cdot k \cdot \max\left(n^{2/3} \cdot \epsilon^{-4/3}, \sqrt{n} \cdot \epsilon^{-2}\right)$ for some (universal) constant $C > 0$. Then, assuming PRG and OT, there exists a* secure distribution testing protocol *for $2\mathrm{PCT}_{n,t,\epsilon}$ which uses $\tilde{O}_k\left(\frac{n^2}{t^2\epsilon^4} + 1\right)$ communication.*

The high level approach is similar to that of the previous protocol, in that we also estimate the squared ℓ_2 distance between samples drawn from split distributions a_S and b_S, for some split-distribution set S, to distinguish between $a = b$ and $\|a - b\|_1 \geq \epsilon$. One challenge in securing the protocol is that we would like Alice and Bob to agree on an alphabet (and specifically, use the same split-distribution set S) without compromising their inputs. To address this, Alice and Bob will run a secure computation over a circuit of size $\tilde{O}_{k,\epsilon}(n^2/t^2 + 1)$ which simulates samples from the split distribution instead of communicating the set S.

The main idea is to approximate the distance of the unsplitted occurrences vectors A and B (representing samples drawn from a and b), and "manually" add the difference between the unsplitted and the splitted distance for each $i \in S$. That means, the protocols will:

1. Approximate $\|A - B\|_2^2$;
2. Add exactly $\|A_S - B_S\|_2^2 - \|A - B\|_2^2$;

There is however an issue with this approach: $\|A - B\|_2^2$ might be much larger than $\|A_S - B_S\|_2^2$, and therefore the summation of (1) + (2) above might not be a good enough approximation of $\|A_S - B_S\|_2^2$. To overcome this issue, we first define the notion of "capped" vectors as follows:

Definition 5.5 (Capped Vectors)**.** *For $X \in \mathbb{R}^n$, we define $X' \in \mathbb{R}^n$ to be the 'capped vector of X with threshold L' iff $\forall i \in [n], X'_i = \min(L, X_i)$.*

Now, instead of approximating $\|A - B\|_2^2$, we will approximate "capped" versions of A, B for some carefully chosen threshold L termed A', B', and show the capped distance is a good–enough approximation to the splitted distance $\|A_S - B_S\|_2^2$. Our revised plan is therefore:

[2] Such restrictions would likely be application-dependent and subjective, as a general theory comparing which leakage functions are qualitatively "better" is an open research area in cryptography.

1. Approximate $\|A' - B'\|_2^2$;

2. Add exactly $\|A_S - B_S\|_2^2 - \|A' - B'\|_2^2$;

Securely Approximating $\|A' - B'\|_2^2$. For this task, we will use the techniques of [IW06]. Recall the $\ell 2 - Approx$ protocol from [IW06] samples an orthonormal matrix with rounded entries R as per Theorem 5.2, and samples sufficient coordinates from Rx. To ensure the protocol works correctly, efficiently and securely for all distances, it scans all possible distance magnitudes (termed T) starting with the largest possible one and dividing by 2 each time, each time with a new circuit. In our case, however, we need only to distinguish if an approximation is more than some threshold, so it suffices to run one ROM sub-circuit with T preselected for such threshold.

Choosing a set S. We also take a slightly different approach for splitting the samples. While in [DK16] and Section 3, we split the distributions using samples from b only, and show it suffices to upper bound the ℓ_2 norm for only one of the distributions we test by comparing a and b second moment approximations, here we will upper bound both a_S, b_S second moments by constructing 2 multisets: S_a from a and S_b from b and setting $S \triangleq S_a \uplus S_b$ (where $\uplus$ denotes the sum of multiplicities of each element in S_a and S_b). While not necessary for correctness, this new technique turns out to both simplify our protocol and improve its security guarantees.

We now show that $\|A' - B'\|_2^2 = \tilde{O}(\|A_S - B_S\|_2^2)$:

Lemma 5.6. *Let p be distribution on $[n]$, and let X, Y be the occurrence vectors of t_1, t_2 samples drawn from p where $t_2 \geq t_1$, then with high probability, we have that for all $i \in [n], X_i \leq 50 \ln(n) \cdot \max(1, \frac{t_1}{t_2} Y_i)$.*

Proof. By the Chernoff bound, for each $i \in [n]$ we have either:

1. $p_i \leq 1/t_1$, and then $\Pr[X_i > 10 \ln(n)] \leq 1/100n^2$; or

2. $p_i > 1/t_1$, and then $\Pr[X_i > 10 \ln(n) t_1 p_i] \leq 1/100n$ and $\Pr[Y_i < t_2 p_i/5] \leq 1/100n^2$.

so the claim follows the union bound over all coordinates. □

Corollary 5.7. *Let A, B be the occurrence vectors of t independent samples drawn from each a, b. Let A', B' be capped vectors of A, B with threshold $L \in [t]$. Let S_a, S_b be multisets of independent samples drawn from a, b of size t/L. Finally, for $S \triangleq S_a \uplus S_b$, we define A_S, B_S to be the occurrence vectors of the t samples encoded into A, B, recasted as being drawn from a_S, b_S. Then, with high probability: $\|A' - B'\|_2^2 \leq 100 \ln(n) \cdot \|A_S - B_S\|_2^2$.*

Proof. For each letter $i \in [n]$, if both A_i and B_i are larger than L, then LHS is 0. Hence, we will now assume that $B_i < L$ w.l.o.g..

By Lemma 5.6, we have with high probability for all $i \in [n]$:

1. $\{i$ multiplicity in $S_b\} \leq 50 \ln(n)$.

2. $\{i$ multiplicity in $S_a\} \leq 50 \ln(n) \max(1, A_i/L)$.

Note that on one hand, since $B'_i = B_i$ and $A'_i = \min(A_i, L)$ we have $\frac{(A'_i - B'_i)^2}{(A_i - B_i)^2} \leq \frac{(A'_i)^2}{(A_i)^2} \leq \frac{L}{A_i}$, and on the other hand, we have that $\|A_S(i, \cdot) - B_S(i, \cdot)\|_2^2 \cdot \{i$ multiplicity in $S\} \geq (A_i - B_i)^2$. By combining both inequalities we obtain that with high probability, all letters satisfy the equation $(A'_i - B'_i)^2 \leq 100 \ln(n) \cdot \|A_S(i, \cdot) - B_S(i, \cdot)\|_2^2$ and the corollary follows the summing over all letters. □

Adding $\|A_S - B_S\|_2^2 - \|A' - B'\|_2^2$. We note that simulating the splitted samples in A_S and B_S can be done independently for each letter $i \in [n]$, as $A_S(i,\cdot)$ and $B_S(i,\cdot)$ are (random) functions of the number of occurrences of the letter i in S, A and B, independently of the other letters. Furthermore, the vectors $A_S - B_S$ and $A' - B'$ differ in only $O(|S|)$, plus $O(t/L) = O(|S|)$, coordinates. This fact allows Alice and Bob to prepare simulated samples for all possible multisets in polynomial offline time, and the secure circuit can look up the correct simulation of $A_S(i,\cdot)$ and $B_S(i,\cdot)$ for each $i \in S$ and calculate $\|A_S - B_S\|_2^2 - \|A' - B'\|_2^2$ using $O(|S|)$ such lookups.

We define a method for Alice and Bob to prepare a look-up table of simulated samples from split distributions a_S and b_S for all possible multisets S of size at most t.

Definition 5.8. *Given some occurrences vector $X \in [t]^n$, we will define the 3D Split Occurrences Matrix $X^{SM} \in [t]^{n \times t \times t}$ as the following random process: for any $i \in [n], j \in [t]$, we will split X_i into j buckets $X^{SM}(i,j,1), \ldots, X^{SM}(i,j,j)$ by recasting each sample to a random bucket uniformly.*

We present our protocol next.

Secure protocol Π for the 2pCT problem
Let ζ be i.i.d. samples from the product distribution $p = a \times b$.
Alice's Input: 1^k, first coordinates of $\zeta_1, \ldots, \zeta_t$
Bob's Input: 1^k, second coordinates of $\zeta_1, \ldots, \zeta_t$

Output: 1 if $a = b$ and 0 if $\|a - b\|_1 \geq \epsilon$

1. Let $K = \Theta(k)$, and c is some constant.
2. Let $t' = t/K$; $L = \max(1, \frac{t'^3 \cdot \epsilon^4}{c \cdot n^2})$; $\alpha = \Omega(\sqrt{L/t'})$; $l = \Theta(k^2 \ln^2(n)/\alpha^2)$.
3. Alice and Bob exchange the seed of the PRG G, and generate matrix R as in Theorem 5.2.
4. Alice partitions the t samples into K Sample Sets of t' samples each, and for each Sample Set prepares ROM entries $RA', A^{SM}, M_a, X_{S_a}$ as follows:
   (a) Generate multi-set S_a using $t'/2L$ samples from the Sample Set according to Lemma 2.3.
   (b) Let X_{S_a} be the occurrence vector of S_a
   (c) Let A be the occurrence vectors of another $t'/2$ samples from the Sample Set.
   (d) Let A' be capped vector of A with threshold L.
   (e) Let $M_a = \{i : i \in S_a \vee A_i > L\}$
   (f) Let A^{SM} be 3D Split Occurrences Matrix of A.
5. Bob similarly prepares ROM entries $RB', B^{SM}, M_b, X_{S_b}$.
6. Alice and Bob run a secure circuit with ROM, C_Π to compute the following function.
   (a) For each of the sample sets $j \in [K]$:
      i. For $i \in M_a \cup M_b$
         A. Let $m_i = 1 + X_{S_a}(i) + X_{S_b}(i)$
         B. Let $y_i = \left\|A^{SM}(i, m_i, \cdot) - B^{SM}(i, m_i, \cdot)\right\|_2^2 - (A'_i - B'_i)^2$.
      ii. Compute $O_j = \sum_{i \in M_a \cup M_b} y_i$
      iii. Let $T = 2(\tau - O_j)$, where τ is the threshold from Lemma 3.2.
      iv. Generate random $i_1, ..., i_l \in [n]$ and compute $d_1 = R(A' - B')|_{i_1}^2, ..., d_l = R(A' - B')|_{i_l}^2$.
      v. Generate $z_1, ..., z_l$ from independent Bernoulli with biases $nd_1/TK, ..., nd_l/TK$.
      vi. Let $D_j = O_j + \frac{TK}{l} \sum_{i \in [l]} z_i$.
      vii. If $D_j > \tau$ vote 0; otherwise, vote 1.
   (b) Output the majority of the votes from the K tests.

5.4 Protocol analysis

We now prove Theorem 5.4 on the correctness and security of the protocol Π.

Proof of Theorem 5.4. The proof proceeds in the following steps:

1. First we define a boolean function $f(\zeta)$.
2. We show that for $p = a \times b$ such that $a = b$ or $\|a - b\|_1 \geq \epsilon$, the function $f(\zeta) = g(p)$, whenever $\zeta \sim_{i.i.d} p$ except with negligible probability.
3. Finally, we show that the protocol Π is a secure computation of $f(\zeta)$.

We define $f(\zeta)$ as follows:

$f(\zeta)$

1. Partition the samples into K Sample Sets $\zeta^1, ..., \zeta^K$ as per steps 4 and 5 of Π.
2. Compute the majority of the K test results, where a test does the following for each of the K Sample Sets ζ^i:
   (a) Generate multi-sets S_a, S_b as in the protocol.
   (b) Let $S = S_a \uplus S_b$.
   (c) Generate A, B and calculate A', B', A_S, B_S.
   (d) Let $\Delta_1 = \|A_S - B_S\|_2^2 - \|A' - B'\|_2^2$.
   (e) Let $T = 2(\tau - \Delta_1)$.
   (f) Generate $z_1, ..., z_l$ from independent Bernoulli with bias $\|A' - B'\|_2^2 / TK$.
   (g) Let $\Delta_2 = \frac{TK}{l} \sum_{i=1}^{l} z_i$.
   (h) If $\Delta_1 + \Delta_2 \geq \tau$ vote 0; otherwise vote 1.

We now show that, whenever $a = b$ or $\|a - b\|_1 \geq \epsilon$, we obtain $\Pr_{\zeta_1 \ldots \zeta_t \sim_{\text{i.i.d.}} p}[f(\zeta) = g(p)] = 1 - neg(k)$, for $p = a \times b$. We show $f(\zeta^i)$ votes correctly for each Sample Set ζ^i, with high probability. First, one can see that if $\|A' - B'\|_2^2 \geq T$ then $E[z_i] \geq 1/K$ and, by the Chernoff bound for $l \geq \Omega(k^2)$, we have that $\Delta_2 \geq \tau - \Delta_1$ with $1 - neg(k)$ probability; the test vote is 0 for this Sample Set. Similarly, if $\|A' - B'\|_2^2 \leq T/4$, the test votes 1 with $1 - neg(k)$ probability

Otherwise, when $\|A' - B'\|_2^2 \in [T/4, T]$, we have that Δ_2 is a $(1 \pm \alpha/100 \ln(n))$-factor approximation of $\|A' - B'\|_2^2$ with $1 - neg(k)$ probability, by the Chernoff bound for $l = \Omega(k^2 \cdot \ln^2 n/\alpha^2)$.

We now invoke Corollary 5.7, and obtain with high probability that:

$$\begin{aligned}\left|\Delta_1 + \Delta_2 - \|A_S - B_S\|_2^2\right| &= \left|\Delta_2 - \left\|A' - B'\right\|_2^2\right| \\ &\leq \alpha/100 \ln(n) \cdot \left\|A' - B'\right\|_2^2 \\ &\leq \alpha \cdot \|A_S - B_S\|_2^2 .\end{aligned}$$

It is left to show we have sufficient samples for the vote to be correct with high probability. For that we use Lemma 2.3 to bound $\|a_S\|_2$ and $\|b_S\|_2$ as follows:

1. We note that $|S_a| = |S_b| = t'/2L$. Therefore, S_a and S_b contain subsets $S'_a \subseteq S_a$ and $S'_b \subseteq S_b$ of size Poi$(t'/4L)$ each, with probability at least $1 - 2\frac{(t'/4L)^2}{t'/4L} = 1 - 2t'/4L = 1 - o(1)$.
2. Therefore, $\mathbb{E}\left[\left\|a_{S'_a}\right\|_2\right], \mathbb{E}\left[\left\|b_{S'_b}\right\|_2\right] \leq \sqrt{4L/t'}$ and we have both norms $O(\sqrt{L/t'})$ with probability 0.99, by the Markov bound.

3. Since $S'_a, S'_b \subseteq S$, then $\|a_S\|_2, \|b_S\|_2 = O(\sqrt{L/t'})$.

4. Furthermore, as $t' = \Omega(\sqrt{n}/\epsilon^2)$, we have $|S| = 2t'/L = O(n)$ by our choice of L.

From now on we assume all the above. According to Lemma 3.2 and Fact 2.2, all we need to make the test successful with high probability is $O(\sqrt{L/t'} \cdot n/\epsilon^2) = O(t'/c + n/\sqrt{t'}\epsilon^2) = O(t'/c + n^{2/3}/\epsilon^{4/3})$ many samples (since $t' \geq O(n^{2/3}\epsilon^{-4/3})$. By choosing c to be high enough constant, $t'/2$ samples suffice to make the vote correct with high probability. Therefore majority vote over K sample sets amplify such probability to $1 - neg(k)$.

We'll now show Π is a secure computation for $f(\zeta)$ for any ζ. For correctness, we'll show that for all ζ, $\mathbb{E}[\Pi(\zeta)] - \mathbb{E}[f(\zeta)] = neg(k)$. This is sufficient, since both Π and f provide an output in $\{0,1\}$ (so the output expectation is the same as the probability the output equals 1). We'll first show that for all sample sets $\zeta^1, \ldots, \zeta^K : \Delta_1 \equiv O_j$.

Lemma 5.9. *Let A_S, B_S denote the occurrences vector of simulated samples from a_S, b_S respectively, then we have for all sample sets $\|A' - B'\|_2^2 + O_j \equiv \|A_S - B_S\|_2^2$*

Proof. For any $i \in [n]$, let $m_i = 1 + \{i \text{ multiplicity in } S\}$. We have either:

(i) $i \notin M_a \cup M_b$, and then:

  (a) $(A'_i - B'_i)^2 = (A_i - B_i)^2$ (since $A_i, B_i \leq L$).

  (b) $\|A_S(i,\cdot) - B_S(i,\cdot)\|_2^2 = (A_i - B_i)^2$ (since $i \notin S$).

  (c) i does not contribute to O_j.

(ii) $i \in M_a \cup M_b$, and then:

  (a) i contribution to O_j is $y_i = \left\|A^{SM}(i, m_i, \cdot) - B^{SM}(i, m_i, \cdot)\right\|_2^2 - (A'_i - B'_i)^2$, and

$$\left\|A^{SM}(i, m_i, \cdot) - B^{SM}(i, m_i, \cdot)\right\|_2^2 \equiv \|A_S(i,\cdot) - B_S(i,\cdot)\|_2^2.$$

Therefore, each i contribution to LHS and RHS are equivalent. This concludes the proof. $\square$

So we are left to show that $\Pr[D_j \geq \tau] - \Pr[\Delta_1 + \Delta_2 \geq \tau] = neg(k)$. If we have $\|A' - B'\|_2 > T$, then both events occurring with prob. $1 - neg(k)$. Otherwise, we use Theorem 5.2 and obtain that:

1. With prob. $1 - neg(k)$, $\forall i (RA'_i - RB'_i)^2 \leq TK/n$.

2. $(1 - 2^{-\Theta(K)}) \|A' - B'\|_2^2 \leq E_i[n(RA'_i - RB'_i)^2] \leq \|A' - B'\|_2^2$.

Therefore, by linearity of expectations and the union bound we obtain that the distribution over the z_i differ by at most $neg(k)$ between $f(\zeta)$ and $\Pi(\zeta)$, and therefore the probability to output 0 differ by at most $neg(k)$.

Next, we show that the protocol Π is a secure computation of f. We note that the only communication in the protocol is sending is the random seed, followed by the secure circuit computation of the final output, which is indistinguishable from $f(\zeta)$. Moreover, as described above, either $\Pr[f(\zeta) = \Pi(\zeta)] \geq 1 - neg(k)$ or the distribution over the z_i (for a given seed) differ by at most $neg(k)$. Thus, replacing the secure circuit C_Π with an oracle computing the output, the simulator (for either Alice or for Bob) outputs its input, a random seed for G, and the final output, to generate a distribution that is statistically close to that in the real-world.

Finally, we analyze the communication complexity of the protocol Π for 2pCT. To analyze the circuit size of C_Π, one can observe that:

- Computing O_j can be done with a circuit of size $\tilde{O}(|M_a| + |M_b|) = \tilde{O}(t'/L) = \tilde{O}(\frac{n^2}{\epsilon^4 \cdot t^2} + 1)$ (by computing each y_i in step 6(a)i using $\tilde{O}(1)$ computations).
- Next, computing D_j (with O_j as input) can be done with a circuit of size $\tilde{O}(l) = \tilde{O}_k(\log^2 n \cdot 1/\alpha^2) = \tilde{O}_k(t'/L) = \tilde{O}_k(\frac{n^2}{t^2\epsilon^4} + 1)$.
- Computing majority over K (sub-)circuits adds multiplicative factor of $\Theta(k)$ to the circuit size.

Thus C_Π is of size $\tilde{O}_k\left(\frac{n^2}{t^2\epsilon^4} + 1\right)$, and the ROM consists of $s \triangleq \mathrm{poly}(n, t, 1/\epsilon, k)$ words of size $r \triangleq \tilde{O}(1)$ each. Therefore, the communication complexity of the secure computation of C_Π is $|C_\Pi| \cdot T(r, s)$, where $T(r, s)$ is is the communication complexity of 1-out-of-s OT on words of size r. Finally, the total communication of the protocol Π additionally also includes the length of the seed for G.

Plugging in our OT and PRG assumptions, we obtain the required communication bound (using Theorem 5.1). □

6 Independence Testing: Communication and Security

In this section, we present our protocol for independence testing. To streamline the presentation, we give directly a secure protocol, noting that obtaining communication-efficiency is the main challenge for this problem. Overall, we prove the following theorem:

Theorem 6.1 (Independence, Secure)**.** *Fix a security parameter $k > 1$. Fix $\epsilon \in (0, 2)$, $1 \le m \le n$, and let t be such that $t \ge C \cdot k \cdot \left(n^{2/3} m^{1/3} \epsilon^{-4/3} + \sqrt{nm}/\epsilon^2\right)$, for some (universal) constant $C > 0$, and assuming OT cryptographic assumption, there is a secure distribution testing protocol for $2\mathrm{pIT}_{n,m,t,\epsilon}$ using $\tilde{O}_k\left(\frac{n^2 \cdot m}{t^2\epsilon^4} + \frac{n \cdot m}{t\epsilon^4} + \frac{\sqrt{m}}{\epsilon^3}\right)$ bits of communication.*

Note that similarly to 2pCT, our protocol for 2pIT obtains better communication than via the straight-forward approach of Alice sending its input to Bob, and Bob invoking standard independence testing algorithm. We show a matching lower bound on communication for $t \to \infty$ in Section 7.

We start by providing some intuition behind our protocol, setting up some definitions along the way. As before, our protocol for 2pIT uses the framework of splitting the distributions to reduce the problem to testing an ℓ_2 distance. While it may seem tempting to reuse the sketching technique to estimate the ℓ_2 distance between the joint distribution and the product of its marginals, this technique does not help here as this distance cannot be approximated without first combining the samples from Alice and Bob. To overcome this inherent obstacle, we use an *alphabet reduction* technique. At a high level, our reduction samples a random rectangle R of the underlying split distribution alphabet, and tests its distance to the marginal product distribution. This concept helps us to improve communication in two ways. First, we obtain better communication as we need to deal with a smaller alphabet problem, and, second, we can use the additional samples for tighter ℓ_2 bound on the smaller alphabet product distribution. Each of these two improvements achieve linear improvement with more samples and therefore we hope for a quadratic improvement overall. The latter improvement is only effective while $t \ll n$ samples, and hence the quadratic improvement kicks in only in that regime.

To formalize the alphabet-reduction idea, we start by making the following definition:

Definition 6.2. *Fix some distribution p over alphabet $[n]$. For $\mathcal{U} \subseteq [n]$ let $p_{|\mathcal{U}}$ be the distribution p conditioned on the set $\mathcal{U}$: i.e., $p_{|\mathcal{U}}$ is a distribution over alphabet $\mathcal{U}$ such that for any $i \in \mathcal{U}$, $p_{|\mathcal{U}}(j) = p(j)/\sum_{i\in\mathcal{U}} p(i)$.*

Given joint distribution $p = (a, b)$, let S_a, S_b be the multisets in $[n], [m]$ of samples from a, b respectively. Define the multiset $S \subset [n]\times[m]$ to be such that $1+\{\text{multiplicity of } (i,j) \text{ in } S\} = (1+\{\text{multiplicity of } i \text{ in } S_a\}) \cdot (1 + \{\text{multiplicity of } j \text{ in } S_b\})$.

Our idea is to sample a rectangle $R = R_a \times R_b \subseteq [n + |S_a|] \times [m + |S_b|]$, and test the closeness of the sub-distributions $\hat{p} \triangleq (p_S)_{|R}$ and $\hat{q} \triangleq (a_{S_a})_{|R_a} \times (b_{S_b})_{|R_b}$. We would like R to satisfy the following properties:

1. If a is independent of b, denoted $a \perp\!\!\!\perp b$, then $\hat{p} = \hat{q}$, and if a and b are ϵ-far from being independent, then $\hat{p}$ is at distance $\Omega(\epsilon)$ from $\hat{q}$.
2. $\Pr_{i\sim p}[i \in R]$ is high enough, so that we have sufficient samples to test $\hat{p}$ and $\hat{q}$ for closeness.
3. R is sufficiently small to allow small communication.
4. The secure circuit can simulate samples from $\hat{p}, \hat{q}$ using $\tilde{O}(1)$ ROM look-ups per sample.

We obtain conditions (1) and (2) above by bounding the ℓ_2 norm of the underlying product distribution through splitting, and use such bound also as a bound over the variance of the ℓ_1 mass of $\hat{p}$ and its distance from $\hat{q}$. We analyze such approach in Section 6.2. For condition (4), we only sample $R_a \subseteq [n + |S_a|]$ (assuming w.l.o.g. $n \geq m$), and have R_b be the complete alphabet of Bob $[m + |S_b|]$. The final communication/sample trade-offs are obtained from the best trade-off between conditions (2) and (3).

In order to be able to look-up and "pair" Alice and Bob samples from some rectangle R, we need a method for storing a table of all indices for each letter. For this, we use the following definition:

Definition 6.3. *Given t indexed samples $X \in [n]^t$, we define* the Indices Set Vector $\mathcal{I} \in (2^{[t]})^n$ *such that $\mathcal{I}(j) = \{i \in [t] : X_i = j\}$.*

6.1 Full protocol description

In contrast to closeness testing, the main challenge for the 2pIT problem is designing a communication-efficient protocol, while adding security introduces relatively minor nuances. Hence, for ease of exposition, we present the secure version of our protocol directly.

Our overall protocol proceeds as follows at a high level. Alice and Bob first generate multi-sets S_a, S_b. Alice then simulates one set of indexed samples from a_{S_a} (termed A) while Bob simulates *two* sets of samples from b_{S_b}. First set (termed B_p) is obtained using the sample set corresponding to the set A, and the second one using a fresh independent sample set (termed B_q). Alice then prepares Indices Set Vector $\mathcal{I}$ of A. Next, a secure circuit with ROM samples a uniform subset $\mathcal{U}$ from $[n + |S_a|]$, looks up all of Alice's samples coming from $\mathcal{U}$, and generates 2 joint sample sets: the first one by pairing each of Alice's samples A with the corresponding sample in B_p (thereby simulating samples from $\hat{p}$ as defined above), and then by pairing each of Alice's samples A with an independent sample from B_q (thereby simulating samples from $\hat{q}$). Finally, those two sets are then being tested (directly) for closeness using Lemma 3.2.

We note that the set $\mathcal{U}$, together with the protocol's output can possibly compromise Bob's data, and therefore we need the secure circuit to compute $\mathcal{U}$ (rather than Alice providing it to the circuit). One challenge with such approach though, is the set $\mathcal{U}$ may be larger than the communication bound. Luckily, we only care about non-empty letters in $\mathcal{U}$ which are (in expectation) of small size, and since we sample $\mathcal{U}$ uniformly from $[n + |S_a|]$, we can overcome the issue by sampling the non-empty letters of $\mathcal{U}$ from the non-empty letters of $[n + |S_a|]$ (e.g. appears at least once in A). For this we need to define a distribution over the size of intersection of subsets chosen uniformly:

Definition 6.4. *For $n \geq 1$, and $\alpha, \beta \leq n$, we define a discrete distribution called Uniform Subset Intersection distribution $\mu(n, \alpha, \beta)$ as follows: Fix S to be a set of size n, and $S_1 \subseteq S$ of size α. Then $\mu(n, \alpha, \beta) \stackrel{\Delta}{\equiv} |S_1 \cap S_2|$, where $S_2 \subseteq S$ of size β is chosen uniformly at random.*

Claim 6.5. *Let S be a finite set of size $n \geq 1$, and let $S_1 \subseteq S$ of size α. Let S_2 be a uniform subset of S of size β. Finally, let S_3 be uniform subset of S_1 of size $\mu(n, \alpha, \beta)$. Then $S_3 \equiv S_1 \cap S_2$.*

Proof. One observe that $|S_3| = \mu(n, \alpha, \beta) \equiv |S_1 \cap S_2|$ by our definition of μ. Furthermore, since S_2 was chosen uniformly at random, then each element of S_1 has equal and independent probability to be part of $S_1 \cap S_2$, and therefore $S_1 \cap S_2$ is also a uniform subset of S_1. $\square$

We now present our protocol for 2pIT. As before, our protocol uses a secure circuit with ROM which will sample from Alice and Bob's input strings.

Secure protocol Π for the 2pIT problem
Let $\zeta_1, \ldots \zeta_t$ be i.i.d. samples from the joint distribution $p = (a, b)$.
Alice's Input: 1^k, first coordinates of $\zeta_1, \ldots, \zeta_t$
Bob's Input: 1^k, second coordinates of $\zeta_1, \ldots, \zeta_t$

Output: 1 if $a \perp\!\!\!\perp b$ and 0 if p is ϵ-far from any product distribution.

1. Let $K = \Theta(k); \epsilon' = \Omega(\epsilon); t' = \min\{t/3K, O(n \cdot \sqrt{m}/\epsilon)\}; l = O(\max\{\frac{n^3 \cdot m}{t'^3 \epsilon^4}, \frac{n^2 \cdot m}{t'^2 \epsilon^4}, \frac{1}{\epsilon^2}\})$.
2. Alice and Bob partition their samples into $3K$ Sample Sets of t' samples each with corresponding indices termed $A_1, \ldots, A_{3K}, B_1, \ldots, B_{3K}$.
3. For $i \in \{0, \ldots, K-1\}$ Alice prepares ROM entries termed $A^i, \mathcal{I}^i, \Gamma^i$ as follows:
   (a) Generate multi-set S_a^i using $\min\{t', n\}$ samples from A_{3i+1} according to Lemma 2.3.
   (b) Let A^i be A_{3i+2} recasted as being drawn from $a_{S_a^i}$.
   (c) Let $\mathcal{I}^i$ be the Indices Set Vector of A^i.
   (d) Let $\Gamma^i = \{$non-empty letters of $\mathcal{I}^i\}$.
4. For $i \in \{0, \ldots, K-1\}$ Bob prepares ROM entries termed B_p^i, B_q^i as follows:
   (a) Generate multi-set S_b^i using m samples from B_{3i+1} according to Lemma 2.3.
   (b) Let B_p^i be B_{3i+2} recasted as being drawn from $b_{S_b^i}$.
   (c) Let B_q^i be B_{3i+3} recasted as being drawn from $b_{S_b^i}$.
5. A secure circuit with ROM$(A^i, B_p^i, B_q^i, \mathcal{I}^i, \Gamma^i)$ computes for each $i \in \{0, \ldots, K-1\}$ the following vote, and then outputs the majority vote:
   (a) Compute $\lambda = \min\{\mu(n + |S_a^i|, |\Gamma^i|, l), 100t'l/n\}$
   (b) Sample $\mathcal{U}'$ uniformly from Γ^i of size λ.
   (c) Let $I^i = \bigcup_{j \in \mathcal{U}'} \mathcal{I}^i(j)$.
   (d) Compute $I_p^i, I_q^i, J_p^i, J_q^i$ as uniform disjoint subsets of I^i of size $\min\{O(t'l/n), \lfloor |I^i|/4 \rfloor\}$ each.
   (e) Compute $X_1^i = \{(A^i(j), B_p^i(j)) : j \in I_p^i\}$.
   (f) Compute $X_2^i = \{(A^i(j), B_p^i(j)) : j \in J_p^i\}$.
   (g) Compute $Y_1^i = \{(A^i(j), B_q^i(j)) : j \in I_q^i\}$.
   (h) Compute $Y_2^i = \{(A^i(j), B_q^i(j)) : j \in J_q^i\}$.
   (i) Count collisions in X_1^i, Y_1^i and produce estimations of $\|\hat{p}\|_2, \|\hat{q}\|_2$ up to factor 2. Compute $\chi^i = 1$ if they these estimations agree up to factor 4 and $\chi^i = 0$ otherwise.
   (j) Compute $\Delta^i = \|X - Y\|_2^2$ where X and Y are the occurrence vectors of X_2^i and Y_2^i.
   (k) Vote $\chi^i \wedge (\Delta^i \leq \tau)$, where τ is threshold from Lemma 3.2

Remark 6.6. *We note that one can obtain a non-secure, 1-round 1-way communication protocol by having Alice computing the set $\mathcal{U}$ of size l (sampled from $[n + |S_a^i|]$), and sending to Bob sufficiently many* *samples coming from $\mathcal{U}$, along with corresponding indices of such samples, while Bob performing the pairing and testing for closeness as per steps $5.(e), \ldots, 5.(k)$ above.*

6.2 Analysis setup: alphabet reduction

Before proceeding to the full protocol analysis, we first argue some auxiliary lemmas, regarding reducing the larger independence testing problem to a smaller closeness problem which requires less samples and therefore smaller communication than the original problem. The lemmas state show that sampling sub-distributions from distributions with bounded ℓ_2 norms preserve some important properties.

We first show that if we sample sufficiently large sub-distribution $p_{|\mathcal{U}}$ uniformly from some distribution p with bounded $\|p\|_2^2$, then with high probability, $p_{|\mathcal{U}}$ has density of the same magnitude as the fraction of the alphabet we are sampling from, and $\left\|p_{|\mathcal{U}}\right\|_2^2$ is similarly bounded.

Lemma 6.7. *Let p be some distribution over $[n]$ such that $\|p\|_2^2 \leq U$, and let $\mathcal{U}$ of size l be a uniformly random subset of $[n]$. For $l \geq 100 \cdot U \cdot n$ we have with 0.95 probability:*

1. $\sum_{i \in \mathcal{U}} p_i = \Theta(l/n)$;
2. $\left\|p_{|\mathcal{U}}\right\|_2^2 = O(U \cdot n/l)$.

Proof. Note that the expectation of (1) is l/n and its variance is at most $U \cdot l/n$ (roughly). Therefore, by the Chebyshev inequality and the lower bound on l, we have (1) with probability ≥ 0.99.
For (2), we note that $\left\|p_{|U}\right\|_2^2 = \sum_{i \in \mathcal{U}} p_i^2 / (\sum_{i \in \mathcal{U}} p_i)^2$. The numerator is $O(U \cdot l/n)$ in expectation, and therefore for high enough constant $O(U \cdot l/n)$ with 0.99 probability. The denominator is $\Omega(l^2/n^2)$ by the above argument. ☐

We now derive by a similar argument that such sampling process also preserves distances.

Lemma 6.8. *Let p be some distribution over $[n]$ such that $\|p\|_2^2 \leq U$, and let $\Delta \in [0,2]^n$. Let $\mathcal{U}$ of size l be a uniformly random subset of $[n]$. If $\langle p, \Delta \rangle \geq \epsilon$ and $l \geq O(U \cdot n/\epsilon^2)$, then we have with 0.9 probability that:*

$$\frac{\sum_{i \in \mathcal{U}} p_i \cdot \Delta_i}{\sum_{i \in \mathcal{U}} p_i} = \Omega(\epsilon).$$

Proof. By Lemma 6.7, the denominator is $\Theta(l/n)$ with 0.95 probability. The expectation of the numerator is $l\epsilon/n$ and its variance is at most $U \cdot l/n$. Therefore, for $l \geq O(U \cdot n/\epsilon^2)$ (using high enough constant), the numerator is $\Theta(l\epsilon/n)$ with probability 0.95 by the Chebyshev inequality. ☐

Finally, we show that we can apply the above lemmas to obtain a reduction from testing a distribution $p = [n] \times [m]$, to testing closeness of a smaller sub-distribution (where we down-sample the letters from $[n]$ and condition on those) to the product distribution.

Lemma 6.9. *Let p be a distribution on $[n] \times [m]$ and let p_1, p_2 be its marginals. Fix $\epsilon < 2$. Let $\mathcal{U} \subset [n]$ be a random subset of size l, such that $l \geq O(U \cdot n/\epsilon^2)$ where $U = \|p_1\|_2^2$. Define $\hat{p} \triangleq p_{|\mathcal{U} \times [m]}$ and $\hat{q} \triangleq p_{1|\mathcal{U}} \times p_2$. There exists $\epsilon' = \Omega(\epsilon)$ such that with 0.9 probability,*

1. *If p is a product distribution, then $\hat{p} = \hat{q}$.*
2. *If p is ϵ-far from any product distribution, then $\|\hat{p} - \hat{q}\|_1 \geq \epsilon'$.*

Proof. We obtain (1) immediately from the fact $p_{1|\mathcal{U}}$ and p_2 are the marginals of $\hat{p}$.

For $i \in [n]$, let $\Delta_i = \|p(i,*) - p_2\|_1$, where $p(i,*)$ is the right-marginal of p conditioned on $p_1 = i$. Since $p(i,j) = p(i,*)(j) \cdot p_1(i)$, we have that $\langle p_1, \Delta \rangle = \|p - p_1 \times p_2\|_1$. If p is ϵ-far from the product distribution, then $\langle p_1, \Delta \rangle \geq \epsilon$. We now invoke Lemma 6.8 using $p = p_1$, and get that with 0.9 probability:

$$\left\|p_{|\mathcal{U}\times[m]} - p_{1|\mathcal{U}} \times p_2\right\|_1 = \frac{\sum_{i\in\mathcal{U}} p_1(i) \cdot \Delta_i}{\sum_{i\in\mathcal{U}} p_1(i)} = \Omega(\epsilon).$$

□

6.3 Analysis of the protocol

Proof of Theorem 6.1. The proof proceeds in the following steps:

1. First we define a boolean function $f(\zeta)$.
2. We show that for $p = (a,b)$ such that $a \perp\!\!\!\perp b$ or p is ϵ-far from any product distribution, the function $f(\zeta) = g(p)$, whenever $\zeta \sim_{i.i.d} p$ except with negligible probability.
3. Finally, we show that the protocol Π is a secure computation of $f(\zeta)$.

For $f(\zeta)$, we define f to be the boolean function producing a bit by simulating steps $(1), \ldots, (5)$ of $\Pi(\zeta)$ by Alice and Bob. Indeed, such simplification is possible for 2PIT since the protocol involves no communication outside of the secure circuit computation.

We now show that, whenever $a \perp\!\!\!\perp b$ or p is ϵ-far from any product distribution, we obtain $\Pr_{\zeta_1 \ldots \zeta_t \sim_{\text{i.i.d.}} p}[f(\zeta) = g(p)] = 1 - neg(k)$. We prove for each $i \in \{0, \ldots, K-1\}$, all the below happen with some high constant probability:

1. X_1^i, X_2^i are distributed as being drawn from $\hat{p}$ and Y_1^i, Y_2^i are distributed as being drawn from $\hat{q}$.
2. $\hat{p} = \hat{q}$ if $a \perp\!\!\!\perp b$ and $\|\hat{p} - \hat{q}\|_1 \geq \epsilon'$ if (a,b) are ϵ-far from the product distribution.
3. X_1^i, Y_1^i contain sufficiently many samples to approximate $\|\hat{p}\|_2, \|\hat{q}\|_2$, and X_2^i and Y_2^i contain sufficiently many samples to test the closeness of $\hat{p}$ and $\hat{q}$ according to Lemma 3.2.

For (1), we note that $|\Gamma^i| \leq t'$ and therefore $\mathbb{E}[\mu(n+|S_a|, |\Gamma^i|, l)] \leq t' \cdot l/n$ and we have $\lambda = \mu(n+|S_a|, |\Gamma^i|, l)$ with 0.99 probability (by Markov). According to Claim 6.5, the process in which the circuit samples $\mathcal{U}'$ from Γ^i simulates the process of sampling $\mathcal{U}$ of size l uniformly from $[n + |S_a|]$ with $\mathcal{U}' \equiv \Gamma^i \cap \mathcal{U}$, and therefore X^i and Y^i are distributed as being drawn from $\hat{p}$ and $\hat{q}$.

We obtain (2) with 0.9 probability immediately from Lemma 6.9.

For (3), let us denote $t_{\mathbf{MIN}}$ to be the samples required both to meet the conditions of Lemma 3.2, and to produce a constant factor approximation for $\|\hat{p}\|_2$ and $\|\hat{q}\|_2$. Let us also denote $t_{\mathbf{REAL}}$ to be the actual samples our protocol produces from $\hat{p}, \hat{q}$ in steps $5.(d), \ldots, 5.(h)$ above. Let $\gamma = \min\{t', n\}$. With $1 - o(1)$ probability, S_a, S_b contain subsets $S'_a \subseteq S_a$ and $S'_b \subseteq S_b$ of sizes $\text{Poi}(\gamma/2)$ and $\text{Poi}(m/2)$ respectively. Therefore, according to Lemma 2.3 and Lemma 6.7:

$$\|\hat{q}\|_2^2 = \left\|a_{S_a|\mathcal{U}}\right\|_2^2 \cdot \|b_{S_b}\|_2^2 = O(\frac{n}{\gamma l}) \cdot O(\frac{1}{m}) = O(\frac{n}{\gamma l m})$$

And therefore:

$$t_{\mathbf{MIN}} = O\left(lm\sqrt{\frac{n}{\gamma lm}} \cdot \epsilon^{-2} + \sqrt{lm}\right) = O\left(\sqrt{\frac{nlm}{\gamma \epsilon^4}}\right)$$

On the other hand we have by Lemma 6.7 with probability 0.95:

$$t_{\mathbf{REAL}} = \Omega\left(\frac{t'l}{n}\right)$$

As a result, there exists some constant c for which we obtain $t_{\mathbf{REAL}} \geq t_{\mathbf{MIN}}$ with probability $0.95 - o(1)$ whenever $l \geq \frac{cn^3m}{t'^2\gamma\epsilon^4} = c \cdot \max\{\frac{n^3 \cdot m}{t'^3\epsilon^4}, \frac{n^2 \cdot m}{t'^2\epsilon^4}\}$.

We note as well that Δ^i is an exact computation of $\|X - Y\|_2^2$ and therefore can also be considered $(1+\alpha)$ approximation of such quantity (for any α).

Last, we need to show l is properly defined, and in particular, $l \leq n + |S_a|$. Note that from the conditions of Theorem 6.1, $t' \geq C \cdot O(\sqrt{nm}/\epsilon^2)$. We thus set C to be a large enough constant such that $l \leq n$ holds.

To sum up, we have shown (by the union bound) that for each $i \in \{0, \ldots, K-1\}$, the conditions of Lemma 3.2 are met with probability $0.85 - o(1)$, and therefore $f(\zeta)$ votes correctly with probability $1/2 + \Omega(1)$. The final majority output over K votes is therefore correct with $1 - neg(k)$ probability as needed.

We'll now show Π is a secure computation for $f(\zeta)$ for any ζ. For correctness, we need show that for all ζ, $\mathbb{E}[\Pi(\zeta)] - \mathbb{E}[f(\zeta)] = neg(k)$. In fact, those are equal by our definition of f above.

Security follows immediately from the security of the secure circuit with ROM, as there is no additional communication or randomness in the protocol.

We now analyze the communication complexity of Π. We note all communication is invoked from the secure circuit in step (5) above. For each of the K (sub-)circuits producing a vote, the circuit first makes $O(1)$ look-ups for each letter in $\mathcal{U}'$. The circuit then calculates the number of corresponding sample indices in $\mathcal{I}^i$ and samples $O(t'l/n)$ such indices. One can observe that such sampling process is possible using a circuit of size $O(t'l/n)$ even with the (low-probability) scenario where there are $\omega(t'l/n)$ such indices since the index sets are disjoint. For each sampled index (in $I_p^i, I_q^i, J_p^i, J_q^i$), the circuit invokes $\tilde{O}(1)$ calculations to produce collision count and occurrence vector squared distance. Therefore, the circuit size is bounded by $\tilde{O}(\lambda + t'l/n) = \tilde{O}(t'l/n) = \tilde{O}\left(\frac{n^2 \cdot m}{t'^2\epsilon^4} + \frac{n \cdot m}{t'\epsilon^4} + \frac{\sqrt{m}}{\epsilon^3}\right)$, and the overall circuit producing majority vote is just K times that:

$$\tilde{O}_k\left(\frac{n^2 \cdot m}{t^2\epsilon^4} + \frac{n \cdot m}{t\epsilon^4} + \frac{\sqrt{m}}{\epsilon^3}\right)$$

By assuming OT, and invoking Theorem 5.1, we obtain the claimed complexity. □

7 Independence Testing: Communication Lower Bound

We now show our lower bounds for 2PIT. We show that any one-way protocol requires $\Omega(\sqrt{m})$ bits of communication (regardless of the security properties).

The lower bound result is obtained by reduction from the Boolean Hidden Hypermatching (BHH) problem from [VY11]. In the BHH problem, Alice is given $x \in \{0,1\}^n$, and Bob is given some complete matching M of $[n]$ such that for all pairs (i,j) in the matching: $i \oplus j = b \in \{0,1\}$. Bob's goal to output b correctly with $2/3$ probability. For this problem, [VY11] show that one-way communication complexity is $\Omega(\sqrt{n})$.

Using the above result, we now show the following:

Theorem 7.1. *For $n, t \in \mathbf{N}$, any one-way protocol for $2\text{PIT}_{n,n,t,1,1/3}$ requires $\Omega(\sqrt{n})$ bits of communication.*

Proof of Theorem 7.1. We show how, for given a BHH instance $p^b = (x, M)$, we can reduce it to the $2\text{PIT}_{n,n,t,1}$ problem. Let $r \in [n]^t$ be some common uniformly random string. In addition, for $b \in \{0, 1\}$, let $X_b = \{i \in [n] : x_i = b\}$. For each $i \in [t]$, Alice, Bob each will generate an indexed sample (A_i, B_i) respectively as follows:

1. Let $b' = x_{r(i)}$; Alice samples A_i uniformly from $X_{b'}$.
2. Bob samples B_i uniformly between $r(i)$ and its pair in the matching M.

One can observe that for any $i \in [t], j \in [n]$, $\Pr[A_i = j] = \Pr[B_i = j] = 1/n$, i.e., each of A, B is distributed uniformly at random from $[n]$. Thus, if the distributions of A and B are to be independent, it must be a distribution uniform over $[n] \times [n]$.

Now let us analyze the probability distribution of each pair (A_i, B_i): for any $i \in [t], j \in [n], k \in [n]$, we have $(A_i, B_i) = (j, k)$ iff the following precise conditions hold:

1. $x_{r(i)} = x_j$.
2. $r(i)$ is either k or its pair in the matching M.
3. j was sampled by Alice.
4. k was sampled by Bob.

Events (3) and (4) are independent events which happen with probability $2/n$ and $1/2$ respectively conditioned on the first 2 events occurring.

For a p^1 instance (ie, when the output $b = 1$), the probability of the first 2 events occurring is $1/n$ for any (j, k) (as there is a single letter in $[n]$ that supports both conditions), and therefore for any $i \in [t], j \in [n], k \in [n]$, we have $\Pr[(A_i, B_i) = (j, k)] = 1/n^2$. Hence (A, B) are distributed as a product distribution.

On the other hand, for a p^0 instance, the probability of the first 2 events occurring is $2/n$ if $x_j = x_k$ and 0 otherwise. Thus for any $i \in [t], j \in [n], k \in [n]$, we have that $\Pr[(A_i, B_i) = (j, k)]$ is either 0 or $2/n^2$. This means that we have an instance which is at distance 1 from the product distribution $[n] \times [n]$ (and hence $\Omega(1)$ from any distribution). $\square$

Acknowledgements

We thank Devanshi Nishit Vyas for her contribution to some of the initial work which led to this paper. We thank Clement Canonne for invaluable comments on an early draft of the manuscript. We thank Yuval Ishai for helpful discussions.

References

[A+98] S Amari et al. Statistical inference under multiterminal data compression. *IEEE Transactions on Information Theory*, 44(6):2300–2324, 1998.

[AAK+07] Noga Alon, Alexandr Andoni, Tali Kaufman, Kevin Matulef, Ronitt Rubinfeld, and Ning Xie. Testing k-wise and almost k-wise independence. In *Proceedings of the Symposium on Theory of Computing (STOC)*, pages 496–505, 2007.

[AC86] Rudolf Ahlswede and Imre Csiszár. Hypothesis testing with communication constraints. *IEEE transactions on information theory*, 32(4), 1986.

[ACFT18] Jayadev Acharya, Clément L. Canonne, Cody Freitag, and Himanshu Tyagi. Test without trust: Optimal locally private distribution testing. *CoRR*, abs/1808.02174, 2018.

[ACT18] Jayadev Acharya, Clément L. Canonne, and Himanshu Tyagi. Distributed simulation and distributed inference. *CoRR*, abs/1804.06952, 2018.

[ADJ+11] Jayadev Acharya, Hirakendu Das, Ashkan Jafarpour, Alon Orlitsky, and Shengjun Pan. Competitive closeness testing. In *Proceedings of the 24th Annual Conference on Learning Theory*, pages 47–68, 2011.

[ADJ+12] Jayadev Acharya, Hirakendu Das, Ashkan Jafarpour, Alon Orlitsky, Shengjun Pan, and Ananda Suresh. Competitive classification and closeness testing. In *Conference on Learning Theory*, pages 22–1, 2012.

[ADK15] Jayadev Acharya, Constantinos Daskalakis, and Gautam C Kamath. Optimal testing for properties of distributions. In *Advances in Neural Information Processing Systems*, pages 3591–3599, 2015.

[ADR17] Maryam Aliakbarpour, Ilias Diakonikolas, and Ronitt Rubinfeld. Differentially private identity and closeness testing of discrete distributions. *CoRR*, abs/1707.05497, 2017.

[AJOS14] Jayadev Acharya, Ashkan Jafarpour, Alon Orlitsky, and Ananda Theertha Suresh. Sublinear algorithms for outlier detection and generalized closeness testing. In *Information Theory (ISIT), 2014 IEEE International Symposium on*, pages 3200–3204. IEEE, 2014.

[AMS99] Noga Alon, Yossi Matias, and Mario Szegedy. The space complexity of approximating the frequency moments. *J. Comp. Sys. Sci.*, 58:137–147, 1999. Previously appeared in STOC'96.

[ASZ17] Jayadev Acharya, Ziteng Sun, and Huanyu Zhang. Differentially private testing of identity and closeness of discrete distributions. *CoRR*, abs/1707.05128, 2017.

[Bar88] Andrew D Barbour. Stein's method and poisson process convergence. *Journal of Applied Probability*, 25(A):175–184, 1988.

[BBM12] Eric Blais, Joshua Brody, and Kevin Matulef. Property testing lower bounds via communication complexity. *Computational Complexity*, 21(2):311–358, 2012.

[BCG16] Eric Blais, Clément Louis Canonne, and Tom Gur. Alice and bob show distribution testing lower bounds (they don't talk to each other anymore.). In *Electronic Colloquium on Computational Complexity (ECCC)*, volume 23, page 1, 2016.

[BCNW08] Amos Beimel, Paz Carmi, Kobbi Nissim, and Enav Weinreb. Private approximation of search problems. *SIAM J. Comput.*, 38(5):1728–1760, 2008.

[BFF+01] T. Batu, L. Fortnow, E. Fischer, R. Kumar, R. Rubinfeld, and P. White. Testing random variables for independence and identity. In *Proceedings of the 42Nd IEEE Symposium on Foundations of Computer Science*, FOCS '01, pages 442–, Washington, DC, USA, 2001. IEEE Computer Society.

[BFR+00] Tugkan Batu, Lance Fortnow, Ronitt Rubinfeld, Warren D Smith, and Patrick White. Testing that distributions are close. In *Foundations of Computer Science, 2000. Proceedings. 41st Annual Symposium on*, pages 259–269. IEEE, 2000.

[BFR+13] Tuğkan Batu, Lance Fortnow, Ronitt Rubinfeld, Warren D Smith, and Patrick White. Testing closeness of discrete distributions. *Journal of the ACM (JACM)*, 60(1):4, 2013.

[BGI16] Elette Boyle, Niv Gilboa, and Yuval Ishai. Breaking the circuit size barrier for secure computation under DDH. In *CRYPTO (1)*, volume 9814 of *Lecture Notes in Computer Science*, pages 509–539. Springer, 2016.

[BGI17] Elette Boyle, Niv Gilboa, and Yuval Ishai. Group-based secure computation: Optimizing rounds, communication, and computation. In *EUROCRYPT (2)*, volume 10211 of *Lecture Notes in Computer Science*, pages 163–193, 2017.

[BGM+16] Mark Braverman, Ankit Garg, Tengyu Ma, Huy L Nguyen, and David P Woodruff. Communication lower bounds for statistical estimation problems via a distributed data processing inequality. In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pages 1011–1020. ACM, 2016.

[BHH11] Sergey Bravyi, Aram W Harrow, and Avinatan Hassidim. Quantum algorithms for testing properties of distributions. *IEEE Transactions on Information Theory*, 57(6):3971–3981, 2011.

[BHN09] Amos Beimel, Renen Hallak, and Kobbi Nissim. Private approximation of clustering and vertex cover. *Computational Complexity*, 18(3):435–494, 2009.

[BKR04] Tugkan Batu, Ravi Kumar, and Ronitt Rubinfeld. Sublinear algorithms for testing monotone and unimodal distributions. In *Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*, pages 381–390. ACM, 2004.

[BO10] Vladimir Braverman and Rafail Ostrovsky. Measuring independence of datasets. In *Proceedings of the forty-second ACM symposium on Theory of computing*, pages 271–280. ACM, 2010.

[Bra18] Zvika Brakerski. Fundamentals of fully homomorphic encryption – a survey. *Electronic Colloquium on Computational Complexity (ECCC)*, 125, 2018.

[BV15] Bhaswar Bhattacharya and Gregory Valiant. Testing closeness with unequal sized samples. In *Advances in Neural Information Processing Systems*, pages 2611–2619, 2015.

[Can00] Ran Canetti. Security and composition of multiparty cryptographic protocols. *J. Cryptology*, 13(1):143–202, 2000.

[Can15] Clément L Canonne. A survey on distribution testing: Your data is big. but is it blue? In *Electronic Colloquium on Computational Complexity (ECCC)*, volume 22, pages 1–9, 2015.

[CDK17] Bryan Cai, Constantinos Daskalakis, and Gautam Kamath. Priv'it: Private and sample efficient identity testing. *arXiv preprint arXiv:1703.10127*, 2017.

[CDVV14] Siu-On Chan, Ilias Diakonikolas, Gregory Valiant, and Paul Valiant. Optimal algorithms for testing closeness of discrete distributions. In *Proceedings of the twenty-fifth annual ACM-SIAM symposium on Discrete algorithms*, pages 1193–1203. Society for Industrial and Applied Mathematics, 2014.

[CMVW16] Michael Crouch, Andrew McGregor, Gregory Valiant, and David P Woodruff. Stochastic streams: Sample complexity vs. space complexity. In *LIPIcs-Leibniz International Proceedings in Informatics*, volume 57. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2016.

[Cov69] Thomas M Cover. Hypothesis testing with finite statistics. *The Annals of Mathematical Statistics*, 40(3):828–835, 1969.

[CR12] Amit Chakrabarti and Oded Regev. An optimal lower bound on the communication complexity of gap-hamming-distance. *SIAM Journal on Computing*, 41(5):1299–1317, 2012.

[DGL+17] I. Diakonikolas, E. Grigorescu, J. Li, A. Natarajan, K. Onak, and L. Schmidt. Communication-efficient distributed learning of discrete distributions. In *Advances in Neural Information Processing Systems*, 2017. To appear.

[DGPP16] Ilias Diakonikolas, Themis Gouleakis, John Peebles, and Eric Price. Collision-based testers are optimal for uniformity and closeness. *arXiv preprint arXiv:1611.03579*, 2016.

[DHS15] Ilias Diakonikolas, Moritz Hardt, and Ludwig Schmidt. Differentially private learning of structured discrete distributions. In *Advances in Neural Information Processing Systems*, pages 2566–2574, 2015.

[DK16] Ilias Diakonikolas and Daniel M Kane. A new approach for testing properties of discrete distributions. In *Foundations of Computer Science (FOCS), 2016 IEEE 57th Annual Symposium on*, pages 685–694. IEEE, 2016.

[DS18] Yuval Dagan and Ohad Shamir. Detecting correlations with little memory and communication. 2018.

[FIM+06] Joan Feigenbaum, Yuval Ishai, Tal Malkin, Kobbi Nissim, Martin J. Strauss, and Rebecca N. Wright. Secure multiparty computation of approximations. *ACM Trans. Algorithms*, 2(3):435–472, July 2006.

[Gen09] Craig Gentry. Fully homomorphic encryption using ideal lattices. In *STOC*, pages 169–178. ACM, 2009.

[GMW86] Oded Goldreich, Silvio Micali, and Avi Wigderson. How to prove all np-statements in zero-knowledge, and a methodology of cryptographic protocol design. In *Advances in Cryptology - CRYPTO '86, Santa Barbara, California, USA, 1986, Proceedings*, pages 171–185, 1986.

[Gol01] Oded Goldreich. *The Foundations of Cryptography - Volume 1, Basic Techniques*. Cambridge University Press, 2001.

[Gol04] Oded Goldreich. *The Foundations of Cryptography - Volume 2, Basic Applications*. Cambridge University Press, 2004.

[Gol16] Oded Goldreich. The uniform distribution is complete with respect to testing identity to a fixed distribution. In *Electronic Colloquium on Computational Complexity (ECCC)*, volume 23, page 1, 2016.

[Gol17] Oded Goldreich. Introduction to property testing (working draft). 2017. `www.wisdom.weizmann.ac.il/~oded/PDF/pt-v3.pdf`.

[GR00] Oded Goldreich and Dana Ron. On testing expansion in bounded-degree graphs. *ECCC 7(20)*, 2000.

[GRT18] Sumegha Garg, Ran Raz, and Avishay Tal. Extractor-based time-space lower bounds for learning. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pages 990–1002. ACM, 2018.

[Gut89] Michael Gutman. Asymptotically optimal classification for multiple tests with empirically observed statistics. *IEEE Transactions on Information Theory*, 35(2):401–408, 1989.

[Han87] Te Han. Hypothesis testing with multiterminal data compression. *IEEE transactions on information theory*, 33(6):759–772, 1987.

[HC70] Martin E Hellman and Thomas M Cover. Learning with finite memory. *The Annals of Mathematical Statistics*, pages 765–782, 1970.

[HKKN01] Shai Halevi, Robert Krauthgamer, Eyal Kushilevitz, and Kobbi Nissim. Private approximation of np-hard functions. In *STOC*, pages 550–559. ACM, 2001.

[IM08] Piotr Indyk and Andrew McGregor. Declaring independence via the sketching of sketches. In *Proceedings of the nineteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 737–745. Society for Industrial and Applied Mathematics, 2008.

[IMSW09] Yuval Ishai, Tal Malkin, Martin J. Strauss, and Rebecca N. Wright. Private multiparty sampling and approximation of vector combinations. *Theor. Comput. Sci.*, 410(18):1730–1745, 2009.

[Ind06] Piotr Indyk. Stable distributions, pseudorandom generators, embeddings and data stream computation. *J. ACM*, 53(3):307–323, 2006. Previously appeared in FOCS'00.

[IW06] Piotr Indyk and David Woodruff. Polylogarithmic private approximations and efficient matching. In *Proceedings of the Third Conference on Theory of Cryptography*, TCC'06, pages 245–264, Berlin, Heidelberg, 2006. Springer-Verlag.

[KMSZ08] Joe Kilian, André Madeira, Martin J. Strauss, and Xuan Zheng. Fast private norm estimation and heavy hitters. In *TCC*, volume 4948 of *Lecture Notes in Computer Science*, pages 176–193. Springer, 2008.

[KOR00] Eyal Kushilevitz, Rafail Ostrovsky, and Yuval Rabani. Efficient search for approximate nearest neighbor in high dimensional spaces. *SIAM J. Comput.*, 30(2):457–474, 2000. Preliminary version appeared in STOC'98.

[KRT17] Gillat Kol, Ran Raz, and Avishay Tal. Time-space hardness of learning sparse parities. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, pages 1067–1080. ACM, 2017.

[LRR13] Reut Levi, Dana Ron, and Ronitt Rubinfeld. Testing properties of collections of distributions. *Theory of Computing*, 9(8):295–347, 2013.

[LRR14] Reut Levi, Dana Ron, and Ronitt Rubinfeld. Testing similar means. *SIAM J. Discrete Math.*, 28(4):1699–1724, 2014.

[MM17] Dana Moshkovitz and Michal Moshkovitz. Mixing implies lower bounds for space bounded learning. In *Conference on Learning Theory*, pages 1516–1566, 2017.

[MNS11] Ilya Mironov, Moni Naor, and Gil Segev. Sketching in adversarial environments. *SIAM J. Comput.*, 40(6):1845–1870, 2011.

[NN01] Moni Naor and Kobbi Nissim. Communication complexity and secure function evaluation. *CoRR*, cs.CR/0109011, 2001.

[Raz16] Ran Raz. Fast learning requires good memory: A time-space lower bound for parity learning. In *Foundations of Computer Science (FOCS), 2016 IEEE 57th Annual Symposium on*, pages 266–275. IEEE, 2016.

[Raz17] Ran Raz. A time-space lower bound for a large class of learning problems. In *Foundations of Computer Science (FOCS), 2017 IEEE 58th Annual Symposium on*, pages 732–742. IEEE, 2017.

[Rub06] Ronitt Rubinfeld. Sublinear time algorithms. In *International Congress of Mathematicians*, volume 3, pages 1095–1110, 2006.

[Rub12] Ronitt Rubinfeld. Taming big probability distributions. *XRDS: Crossroads, The ACM Magazine for Students*, 19(1):24–28, 2012.

[RX10] Ronitt Rubinfeld and Ning Xie. Testing non-uniform k-wise independent distributions over product spaces. *Automata, Languages and Programming*, pages 565–581, 2010.

[RX13] Ronitt Rubinfeld and Ning Xie. Robust characterizations of k-wise independence over product spaces and related testing results. *Random Structures & Algorithms*, 43(3):265–312, 2013.

[Sha11] Ofer Shayevitz. On rényi measures and hypothesis testing. In *Information Theory Proceedings (ISIT), 2011 IEEE International Symposium on*, pages 894–898. IEEE, 2011.

[She12] Alexander A. Sherstov. The communication complexity of gap hamming distance. *Theory of Computing*, 8(1):197–208, 2012.

[She18] Or Sheffet. Locally private hypothesis testing. In *ICML*, 2018.

[Unn12] Jayakrishnan Unnikrishnan. On optimal two sample homogeneity tests for finite alphabets. In *Information Theory Proceedings (ISIT), 2012 IEEE International Symposium on*, pages 2027–2031. Ieee, 2012.

[Val11] Paul Valiant. Testing symmetric properties of distributions. *SIAM Journal on Computing*, 40(6):1927–1968, 2011. Previously in STOC'08.

[VY11] Elad Verbin and Wei Yu. The streaming complexity of cycle counting, sorting by reversals, and other problems. In *Proceedings of the twenty-second annual ACM-SIAM symposium on Discrete Algorithms*, pages 11–25. SIAM, 2011.

[Yao82] Andrew Chi-Chih Yao. Protocols for secure computations (extended abstract). In *23rd Annual Symposium on Foundations of Computer Science, Chicago, Illinois, USA, 3-5 November 1982*, FOCS '82, pages 160–164, 1982.

A Lower bound for Exact GHD for two-way communication protocols

We prove Lemma 4.2 here. We first prove the following claim.

Claim A.1. *Let $n \geq 1$ be even. Let $\beta = \beta(n) = \sqrt{n/2}/4$, and $\gamma = O(\sqrt{\log n})$. Consider a two-way communication protocol $\mathcal{A}$ that, with probability at least 0.9, for $x, y \in \{0,1\}^n$ with $\|x\|_1 = \|y\|_1 = n/2$, can distinguish between the case when $\|x - y\|_1 = n/2$ versus $\|x - y\|_1 - n/2 \in [\beta, \gamma\beta]$. Then $\mathcal{A}$ must exchange at least $\Omega(n/\log n)$ bits of communication.*

Proof. We will first prove a lower bound for a related distributional problem, and then show a reduction from this distributional problem. Consider the distributional problem where, for random $x, y \in \{0,1\}^{n/2}$, a deterministic communication protocol $\mathcal{D}$ satisfies the following:

- if $\|x-y\|_1 = n/4$, the protocol outputs -1 with probability at least $1 - 1/100\sqrt{n}$;
- if $|\|x-y\|_1 - n/4| \geq \beta(n)/2$, the protocol outputs $+1$ with probability at least $1 - 1/100\sqrt{n}$.

We will now show that any such $\mathcal{D}$ must use at least $\Omega(n)$ bits. We use the results of [She12]. Let μ be the (joint) distribution on x, y which are uniformly random from $\{0,1\}^{n/2}$. Now define $f_{n/2}(x,y)$ to be the partial function solved by the above assumed protocol $\mathcal{D}$: -1 if $\|x-y\|_1 = n/4$, and $+1$ if $|\|x-y\|_1 - n/4| \geq \beta(n)/2$. Theorem 3.3 from [She12] guarantees that, for any combinatorial rectange R with $\mu(R) \geq 2^{-\delta n}$, for small $\delta > 0$, we have that $\Pr_{(x,y)\in R}[|\|x-y\|_1 - n/4| \geq \beta] \geq 1/4$, for $\beta(n) = \sqrt{n/2}/4$. We now use the corruption bound (Theorem 2.3 from [She12]). In particular, we have that $\mu(R \cap f_{n/2}^{-1}(+1)) \geq 1/4 \geq 1/4 \cdot \mu(R) \cdot \mu(R \cap f_{n/2}^{-1}(-1))$. Also, $\mu(f_{n/2}^{-1}(-1)) \approx \frac{\sqrt{2}}{\sqrt{\pi n/2}}$. Hence we conclude that the communication complexity of $\mathcal{D}$ is at least

$$\delta n + \log\left(\mu(f_{n/2}^{-1}(-1)) - \tfrac{1/100\sqrt{n}}{1/4}\right) = \Omega(n),$$

where we used the fact that the failure probability of the protocol is $\xi = 1/100\sqrt{n} \ll \mu(f_{n/2}^{-1}(-1))$.

It remains to show that the claimed $\mathcal{A}$ exists, then there's also a protocol $\mathcal{D}$. In particular, suppose $\mathcal{A}$ exists. Then consider input strings $x, y \in \{0,1\}^{n/2}$ to $\mathcal{D}$, chosen from the uniform distributon μ. Consider string x' formed as x, followed by the negation of x; note that $\|x'\| = n/2$. Similarly, construct y'. Note that $\|x'\|_1 = \|y'\|_1 = n/2$, and that $\|x'-y'\|_1 = 2 \cdot \|x-y\|_1$. With probability at least $1-1/n$, we also have that $\|x'-y'\|_1 = 2\|x-y\|_1 \in [n/2 - O(\sqrt{n\log n}), n/2 + O(\sqrt{n \log n})]$, i.e., $\|x'-y'\|_1 - n/2 \in [-\gamma\beta(n), \gamma\beta(n)]$. Hence Alice and Bob can run the protocol $\mathcal{A}$, on inputs x', y', to distinguish whether $\|x'-y'\|_1 = n/2$ or $\|x'-y'\|_1 - n/2 \in [\beta(n), \gamma\beta(n)]$; note that this is precisely equivalent to distinguishing between $\|x-y\|_1 = n/4$ versus $\|x-y\|_1 - n/4 \in [\beta(n)/2, \gamma\beta(n)/2]$.

This is not enough though, as $f_{n/2}(x,y) = +1$ also when $\|x-y\|_1 \leq n/4 - \beta(n)/2$. Thus, Alice and Bob will instead run the protocol $\mathcal{A}$ on x', y', as well as on inputs x' and negation of y'. Thus, if $\|x-y\|_1 = n/4$, then $\mathcal{A}$ will return -1 both times, with probability at least 0.8. If $|\|x-y\|_1 - n/4| \in [\beta/2, \gamma\beta/2]$, then $\mathcal{A}$ will return $+1$ at least once, with probability at least 0.9. Hence, we say $\|x-y\|_1 = n/4$ iff $\mathcal{A}$ returns -1 both times.

The above reduction gives a 0.8 probability of success. We can amplify this to $1 - o(1/\sqrt{n})$ by running $O(\log n)$ independent copies of the (randomized) protocol reduction from above, and taking the median answer. This way we obtain a randomized protocol with success probability at least $1 - o(1/\sqrt{n})$ in solving problem $f_{n/2}$. We can further extract a deterministic protocol $\mathcal{D}$ that also achieves a success probability of $1 - o(1/\sqrt{n})$.

Overall, we conclude that since $\mathcal{D}$ has $\Omega(n)$ communication, the protocol $\mathcal{A}$ must have $\Omega(\frac{n}{\log n})$ communication, since the reduction uses $2 \cdot O(\log n)$ copies of $\mathcal{A}$.

□

We are now ready to prove Lemma 4.2.

Proof of Lemma 4.2. Assume the contrapositive: that for each interval $[l, r]$ with $r/l = 2$ and $l \in [\Theta(\sqrt{n}), \Theta(\sqrt{n \log n})]$, there exists a protocol $A_{[l,r]}$ which can distinguish $\|x-y\|_1 = n/2$ versus $\|x-y\|_1 - n/2 \in [l, r]$ with constant probability, using only $\leq C$ communication bits. For each such protocol, we can obtain boosted protocol $A'_{[l,r]}$ whose failure probability is $o(1/\log\log n)$ and communication complexity is $O(C \log\log\log n)$ — as usual, by running $O(\log\log\log n)$ independent copies of the protocol and taking the majority answer.

Now, in Lemma A.1, we have the "far" interval $I = [\sqrt{n/2}/4, \sqrt{n/2}/4 \cdot \Theta(\sqrt{\log n})]$. We partition this interval into $q = O(\log\log n)$ intervals $I_1, \dots I_q$, where each interval $I_i = [l_i, r_i]$ has $r_i/l_i \leq 2$. For each such interval, we have (by the above) a boosted protocol A'_{I_i} for $i \in [q]$.

We now construct a protocol $\mathcal{A}$ able to distinguish $\|x-y\|_1 = n/2$ versus $\|x-y\|_1 - n/2 \in I$. In particular, we run all protocols $A'_{I_1}, \dots A'_{I_q}$ on the same inputs. If $\|x-y\|_1 = n/2$, then all of them will return -1, except with probability $\leq q \cdot o(1/\log\log n) = o(1)$. On the other hand, if $\|x-y\|_1 - n/2 \in I$, then for $i \in [q]$ such that $\|x-y\|_1 - n/2 \in I_i$, the algorithm A'_{I_i} will return $+1$, except with probability $o(1/\log\log n)$. In other words, in the far case, at least one of the algorithms will return $+1$, with probability $1 - o(1)$. Hence we can distinguish these two cases with probability $1 - o(1)$.

The overall communication of $\mathcal{A}$ is $O(C \log\log n \cdot \log\log\log n)$. By Lemma A.1, we conclude that $C = \Omega(\frac{n}{\log n \log\log n \log\log\log n})$. $\square$